Survey on Automatic License Plate Detection Methods

Amritha Mary Davis
Dept. Computer Science and Engineering
Royal college of Engineering and Technology
Thrissur, India

Arun Vinodh C
Dept. Computer Science and Engineering
Royal college of Engineering and Technology
Thrissur, India

***Abstract*—License Plate Recognition (LPR) systems commonly have framework of processing steps such as: Detection of number plate, Segmentation of plate characters and Recognition of each character. Number plate detection is a challenging task due to diversity of plate formats and environmental conditions during the image acquisition. Accuracy of character segmentation and recognition rely on the efficiency of plate detection. Various algorithms are developed for this work. Purpose of this paper is to categorize and brief them**.

Keywords—License plate detection, Vertical edge matching, Histogram equalization, Gradient features, Plate template.

I. Introduction

License Plate Recognition is an advanced machine vision technology used to identify vehicles by their number plates without direct human intervention. This development of Intelligent Transportation System provides the data of vehicle numbers which can be used in follow up, analyses and monitoring. Besides traffic monitoring this technology is gaining popularity in various applications day by day, such as, highway toll collection, border and custom checkpoints, parking access control system [1]. Recently, with the increase of terrorism acts, it is strongly expected that LPR will gain great importance in security all over the world. LPR algorithm consists of the following three processing steps: 1) Number plate detection, 2) Character segmentation, and 3) Character recognition. The accuracy of plate extraction relies on the character segmentation and character recognition [2].

Detection of number plate is difficult task, due to the following reasons: 1) Generally number plate occupies a small portion of the whole image 2) Large varieties in number plate formats, styles and colors which differs from one country to another, 3) There are several other factors which could influence the efficiency of the extraction, are, blurry image, uneven or low illumination, vehicle motion, low resolution of the input image, distorted characters, dirty plates, shadow and reflection etc.

Various algorithms have been developed for number plate detection [3]. Generally, number plate detection algorithm consists of three state frameworks: 1) Preprocessing of the input image, 2) Number plate detection and 3) Extraction of number plate and plate validation. Literature survey is based upon the number plate extraction algorithm that follows three states framework. Methods have been evaluated at different conditions such as size of input image, distance between camera and vehicle, different illumination conditions, type of name plate.

The paper is organized as follows: Section 2 explains various algorithms employed for number plate detection. Section 3 concludes the paper followed by references.

II. METHODS FOR NUMBER PLATE DETECTION

Character segmentation and recognition relies on the accuracy of number plate extraction. Hence this processing step is critical one and emphasizes on developing numerous techniques so that number plate detection is faithful. Methods to detect the number plate are based on: vertical edge matching, histogram equalization, gradient features and template matching. In this paper we have covered the brief addressing of these methods.

A. Vertical Edge Matching based

License plate recognition (LPR) has many applications in traffic monitoring systems. In this paper [4], a vertical edge matching based algorithm to recognize license plate from input gray-scale image is proposed. The algorithm is able to recognize license plates in normal shape, as well as plates that are out of shape due to the angle of view. The algorithm is fast enough, so the recognition unit of a LPR system can be implemented only in software hence the cost of the system is reduced.

For a given gray-scale image its corresponding edge image can be obtained by using Prewitt or other masks. It is noticed that most of vehicles usually have more horizontal lines than vertical lines. If the two vertical edges of a license plate can be detected correctly, four corners of the plate can then be located, so that the license plate can be extracted exactly from the input image even if it is out of shape. Thus, as an alternative, only the vertical edges of input image are used to extract license plate.

Binary size-and-shape filter is very useful in pattern recognition, because it is usually needed to recognize objects with special shapes in images. As a pre-processing, the filter removes objects that do not satisfy some specific features, so makes it easy to recognize objects in post processing. The algorithm proceeds as follows: the contiguous horizontal run of pixels containing the starting seed is filled in. Then the row above the just-filled run is examined from right to left to find the rightmost pixel of each run, and these pixel addresses are stacked. [5] The same is done for the row below the just-filled run. When a run has been processed in this manner, the pixel address at the top of the stack is used as a new starting seed. When the stack is empty, the algorithm terminates.

In a binary image, a region is defined as a set of white pixels that are eight connected with each other. To filter the image, each region is browsed by using the seed filling algorithm, during which the information about the region is collected according to the specified features, such as width, height, coordinates of significant points, slope, etc, with respect to the interested object. If the region does not satisfy the specified features, it is removed as noise by filling it with black. Otherwise, the region is kept and its information is also recorded for post-processing. For the binary image, $\{\mathbf{E_{I,J}}\}$ the size-and-shape filter based on seed filling algorithm is described as follows:

1) Search the entire image row by row, for each white pixel $\{\mathbf{E_{I,J}}\}$ in image, if it has not been checked, then run over the eight connected white region by using seed filling algorithm in which $\{\mathbf{E_{I,J}}\}$ is adopted as the first starting seed of the region.

2) Check whether the region have specified features. If it does not satisfy some predefined restricted conditions, then fill the region with black, that is, remove the region as noise, since it is impossible to be the region of interest (ROI). Otherwise left it in the image as a possible interested object and record its information for pos t-processing. After that, mark whole pixels of this region as checked pixels.

3) Continue to scan the image row by row to find another unchecked white pixel as the first starting seed of a new region, until all white pixels in the image have been checked.

In license plate extraction, after the vertical edge image has been obtained, it is filtered so as to remove the edges that are impossible to be the vertical edge of a license plate. Before filtering, morphological operation “dilation” is applied as a pre-processing [6]. In the vertical edge image, an edge area is defined as a set of white pixels that are eight connected neighbors with each other. For each edge area, we check its size and shape by the seed filling based filter.

If the edge area is smaller than a predefined size, or does not form a beeline whose slope is within a predefined interval, then it is removed, since it is impossible to be the vertical edge of a license plate. Otherwise, the coordinates of the top and bottom pixels of the edge area are recorded for edge matching in post-processing. Since the vertical edges of a license plate may be cut off in the vertical edge image, edge areas are also recorded as one edge area if they have similar slope and one’s top is quite close to another’s bottom.

The ratio of width to height of license plate is about 2: 1, it can be used to judge whether two edge areas are the pair of vertical edges of a license plate. Consider that the real ratio in the image may deviate from the standard value due to the angle of view; the possible range of the ratio is adapted from 1.4: 1 to 3.3: 1. It is assumed that license plate does not lean quite a lot in input image; therefore, the vertical coordinates of the two vertical edges of a license plate should have small difference.

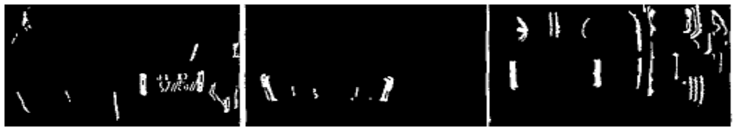
It is restricted that the difference should be within half of the plate’s height. Additionally, the two vertical edges of a license plate will have similar height within the vertical edge image, which is also a restricted condition of the edge matching.

Here, the height ratio of plate’s two vertical edges is from 0.8 to 1.2. Let L be an edge area, X and Y be its corresponding horizontal and vertical coordinates, Top, Bottom and Middle represent its top, bottom and middle respectively and the top has lower vertical coordinate than the bottom. Let N be the total number of edge areas in a vertical edge image.

(a) Input gray scale image

(b) Edge images

(c) Vertical edge images



(d) Result of Size-and-Shape filtering

(e) Extracted license plates

Examples of License plate extraction.

Here, edge areas in vertical edge image are matched with each other according to the above restricted conditions. If a pair of edge areas satisfies the conditions, it is regarded as the possible vertical edges of license plate. Otherwise, another pair is checked. After the possible vertical edges of license plate have been found, the region is extracted and continuously checked whether it is real license plate region during plate segmentation and recognition.

B. Gradient features and Plate Template based

The templates are used to format your paper and style the text. All margins, column widths, line spaces, and text fonts are prescribed, please do not change them. You may note peculiarities. For example, the head margin in this template measures proportionately more than is customary. This measurement and others are deliberate, using specifications that anticipate your paper as one part of the entire proceedings, and not as an independent document. Please don’t revise any of the current designations.

The approach focuses on dealing with images taken under a weak lighting condition. The proposed method is divided into two steps:

1) Searching candidate areas from the input image using the gradient information.

2) Determining the plate area among the candidates we obtained and adjusting the boundary of the area by introducing a plate template.

A two-stage search process is proposed in the approach [7]: the global search in which candidates of the plate region are found using the combination of the gradient features, and the local search where the correct plate region is selected from the candidates and the boundary of the region is adjusted using a plate template.

In the starting part of the stage, histogram stretching is performed as a pre-processing, in order to increase the contrast of the image and protect the important edges from subsequent operations.

Step 1: Obtaining three statistical features

Conventional approaches, which rely only on the variance of the gradient while ROIs being searched, could result in limited success when the image under consideration contains a tilted plate and/or complicated background. Therefore three statistical features are introduced in the proposed approach and combined by a simple neural network.

1) Feature 1 - the gradient variance:

The gradient of the image is obtained using the Sobel operator. Based on the observation that gray scale changes are more frequent in the plate area, local variances are obtained from windows of size 1x9 pixels. The window size is determined experimentally.

2) Feature 2 - the density of edges:

Plate regions tend to have a high density of edges. The density is measured in a block of size 1x9 pixels by summing all edge pixels.

3) Feature 3 - the density variance:

The feature is based on the concept that if a candidate area indeed contains a plate, the foreground pixels are distributed evenly comparing to the areas with simple structures. Therefore, this feature can be used to discriminate text regions from background regions. To obtain the feature, a block of size 9x9, centered at (i,j), is divided into nine equal-sized sub-blocks. For each sub-block i , let d_i denote the total number of edge points in the sub-block. Then the feature, the density variance, is defined.

Those three features are combined with a neural network to determine if a particular pixel under examination could belong to the plate region. To train the network, around 300,000 patterns were collected from 50 images. The output of the network forms a map, which is called the feature map hereafter.

Step 2: Finding candidate areas

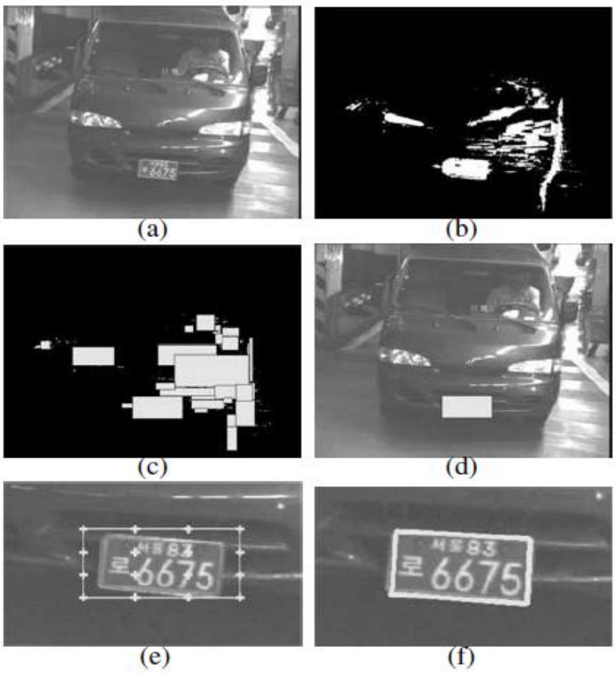
In the feature map, areas with higher variance are grouped and labeled using chain code-based contour follower. While the processing is being performed, careful examination is carried out to apply certain limitation on the size of candidates. The examination prevents from creating candidates with too big or small aspect ratio comparing to that of the license plate.

Step 3: Selecting an ROI

Candidate areas are represented by bounding boxes and a plate region may consist of one or more than one boxes as shown in Figure 2.4(left). To merge the boxes, a simple neural network is employed and it determines whether two boxes under examination need to be merged or not. As inputs to the neural network, corner points of the boxes, aspect ratios, and box filling ratios are provided.

After the merging is completed, the boxes with both the aspect ratio of less than 1/2 and horizontal length of less than 200 pixels are used for the next step [8]. A region of interest is determined by another back-propagation neural network which inputs box diagonal indices, average value of intensities and box filling ratio.

Once an ROI is selected, it is necessary to make sure that the box contains the plate. The similarity transform is performed to the plate template so that it fits into the ROI selected. Then, projection profiles on both directions are examined to judge whether the ROI contains the plate or not. When the parameters collected from the projection profiles satisfy the conditions shown in Figure 2.5, the ROI is regarded as the correct plate region. If the ROI does not satisfy either of the conditions, the process is repeated until the selected ROIs are obtained.



Processing steps: (a) an input image, (b) the feature map, (c) bounding boxes, (d) selected ROI, (e) initial setting for the local search, and (f) the local search result.

C. Gradient features and Plate Template based

Preprocessing is mainly used to enhance the processing speed, improve the contrast of the image, and reduce the noise caused by equipment or environment [9]. It is composed of the following three steps.

1) Searching Range Reduction: In general, the interested object is placed at the center of a picture when a shot is made. Hence, in this paper, the region of interest is fixed at the center 4/9 area.

2) Histogram Equalization Enhancement: Many studies adopted conventional contrast stretching to enhance the low contrast images.

3) Median Filtering: Some images may undergo noise interference. To solve the problem, many filters are possible candidates, e.g., uniform filter, Gaussian filter, median filter, etc. In this paper, the median filter is adopted to remove the noise, since it preserves most of the fidelity of an image.

In this subsection, the color information and textures of LP are employed to further locate the exact position.

1) Three-Map Retrieving: The LPs in Taiwan are all in the same color distribution, i.e., black characters and white background. In this paper, the Saturation map (S-map), Intensity map (I-map), and Edge map (E-map) are employed for CS. Among these, the S- and I-maps are obtained by converting the RGB color model to the HSI model.

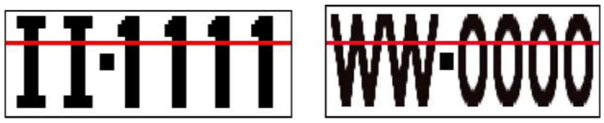
2) Non-LP Region Removing: In general, LP has the strongest gradient in the E-map. The high-gradient-averaging (HGA) method is applied to remove the non-LP regions. The first HGA removes regions with lower gradients than the average gradient of the E-map. The second HGA further removes regions with lower gradients than the average gradient of regions with higher gradients than the average gradient of the original E-map, and so forth.

Each character in LP has 35 possibilities (0–9, A–Z, where the letter O and the number 0 are considered the same). If the LP is scanned with a horizontal line, the number of black to white (or white to black) is at least six and at most 14. Based on this observation, a horizontal line in the E-map is reserved with six–14 black-to-white switched numbers. In the same way, a vertical line in the E-map is reserved with one to three black-to-white switching numbers. [10] The remaining areas in the E-map are then connected with the following method.

Suppose that the distances between each character of LP are lower than a predefined parameter *CD*. Two white pixels in the same horizontal line of the E-map with distance lower than *CD* are connected. Hence, many objects can be obtained with this method. These regions are then filtered according to the aspect ratio and area of the LP.

The aspect ratio of the LP is generally around 2.8–4.8. On the other hand, according to our experiments, the extracted LP of size lower than $22 \times 50 = 1100$ pixels is difficult to recognize.

Hence, reserved the objects that meet the aspect ratio and with an area higher than 1100 pixels. These reserved objects are named candidate objects. Each horizontal line of a candidate object is checked if the black-to-white switched number is between six and 14. The object with the highest percentage (more than 80% horizontal lines) with this feature is determined as the LP.



(a) (b)

Example of characters crossed by a horizontal line. (a) Six black-to white switched numbers. (b) Fourteen black-to-white switched numbers.

Finally, complete content and organizational editing before formatting. Please take note of the following items when proofreading spelling and grammar:

D. Vertical Edge Detection (VEDA) based

The first contribution is that it proposes a fast vertical edge detection algorithm (VEDA) based on the contrast between the grayscale values, which enhances the speed of the CLPD method [11]. After binarizing the input image using adaptive thresholding (AT), an unwanted-line elimination algorithm (ULEA) is proposed to enhance the image, and then, the VEDA is applied. The second contribution is that CLPD method processes very-low-resolution images taken by a web camera. After the vertical edges have been detected by the VEDA, the desired plate details based on color information are highlighted. Then, the candidate region based on statistical and logical operations will be extracted.

After the color input image is converted to grayscale, an AT process [12] and [13] is applied to constitute the binarized image.

Technique of AT: The AT technique used in this paper is just a simple extension of Bradley and Roth's [12] and Wellner's methods [14]. The idea in Wellner's algorithm is that the pixel is compared with an average of neighboring pixels. Specifically, an approximate moving average of the last S pixels seen is calculated [15] while traversing the image. If the value of the current pixel is T percent lower than the average, then it is set to black; otherwise, it is set to white. This technique is useful because comparing a pixel to the average of neighboring pixels will keep hard contrast lines and ignore soft gradient changes.

The advantage of this technique is that only a single pass through the image is required. Wellner uses one eighth of the image width for the value of S and 0.15 for the value of T to yield the best results for a variety of images. The value of T might be a little bit modified from the proposed value by Wellner depending on the used images; whereas it should be in the range $0.1 < T < 0.2$ in this method.

However, Wellner's algorithm depends on the scanning order of pixels. Since the neighborhood samples are not evenly distributed in all directions, the moving average process is not suitable to give a good representation for the neighboring pixels. Therefore, using the integral image in [12] has solved this problem.

Thresholding process in general produces many thin lines that do not belong to the LP region. In Fig. 2(b), we can see that there are many long foreground lines and short random noise edges beside the LP region. These background and noise edges are unwanted lines. These lines may interfere in the LP location. Therefore, it has proposed an algorithm to eliminate them from the image.[16] This step can be considered as a morphological operation and enhancement process. There are four cases in which unwanted lines can be formed. In the first case, the line is horizontal with an angle equal to 0◦ as (−).

In the second case, the line is vertical with an angle equal to 90◦ as (|). In the third case, the line is inclined with an angle equal to 45◦ as (/). In the fourth case, the line is inclined with an angle equal to 135◦ as (\). Therefore, the ULEA has been proposed to eliminate these lines. In this step, while processing a binary image, the black pixel values are the background, and the white pixel values are the foreground. A 3 × 3 mask is used throughout all image pixels.

After thresholding and ULEA processes, the image will only have black and white regions, and the VEDA is processing these regions. [17] The idea of the VEDA concentrates on intersections of black–white and white–black. A 2 × 4 mask is proposed for this process.

The center pixel of the mask is located at points (0, 1) and (1, 1). By moving the mask from left to right, the black–white regions will be found. Therefore, the last two black pixels will only be kept. Similarly, the first black pixel in the case of white–black regions will be kept.

After applying the VEDA, the next step is to highlight the desired details such as plate details and vertical edges in the image. The HDD performs NAND–AND operation for each two corresponding pixel values taken from both ULEA and VEDA output images.

This process depends on the VEDA output in highlighting the plate region. All the pixels in the vertical edge image will be scanned. [18] When there are two neighbor black pixels and followed by one black pixel, as in VEDA output form, the two edges will be checked to highlight the desired details by drawing black horizontal lines connecting each two vertical edges.

This process is divided into four steps as follows.

1) Count the Drawn Lines per Each Row: The number of lines that have been drawn per each row will be counted and stored in matrix variable HwMnyLines[a], where a = 0, 1, ... height-1.

2) Divide the Image into Multigroups: The huge number of rows will delay the processing time in the next steps. Thus, to reduce the consumed time, gathering many rows as a group is used here.

3) Count and Store Satisfied Group Indexes and Boundaries: Most of the group lines are not parts of the plate details. Therefore, it is useful to use a threshold to eliminate those unsatisfied groups and to keep the satisfied groups in which the LP details exist in. Each group will be checked; if it has at least 15 lines, then it is considered as a part of the LP region. Thus, the total number of groups including the parts of LP regions will be counted and stored. The remaining groups after thresholding step should have the LP details. Therefore, their locations are stored. [19] The final step here is to extract both upper and lower boundaries of each satisfied group by using its own index.

4) Select Boundaries of Candidate Regions: This step draws the horizontal boundaries above and below each candidate region. There are two candidate regions interpreted from horizontal-line plotting, and these conditions require an additional step before the LP region can be correctly extracted.

This process aims to select and extract one correct LP. The process is discussed in five parts. The first part explains the selection process of the LP region from the mathematical perspective only. The second part applies the proposed equation on the image. The third part gives the proof of the proposed equation using statistical calculations and graphs. The fourth part explains the voting step. [20] The final part introduces the procedure of detecting the LP using the proposed equation.

CONCLUSION

Different license plate detection methods are compared and found out that license plate detection using VEDA is best among them. The system in gradient features and plate template based method work under certain fixed scale and license plate template is difficult to construct. But VEDA does not have any constraints.

Similarly the system in vertical edge matching based method fails to detect unclear edges of license plate but VEDA detects edges even for blur images. The system in vertical edge matching based method also sometimes edges are eliminated by considering it as noise. This situation can be avoided in VEDA. The system in histogram equalization and feedback self-learning based has high complexity.

A new and fast algorithm for vertical edge detection is proposed, in which its performance is faster than the performance of Sobel by five to nine times depending on image resolution. The VEDA contributes to make the whole proposed CLPD method faster.

ACKNOWLEDGMENT

Authors are thankful to Media Lab LPR database for providing the database for research work. They are also thankful to Royal College of Engineering, C.S.E Department, Calicut University, for providing necessary infrastructural facilities during the progress of the work.

REFERENCES

[1] K. V. Mardia and T. J. Hainsworth, A spatial thresholding method for image segmentation, *IEEE Trans. Pattern Anal. Mach. Intell., vol. 10*, no. 6, pp. 919–927, 1988.

[2] H. Bai and C. Liu, *A hybrid license plate extraction method based on edge statistics and morphology*, Proceedings of the 17th IEEE Computer Society Int. Conf. Pattern Recognition (Page: 831-834, Year of Publication: 2004, ISBN: 0-7695-2128-2).

[3] M. Fukumi, Y. Takeuchi, H. Fukumoto, Y. Mitsura, and M. Khalid, *"Neural network based threshold determination for Malaysia license plate character recognition*," Proceedings of the 9th Int. Conf. *Mechtornics echnology* (Page : 1-5, Year of Publication: 2005, ISBN: 0-7796-5312-7).

[4] M. Yu and Y. D. Kim, *An approach to Korean license plate recognition based on vertical edge matching*, Proceedings of the IEEE Computer Society Int. Conf. System, Man, and Cybernetics (Page: 2975 - 2980 , Year of Publication : 2000 ISBN: 0-7803-6583-6).

[5] T. Naito, T. Tsukada, K. Yamada, K. Kozuka, and S. Yamamoto, Robust license-plate recognition method for passing vehicles under outside environment, *IEEE Trans. Veh. Technol., vol. 49*, no. 6, pp. 2309–2319, 2000.

[6] H.-H. P. Wu, H.-H. Chen, R.-J. Wu, and D.-F. Shen, *License plate extraction in low resolution video,* Proceedings of 18th IEEE Computer Society Int. Conf. Pattern Recognition (Page: 824-827, Year of Publication: 2006, ISBN: 0-7695-2521-0).

[7] S. Kim, D. Kim, Y. Ryu, and G. Kim, *A robust license-plate extraction method under complex image conditions*, Proceedings of the 16th IEEE Computer Society Int. Conf. Pattern Recognition (Page: 216 - 219 . Year of Publication : 2002 ISBN: 0-7695-1695-X).

[8] H.-H. P. Wu, H.-H. Chen, R.-J. Wu, and D.-F. Shen, *License plate extraction in low resolution video,* Proceedings of 18th IEEE Computer Society Int. Conf. Pattern Recognition (Page: 824-827, Year of Publication: 2006, ISBN: 0-7695-2521-0).

[9] J.-M. Guo and Y.-F. Liu, License plate localization and character segmentation with feedback self-learning and hybrid binarization techniques, *IEEE Trans. Veh. Technol., vol. 57*, no. 3, pp.1417–1424, 2008.

[10] F. F. Moghaddam, O. Karimi, *An Efficient and Intelligent Model to Control Driving Offenses by Using Cloud Computing Concepts*, Proceedings of 4th IEEE computer Society Control and System Graduate Research Colloquium (ICSGRC) (Pages:17-20, Year of Publication: 2013, ISBN: 978-1-4799-0550-8).

[11] M. Al-Ghaili, S. Mashohor, A. R. Ramli, and A. Ismail, Vertical-Edge-Based Car-License-Plate Detection Method, *IEEE transaction on vehicular technology, vol. 62,* no. 1, pp.26-38, 2013.

[12] D. Bradley and G. Roth, Adaptive thresholding using the integral image, *J. Graph. Tools*, *Vol. 12*, no. 2, pp. 13–21, 2007.

[13] F. Shafait, D. Keysers, and T. M. Breuel, *Efficient implementation of local adaptive thresholding techniques using integral images*, Proceedings of SPIE Int. Conf. Document Recognition and Retrieval (Page: 681510-1–681510-6., Year of Publication: 2007, ISBN: 0-7562-3587-4).

[14] P. D. Wellner, *Adaptive thresholding for the DigitalDesk*, Technical Report EPC-1993-110 Rank Xerox Research Centre, Cambridge Laboratory, Birmingham, U.K., 1993.

[15] F. Porikli and O. Tuzel, *Fast construction of covariance matrices for arbitrary size image windows*, Proceedings of IEEE Computer Society Int. Conf. Image Processing (Page:1581–1584, Year of Publication: 2006, ISBN: 1-4244-0480-0).

[16] H. Bai and C. Liu, *A hybrid license plate extraction method based on edge statistics and morphology*, Proceedings of the 17th IEEE Computer Society Int. Conf. Pattern Recognition (Pages: 831-834, Year of Publication: 2004, ISBN: 0-7695-2128-2).

[17] H. Zhang, W. Jia, X. He, and Q. Wu, *A fast algorithm for license plate detection in various conditions,* Proceedings of the IEEE Computer Society Int. Conf. Syst., Man(Pages:2420-2425, Year of Publication: 2006, ISBN: 1-4244-0099-6).

[18] P. Comelli, P. Ferragina,M. N. Granieri, and F. Stabile, Optical recognition of motor vehicle license plate, *IEEE transaction on vehicular technology, vol. 44*,no. 4, pp. 790-799,1995.

[19] K. I. Kim, K. Jung, S. H. Park, and H. J. Kim, Support vector machines for texture classification, *IEEE transaction on Pattern Anal. Mach. Intell., vol. 24*, no. 11, pp. 1542–1550, 2002.

[20] S.-L. Chang, L.-S. Chen, Y.-C. Chung, and S.-W. Chen, Automatic license plate recognition, *IEEE Trans. Intell. Transp. Syst., vol. 5*, no. 1, pp. 42–53, 2004.