Comparison of two commonly used techniques for Vehicle License Plate Recognition

Jaya Gupta[1], Prof. Supriya Agrawal[2]
Computer Engineering Department, SVKM's NMIMS University Mumbai, Maharashtra, India
1. jayagupta286@gmail.com 2. supriya.agrawal@nmims.edu

Abstract: *Many algorithms and techniques have been developed to deal with the problem of extracting and recognizing the license plate characters from images of cars. Vehicle license plate recognition is the essence of present world vehicle related security and vehicle management systems. In this paper we compare two commonly used algorithms. Both the algorithms have been tested on more than 400. Presence of many types of license plate makes the problem difficult to solve. We have made some assumptions in the algorithms like a standard license plate is present with light background and darker characters, not more than two rows are present, same type of characters are present all through, etc. One algorithm uses morphological operators for edge detection and the other one uses sliding concentric windows technique.*

Index Terms: License plate recognition, edge detection, character recognition, tilt correction, parking lot automation, morphological operations, Sliding concentric windows, position histograms

I. INTRODUCTION

Vehicle license plates have been recorded in various places like parking lots, tolls, entry to apartment complexes as security or maintenance purposes. Earlier this was done only manually with people sitting and writing down the number but in today's world of automation there are so many techniques being developed to get things done manually. The efficiency of these systems can be compared by taking a count of the number of plates segmented from a collection of images and the speed with which it gives results. The two algorithms being compared here use different methods for edge detection and character segmentation. One is the use of sliding concentric windows to find the edges and the other is using morphological operations for the same. Both can be divided in same kind of modules, which are (a) Image acquisition (b) Image preprocessing (c) Edge detection (d) License Plate segmentation (e) Character segmentation and (f) Character recognition. The rest of this report is organized as follows. The next section (Section II) composes a review of similar research that have been implemented and tested for vehicle license plate recognition. Section III explains and compares the two algorithms in each of the above modules. Section IV lists the results obtained by the two algorithm and example images and Section V concludes the paper.

II. RELATED RESEARCH

A lot of research has been done in the field of vehicle license plate recognition system. This survey [2] compares many methods being used for the license plate location and extraction. [1], [3], [4], [5] have done research on improving the algorithms using sliding concentric windows which is one of the algorithms being compared here. Sobel based edge detection methods are also used frequently [11], [16] but the limitations to this method are many including proper lighting, good resolution, good contrast between the license plate and the surrounding car color. [6] Uses wavelet transforms but has a limitation on the distance between the camera and the car. Another method being explored here is the extraction using feature based morphological operations, which is the most commonly studies methods of vehicle license plate recognition systems [7], [9], [10], [13], [14]. Position histograms [8] have been used to reduce the steps involved in the process and give good results but only for very limited images. Color based [12] methods have also been developed. For character segmentation position histograms [1] have been used but most popular is the use of bounding boxes in connected components analysis. For character recognition, various methods have been used [17]. Correlation, neural networks [1], [18] and support vector machine [15] are being used widely.

III. COMPARISION OF THE ALGORITHM STEPS

Both the algorithms employ an edge detection technique and follow the same steps as listed in the introduction hence can be compared easily step by step. Further on we will be calling the algorithm using morphological operator for edge detection as the algorithm 1 and the one with sliding concentric windows as algorithm 2.

a. Image acquisition:

Both the algorithms have be tested on the same set of images taken by a camera with good resolution, images are taken in varying lighting conditions and the distance of the cars from the camera is varied from 1-7 m.

b. Image Preprocessing:

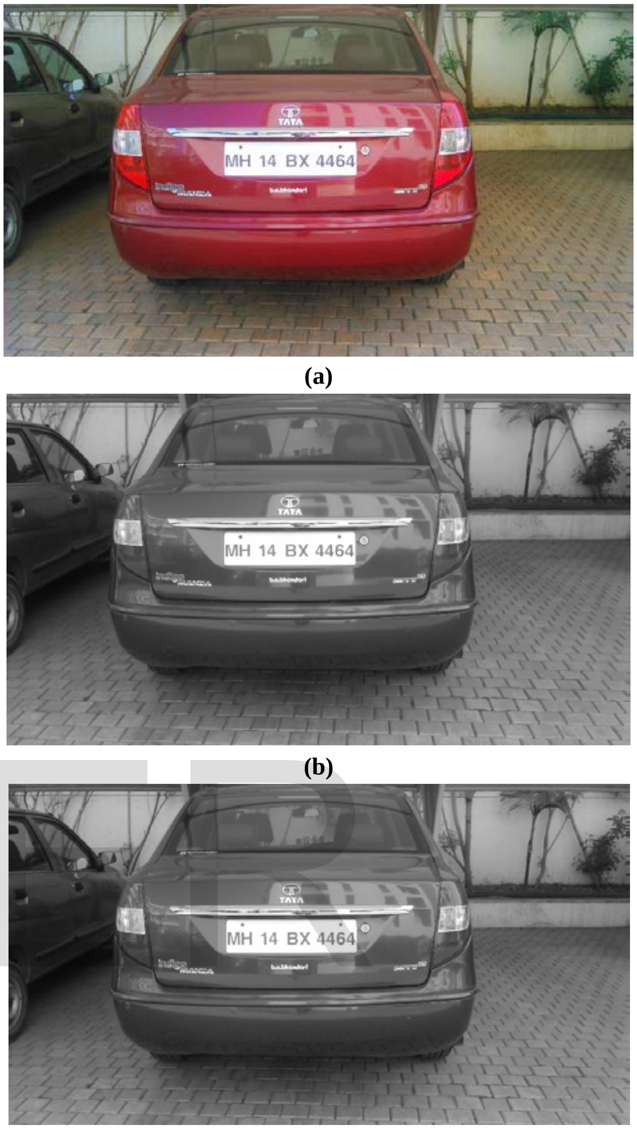
Input RGB image was converted to gray-scale and was used for further processing the same way in both the algorithms. This conversion was done using equation (1) which converts the RGB values to gray values by forming a weighted sum of the R, G and B components. Result is shown in Fig. 1.

$$G = 0.2989 * R + 0.5870 * G + 0.1140 * B \qquad (1)$$

After conversion to gray-scale the image was passed through a 3x3 median filter in case of the algorithm 1 to remove noise and smooth the image for a better performance of morphological operators.

c. Edge detection:

Algorithm 1 uses morphological operations like dilation and erosion for finding the edges. Erosion replaces the current pixel with the minimum pixel value found in the defined pixel set. Dilation is the complementary operator, and it replaces the current pixel with the maximum pixel value found in the defined pixel set.



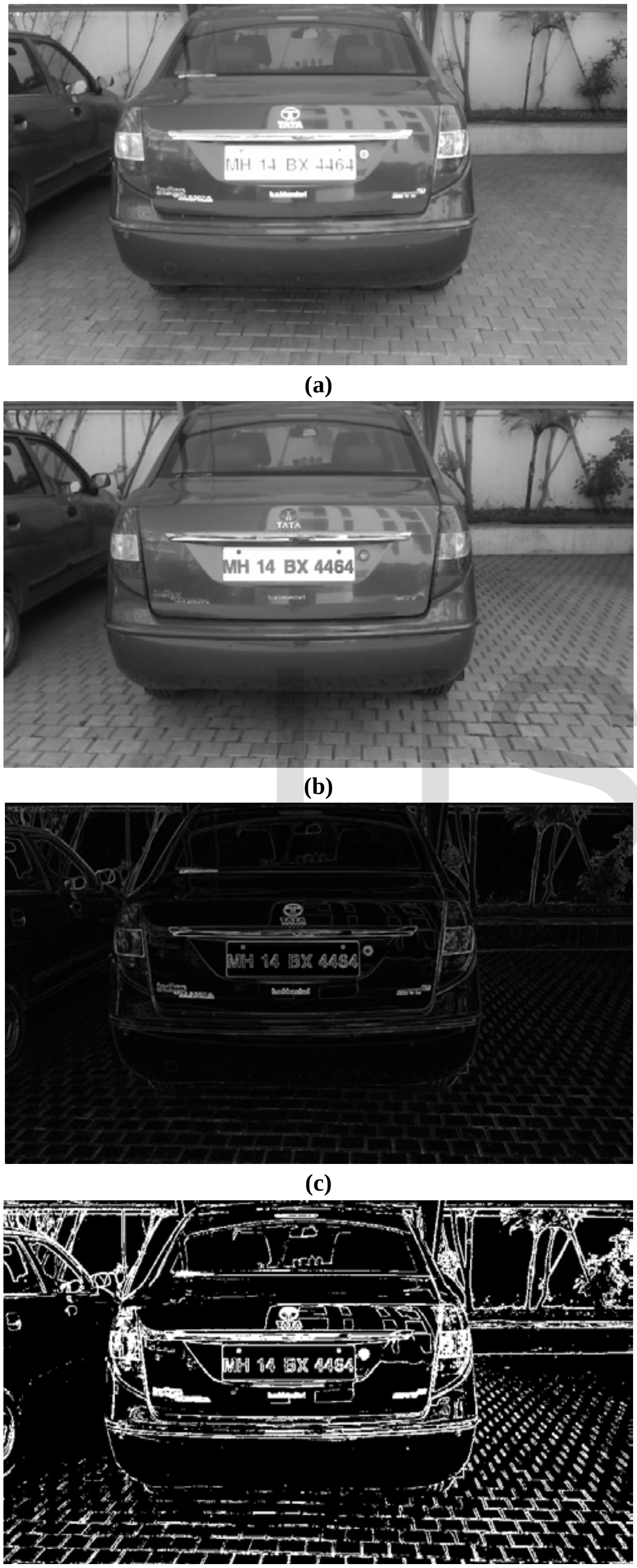
(a)

(b)

(c)

Fig. 1: (a) Original RGB Image, (b) Gray-scale and filtered image (Algorithm 1), (c) Gray-scale image (Algorithm 2)

A disk structuring element is used for the dilation and erosion in algorithm 1. A structuring element is defined as a configuration of pixels on which an origin is defined. The structuring element can be of any shape, but most often, a simple shape such as a square, circle/disk, line or diamond with the origin at the center is used for ease of use. After obtaining the dilated (Fig. 2(a)) and eroded (Fig. 2(b)) images, the eroded image is subtracted from the dilated image to get the gray-scale edge image (Fig. 2(c)). This gray-scale image is then binarized using Otsu's global image thresholding method (Fig. 2(d)).



(a)

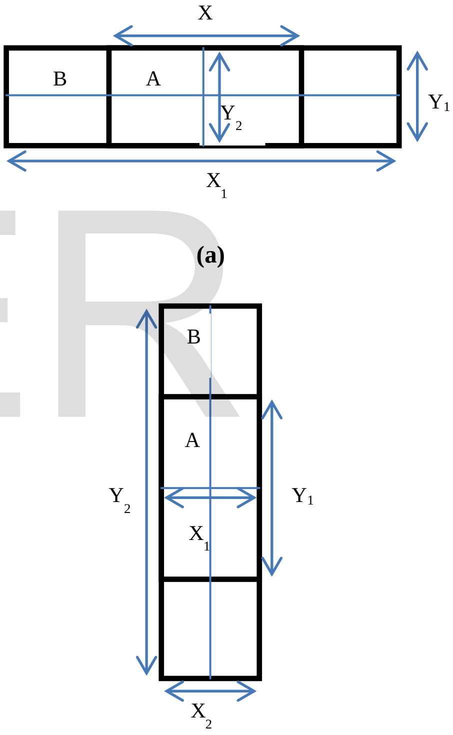
(b)

(c)

(d)

Fig. 2: (Algorithm 1) (a) Dilated image, (b) Eroded image, (c) Gray-scale edge image, (d) Binarized edge image

In algorithm 2 an image segmentation method called Sliding Concentric Windows (SCW) is used, which extracts vertical and horizontal edges in the image by comparing statistical properties, like mean or standard deviation, inside two concentric windows. For testing we have used mean since standard deviation was producing a lot of noise in the image. Two concentric windows were created with a size (X1×Y1) and(X2×Y2) for vertical and two for horizontal edge detection as shown in the Fig. 3(a) and (b). Smaller window will be called A and the bigger one as B.



(a)

(b)

Fig. 3: (a) Sliding Concentric Windows for detection of vertical edges and (b) horizontal edges (Algorithm 2)

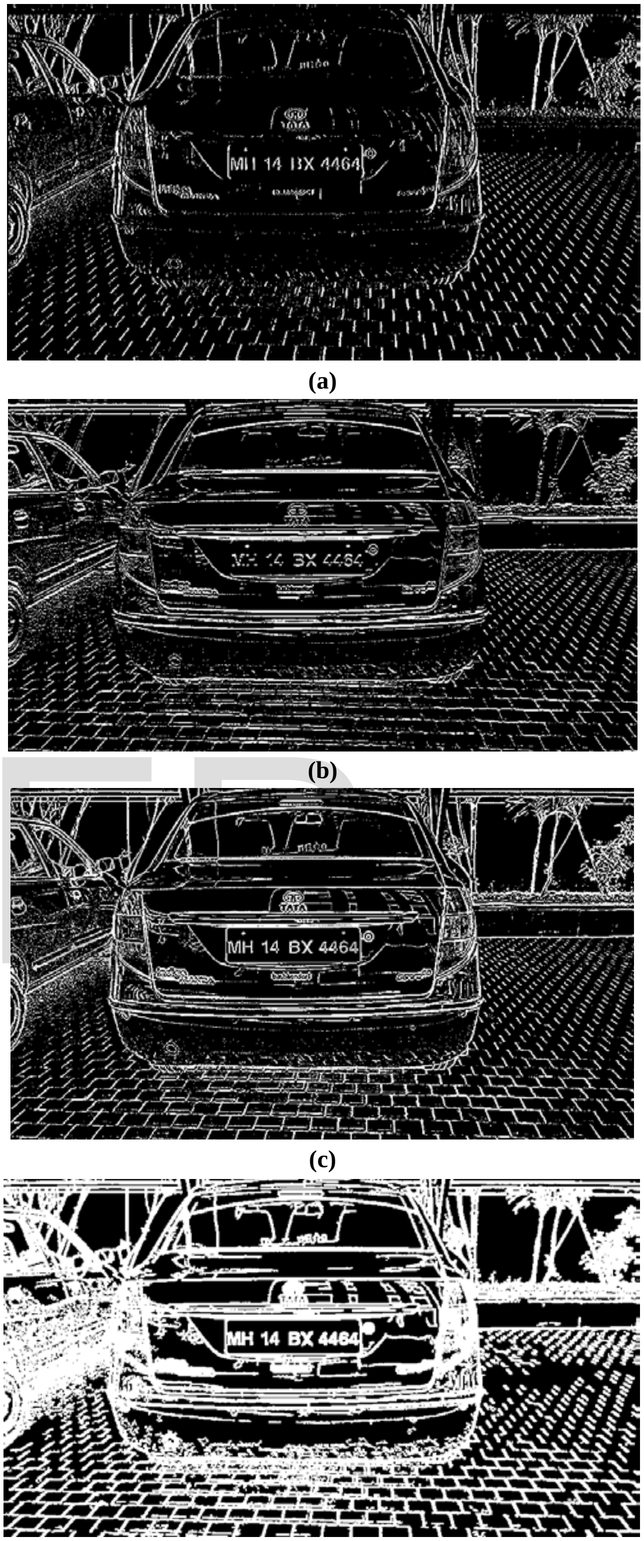
The size of the small widow was chosen to be proportional to the size of the license plate. The aspect ratio is defined as the ratio of the width of the license plate, which is roughly close to 1:4 for most of the test images and that of the larger one was made to have length double the size of smaller one. Here, the sizes used by us are:

For vertical edge detection, windows shown in Fig. 3(a) were chosen such that
X1 = 8, X2 =16 and Y1=Y2=2.
For horizontal edge detection, windows shown in Fig. 3(b) were chosen such that
X1 = X2 =2, Y1= 8 and Y2=16.
All the pixels in the gray scale image were selected one by one and the mean of the pixels in the above windows around that pixel was computed (separately done for vertical and horizontal edges). If the mean/standard deviation ratio of the two windows exceeds a threshold, set by trial and error, then the central pixel of the windows was considered to belong to a vertical or horizontal edge region. If (x, y) are the coordinates of the observed pixel in input gray scale vehicle image I, the pixel value in the respective coordinates (x', y') of the resulting image I1 was set to either 0 (no edges) or 1 (edges) according to the following equation:

$$I_1(x', y') = \begin{cases} 1, & \text{if } \dfrac{\text{mean}(B)}{\text{mean}(A)} > T \\ 0, & \text{Otherwise} \end{cases} \qquad (2)$$

After the two images (Vertical edge image and Horizontal edge image) were obtained (Fig. 4 (a) and (b)), OR masking was used to merge the images and obtain the final edge image (Fig. 4(c)) having 1's at all the edges and 0's at other places. The image is then dilated to fill gaps in the license plate boundary, if any (Fig. 4(d)).

After the edge detection, both the algorithms go through a set of steps to eliminate the unnecessary components to make the license plate extraction faster. In both the algorithms, the binary edge image is inverted (Fig. 5) to make the license plate as one connected components surrounded by a black border. Now, the areas that are assumed to be too small and too big when compare the license plate (areas greater than 20000 pixels and the ones less than 400 pixels) are removed from the edge image (Fig. 6). After this step we have the final image on which more processing is done to extract the license plate.



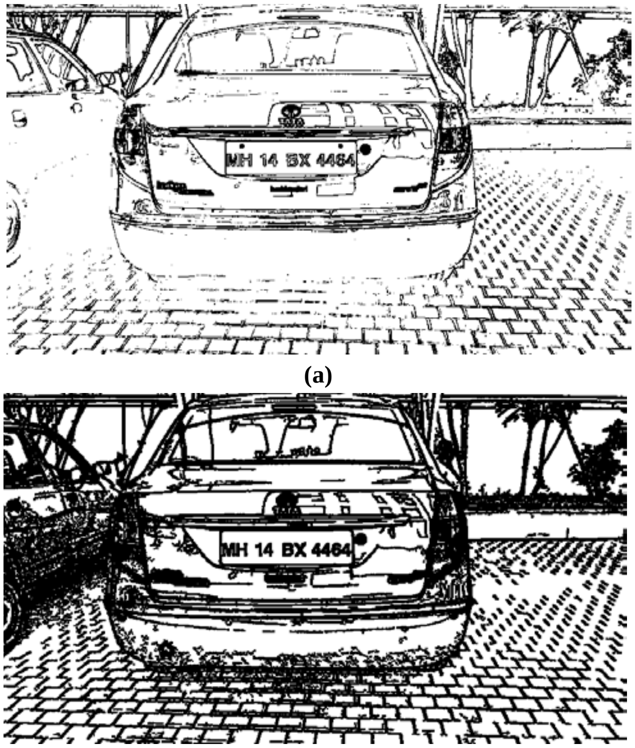
(a)

(b)

(c)

(d)

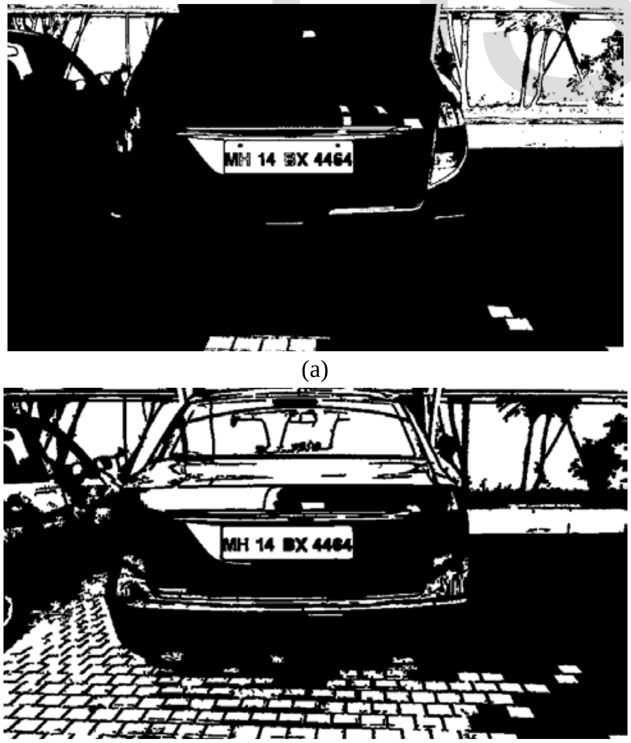
Fig. 4: (Algorithm 2) (a) Vertical edge image, (b) Horizontal edge image, (c) OR masking (a) and (b) image, (d) Dilated edge image



(a)

(b)

Fig. 5: (a) (Algorithm 1) Inverted edge image, (b) (Algorithm 2) Inverted edge image



(a)

(b)

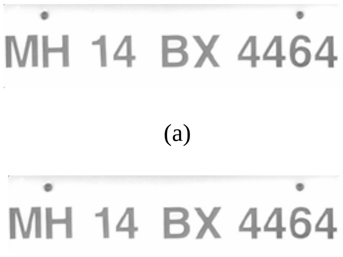
Fig. 6: (a) (Algorithm 1) Unwanted components removed, (b) (Algorithm 2) Unwanted components removed

d. License plate segmentation:

The procedure for license plate segmentation is also shared by the two algorithms. A connected components analysis is done and all the components are tested if they fulfill the requirements for them to be the probable license plate. Aspect ratio of the license plate was used for selection. Aspect ratio is the ratio of width to height of the sub-image, which can be computed using

$$\mathrm{ar} = \frac{(\mathrm{C}_{\max} - \mathrm{C}_{\min}) + 1}{(\mathrm{R}_{\max} - \mathrm{R}_{\min}) + 1} \qquad (3)$$

Where, Cmax , Cmin , Rmax and Rmin are the column and row extremities obtained by connected components analysis. The images acquired for this project have license plates with aspect ratio varying as 1.5 < ar < 6 (license plates with two rows have a lower aspect ratio than the ones with a single row). Hence, this was used as a criterion for the license plate extraction. More criteria for selection were used to improve the performance. Every sub-image selected was inverted and a connected components analysis was done. Only those sub-images that have more than 5 connected components (as there should be characters present for it to be a license plate) were considered to be a probable license plate. Finally, the width of the sub-image was to be greater than 1/15th of width of the image (assuming the size of license plate will be greater than that). Fig. 7 shows the license plate extracted by the algorithms 1 and 2 (shown in gray-scale). Now, from the probable license plate areas, the one with the highest standard deviation is selected as the license plate.



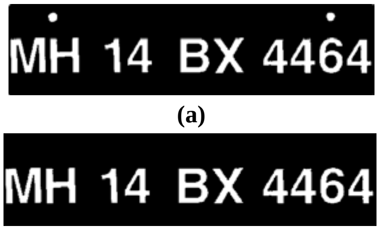
(a)

(b)

Fig. 7: Selected license plate (a) (Algorithm 1), (b) (Algorithm 2)

e. Character segmentation:

The selected license plate then undergoes a series of processing to elliminate the non character elements and segment the license plate into rows of characters and the characters itself. First, the license plate is binarized and inverted (inversion is done to make the characters white with black background since the template characters are that way) (Fig. 8(a)). Boundary elements are removed if present and small components with 100 pixels or less are removed. Average height of the components are measured and anything less than 80% of the average height is eliminated. This helps in getting rid of small designs, unrelated characters and also the bolt that is used to fix the plate (Fig. 8 (b)). This elimination procedure is common in both the algorithms.

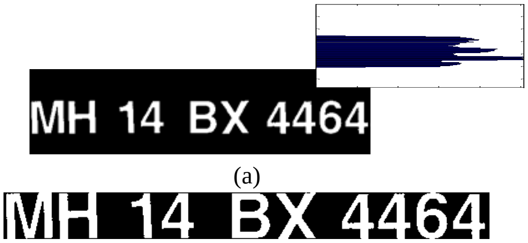


(a)

(b)

Fig. 7: (a) Binarized and inverted license plate, (b) Eliminating unwanted components

Next steps are to crop out the rows of the license plate and are common for both the algorithms. A vertical position histogram is used to find the rows of characters and crop it out as shown in Fig. 8 in both the algorithms. Same procedure is used to crop out rows if there are two rows present in the plate. When cropping the rows only one more row which is at least 1/3 the height of the tallest row and has a mean value more than 70% of mean of the histogram is considered.



(a)

(b)

Fig. 8: (a) Vertical histogram, (b) Cropped row

Next is the character segmentation. We decided to take two different approaches in algorithm 1 and 2. In Algorithm 1 bounding boxes are obtained around the characters using inbuilt function in Matlab. This returns the coordinates of the left top corner of the smallest rectangle around each of the characters along with the height and width of the rectangle, as shown in Fig. 9. With these details it is easy to crop the characters.

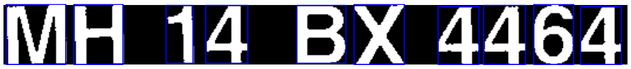
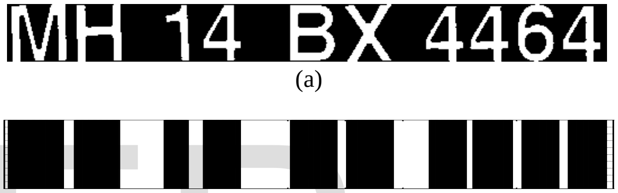


Fig. 9: Character segmentation in algorithm 1

Algorithm 2 uses the same method used for row segmentation i.e. position histogram to segment the characters. Sometimes the characters are too close to each other and might make position histogram a difficult way to segment. To help with that, line erosion is done on the license plate to thin the characters a bit. The process is shown in Fig. 10



(a)

(b)

Fig. 10: (Algorithm 2) (a) Cropped and eroded character row, (b) Horizontal position histogram for cropping characters

f. Character recognition

Here we have used correlation with the templates for recognizing characters. The template characters are of size 42x24 and hence the extracted characters are resized to 42x24 and are correlated with each of the template characters. The highest correlation value represents the recognized character.

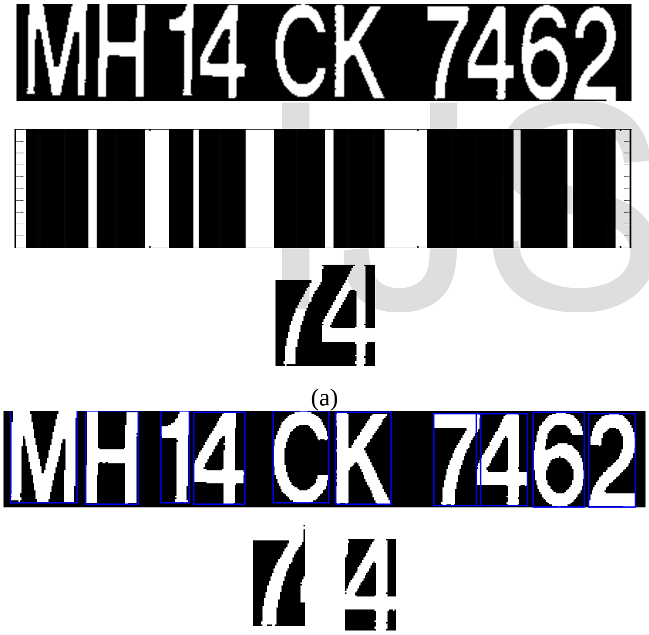
IV. RESULTS

Both the algorithms are efficient in license plate recognition and segmentation if the kinds of images are different. For example sliding concentric windows method works great if the images are of very good resolution with a good ambient lighting. But when the same set of images are given to both the algorithm, like our test images, algorithm with morphological operators has a much higher efficiency than sliding concentric windows. We have used images with varying resolutions and lighting conditions for testing.

Algorithm 1 was able to segment 90% of the license plates from the test images whereas algorithm 2 could locate and segment 80%. In both the algorithms the returned license plate number for the images used for this paper was: MH14BX4464.

V. CONCLUSION

From our study we found algorithm 1 using morphological operators was 4 times faster than algorithm 2 using sliding concentric windows and was more efficient too. After comparing the histogram method and bounding box method for character segmentation, it was found that the bounding box works better. For example, when the characters are close and overlapping each other's vertical space, histogram method fails to segment the characters (Fig. 11)



(a)

(b)

Fig. 11: (a) position histogram extracts 7 and 4 as one character (b) Bounding box successfully separates 7 and 4

VI. REFERENCES

[1]. Anagnostopoulos, C.N.E.; Anagnostopoulos, I.E.; Loumos, V.; Kayafas, E., "A License Plate-Recognition Algorithm for Intelligent Transportation System Applications," *Intelligent Transportation Systems, IEEE Transactions on* , vol.7, no.3, pp.377,392, Sept. 2006

[2]. Anagnostopoulos, C.N.E.; Anagnostopoulos, I.E.; Psoroulas, I.D.; Loumos, V.; Kayafas, E., (Sept. 2008), "License Plate Recognition from Still Images and Video Sequences: A Survey", IEEE Transactions on Intelligent Transportation Systems, vol.9, no.3, pp.377-391

[3]. Chirag Patel, Atul Patel, Dipti Shah,(2015), "A Novel Approach for Detecting Number Plate Based on Overlapping Window and Region Clustering for Indian Conditions", IJIGSP, vol.7, no.5, pp.58-65

[4]. Deb, K., Chae, H., Jo, K.. (Aug. 2009), "Vehicle License Plate Detection Method Based on Sliding Concentric Windows and Histogram", Journal of Computers, Vol 4, No 8 (2009), 771-777

[5]. Deb, K.; Hyun-Uk Chae; Kang-Hyun Jo, (April 2008), "Parallelogram and Histogram based Vehicle License Plate Detection", International Conference on Smart Manufacturing Application, 2008. ICSMA 2008, vol., no., pp.349,353, 9-11

[6]. Kanani, P.; Gupta, A.; Yadav, D.; Bodade, R.; Pachori, R.B.,(April 2013), "Vehicle license plate localization using wavelets", IEEE Conference on Information & Communication Technologies (ICT), 2013, vol., no., pp.1160,1164, 11-12

[7]. Kulkarni, P.; Khatri, A.; Banga, P.; Shah, K.,(June 2009), "A feature based approach for localization of Indian number plates", IEEE International Conference on Electro/Information Technology, 2009. eit '09, vol., no., pp.157,162, 7-9

[8]. Manisha Rathore, Saroj Kumari, (May 2014), "Tracking number plate from vehicle using Matlab", International Journal in Foundations of Computer Science & Technology (IJFCST), Vol.4, No.3

[9]. Manisha Shirvoikar, Jairam Parab, Vinay Mirashi, Ramesh Kudaskar, (May 2013),"Effective method of license plate localization and segmentation of vehicles", International Journal of Advances in Engineering & Technology Vol. 6, Issue 2, pp. 1043-1048

[10]. R.Radha1 and C.P.Sumathi, (August 2012), "A novel approach to extract text from license plate of vehicles", Signal & Image Processing : An International Journal (SIPIJ) Vol.3, No.4

[11]. Ragini Bhat, Bijender Mehandia, (August 2014), "Recognition of vehicle number plate using matlab", International jounal of innovative research in electrical, electronics, instrumentation and control engineering,Vol. 2, Issue 8, August 2014

[12]. Reza Azad, Fatemeh Davami and Babak Azad, (July 2013) "A novel and robust method for automatic license plate recognition system based on pattern recognition", CSIJ Advances in Computer Science: an International Journal, Vol. 2, Issue 3, No. 4

[13]. Sarbjit Kaur and Sukhvir Kaur,(February 2014), "An Efficient Method of Number Plate Extraction from Indian Vehicles Image", International Journal of Computer Applications 88(4):14-19

[14]. Sneha G. Patel, (February 2013), "Vehicle license plate recognition using morphology and neural network",

International Journal on Cybernetics & Informatics (IJCI) Vol.2, No.1

[15]. Ms. Sonali Sonavane Prof. Alka Khade Prof. V. B. Gaikwad, (August 2013), "Novel Approach for Localization of Indian Car Number Plate Recognition System using Support Vector Machine", International Journal of Advanced Research in Computer Science and Software Engineering, Volume 3, Issue 8

[16]. Sourav Roy, Amitava Choudhury, Joydeep Mukherjee, (March 2013),"An Approach towards Detection of Indian Number Plate from Vehicle", International Journal of Innovative Technology and Exploring Engineering (IJITEE) ISSN: 2278-3075, Volume-2, Issue-4

[17]. Sukhpreet Singh, (June 2013), "Optical Character Recognition Techniques: A Survey", Journal of Emerging Trends in Computing and Information Sciences, Vol. 4, No. 6

[18]. Warren S. McCulloch, Walter Pitts, (December 1943), "A logical calculus of the ideas immanent in nervous activity", The bulletin of mathematical biophysics, Volume 5, Issue 4, pp 115-133

[19]. http://in.mathworks.com/help/matlab