International Journal of Engineering Research and General Science Volume 3, Issue 4, Part-2, July-August, 2015
ISSN 2091-2730

APPLICATION OF VERTICAL EDGE BASED METHOD IN NUMBER PLATE EXTRACTION USING HORIZONTAL AND VERTICAL SCANNING

Silpa S Kumar, Radhakrishnan B, Raji P G
P.G Scholar, Dept of CSE, BMCE, Kollam, Kerala, India
skumarsilpa09@gmail.com

Abstract— This is a method used for car license plate detection. Firstly the Adaptive thresholding is applied for binarizing the gray scale image. Then the binarized image is enhanced by removing noise. After that, new Vertical edge based method is used to detect the number plate because of the high availability of the vertical edges in the number plate rather than the horizontal edges. Using vertical edge detection reduces the computation time. After that scan the image horizontally and vertically by pixel information are used to detect the license plate. This new Vertical edge based method to make proposed car license plate detection method faster.

Keywords— Gray scale, Binerizing, Enhance, Noise removal, Car license plate detection, Adaptive thresholding, Vertical edge based detection.

1. INTRODUCTION

The Car license plate detection and recognition system (CLPDRS) became an important area of research in image processing technology. It is mainly due to its various applications, such as the payment of parking fees, highway toll fees, traffic data collection (vehicle access control, border control) and crime prevention [1]. To identify a car, features such as model, color, and LP number can be used [2]. Among these, License plate numbers are being used to uniquely identify a car. For example in parking fee payment, number plates are used to calculate the duration of the parking. When a vehicle enters the gate, license plate is automatically recognized and stored in database. On leaving, the license plate is recognized again and compared with the stored numbers in the database. The time difference is used for calculating the parking fee.

License plates are available in various styles and colors in various countries. Every country has their own license plate format. So each country develops the CLPDRS system appropriate for the license plate format. In India the license plate containing white background with black foreground colour for private cars and for the commercial vehicle used yellow as background and black as foreground colour. The number plate start with two digit letter "state code" followed by two digit numeral followed by single letter after those four consecutive digits as the below Fig.1. From the Fig.1, 1 indicates the Country code, 2 indicate the state code, and 3 indicate the district code, 4 indicate the type of vehicle and 5 indicates the actual registration number. Locating the number plate is very challenging work in the field of image processing. The whole system mainly consists of two steps. First is to identify the position of the number plate from the particular vehicle and then segmentation of all the numbers and letters of the number plate.

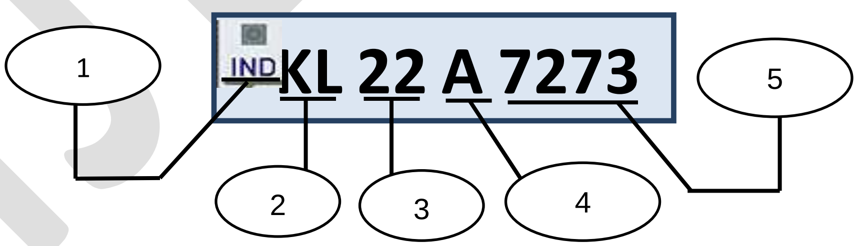

Fig.1 Sample of Indian license plate.

CLPDRS is convenient and cost efficient as it is automated. CLPDRS consists of three parts: license-plate (LP) detection (LPD), character segmentation, and character recognition. Among these, LPD is the most important part in the system because it affects the system's accuracy. LP detection is the process of finding the region or location in an image that contains the LP. To make the license plate detection process successful and fast, many difficulties must be settled, such as the poor image quality from the uneven lighting condition, plate sizes and designs, processing time, background details and complexity and various observation angles from the vehicles and cameras [3]. To detect the region of car license plate, in the past, many techniques have been used, for example the morphological operations [4],[5], edge extraction[6], combination of gradient features [7], salience features [8], a neural network for color [5] or grayscale [9] classification, and vector quantization [10].

Edges characterize boundaries are important techniques in image processing. Edges in images are areas with strong intensity contrasts. An edge in an image is a significant local change in the image intensity, usually associated with a discontinuity in the image intensity. Edges typically occur on the boundary between two different regions in an image. Edges can be modelled according to their intensity profiles. Edge detection refers to the process of identifying and locating sharp discontinuities in an image. The discontinuities are abrupt changes in pixel intensity which characterize boundaries of objects in a scene. So, an edge map has vastly reduced complexity, and it retains the important structure present in the original image.

Vertical edge detection is one of the most crucial steps in CLPD because it influences the whole system to correctly detect the LP [11]. Vertical edge-based CLPD is the most suitable technique to overcome some of the difficulties in the license plate detection, because of the abundance availability of vertical edges. Car license plate details always have a lot of vertical edges that can be used to generate the candidate regions for classification [12]. Historically, the most common and earliest edge detection algorithms are those based on the gradient, such as the Sobel operator [13] and the Roberts operator [3]. Sobel operator is one of the most important algorithms that has been widely used to detect the vertical edges in many CLPD methods [13].

This paper aim to propose a new and fast technique for detecting the vertical edge. It is faster than Sobel operator with an accurate performance of showing the vertical edges. Furthermore, the use of Vertical Edge based Method in CLPD would enhance the system performance. In this car-license-plate detection, the web camera is used to capture the images, and an offline process is performed to detect the plate from the whole scene image. This color input image is converted to a grayscale image, and then, Adaptive thresholding is applied on the image to constitute the binarized image. Thresholding is computed using integral image.

After thresholding, the image will only have black and white regions, and the vertical edges are extracted from these regions. Here a vertical edge detection is used to reduce the computation time of the whole car-license-plate detection method. The idea of the vertical edge detection concentrates on intersections of black–white and white–black in the black and white image. The next process is to detect the license-plate; the plate details are Scanned based on the pixel value with the help of the vertical edge detection. The image is scanned horizontally to detect candidate regions and vertically to search for the true candidate region. Finally, the true plate region is detected in the original image.

The proposed system for car license plate detection is shown in Fig.2.

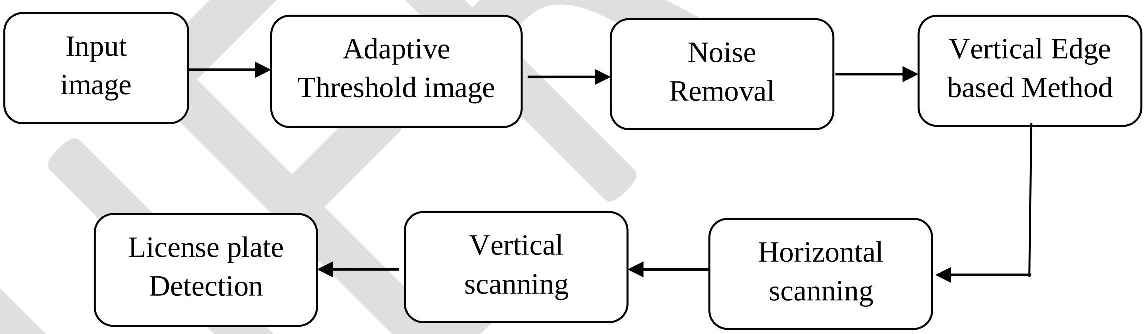

Fig.2 Block diagram of proposed system for car license plate detection.

Section II will discuss the procedures of this car license plate detection in detail. Section III draw our conclusion.

2. PROPOSED SYSTEM

Firstly an image of the car-license-plate is captured by the web camera. And this color input image is converted to a grayscale image, which enhance the speed of the car license plate detection method.

A. Adaptive Thresholding Process

This Adaptive Thresholding process is just a simple extension of Bradly and Roth"s [14] and Wellner's methods [15]. The Wellner's method of quick adaptive thresholding is that pixel is compared with an approximate moving average of last S pixels, is calculated while passing through the image. If the value of the current pixel is T percent lower than the average, then it is set to white: otherwise it is set to black. Wellner uses S equal to one eighth of the image width, T equal to 0.15 to yield the best results for a verity of image. T should be the range $0.1<T<0.2$ in our method. The advantage of this method is the single pass through the image is required. But Wellner"s algorithm depends on the scanning order of the pixels. Since the moving average is not suitable for the good representation because the neighborhoods are not evenly distributed in all direction. So the Bradley's method used to solve the problem by using integral image. It uses the mean of local window, where local mean is computed using integral image.

Step 1: To compute integral image I(x, y) from the gray scale image g(x, y). To store each location of I(x, y) the sum of all g(x, y) terms to the left and above the pixel (x, y). Here linear time used in the following equation for each pixel

$$\mathrm{I(x,y) = g(x,y) + I(x-1,y) + I(x,y-1) - I(x-1,y-1)} \tag{1}$$

Step 2: The intensity summation for each local window should be computed by using two subtrac6ion and one addition operation. The window with upper left corner (x-(s/2), y-(s/2)) and lower right corner (x+(s/2), y+(s/2)) can be computed in constant time using the following equation, s represent the local window size or length for the computed integral image, whereas s=image width/8.

$$\mathrm{sum}_{window} = \left(I\left(x+\frac{s}{2}, y+\frac{s}{2}\right)\right) - \left(I\left(x+\frac{s}{2}, y-\frac{s}{2}\right)\right) - \left(I\left(x-\frac{s}{2}, y+\frac{s}{2}\right)\right) + \left(I\left(x-\frac{s}{2}, y-\frac{s}{2}\right)\right) \tag{2}$$

Step 3: Perform threshold for each pixel in an image. The gray scale image, g(x, y) has the values between [0-255].

$$AT(x,y) = \begin{cases} 255, & g(x,y) \times S^2 < (1-T) \times \mathrm{sum}_{window} \\ 0, & otherwise \end{cases} \tag{3}$$

where AT(x, y) represents the adaptive threshold value of the pixel g(x, y),and S^2 represents the computed area of the local window for the selected region. The speed of this adaptive thresholding is increased by using mean and variance of local windows [16].

B. Noise Removal

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. Only black pixel values in the threshold image are tested. Once, the current pixel value located at the mask center is black, the eight-neighbor pixel values are tested. If two corresponding pixel values are white together, then the current pixel is converted to a white value as a foreground pixel value (i.e., white pixel) based on four cases. In the first case, the pixels are horizontal with an angle equal to 0◦ as (−). In the second case, the pixels are vertical with an angle equal to 90◦ as (|). In the third case, the pixels are inclined with an angle equal to 45◦ as (/). In the fourth case, the pixels are inclined with an angle equal to 135◦ as (\). This step is enhance the binarized image.

C. Vertical Edge Based Method

After thresholding is applied, the character region appears white and background as black. This Vertical edge based method processing these regions efficiently. It distinguish the plate details region of image particularly beginning and end of each character the plate. Therefore using this Vertical edge based method will easily detect the plate details, and the character recognition process will done faster. The idea of the Vertical edge based method concentrates on black and white pixel regions.

	0	1	2
0	x-1, y-1	x-1, y	x-1, y+1
1	x, y-1	x, y	x, y+1
2	x+1, y-1	x+1, y	x+1, y+1

Fig.3 Mask of proposed system.

A 3×3 mask is used as shown in figure 3. The center pixels located at the positions are (0, 1), (1, 1), (2, 1). The mask is moved from left to right, if combination of black and white pixels appears in the mask then no change in center pixel value, but if the mask have all black pixels, then it changes to white. The proposed mask has the size of 3×3. If the three pixels at location (0, 1), (1, 1) and (2, 1) are black, then the other two columns are tested and if both or any are black, the three center pixels will be converted to white. Otherwise the second column (center pixels values) will not change. This process is repeated for the whole image. Here the three rows are checked at once. Therefore consumed time in this case is reduced.

D. Horizontal Scanning

From the Vertical Edge based Method we got a lot of vertical edges .The image is scanned from top to bottom and from left to right. This process is divided into 2 steps:

1. Count Black Pixel Availability Per Each Row: The number of black pixels in each rows are counted and stored in an array variable CountBlkPix[a], Where [a=0,1,2,3,……..height-1].
2. Divide The Image into Candidate Regions: Candidate regions are selected based on the count of black pixel values from previous step. The threshold values are used to select the rows as candidate region or not. If the CountBlkPix array value is greater than the threshold, that rows are present in candidate region.

E. Vertical Scanning

This process aims to select and extract LP from the candidate regions. The process is divided into two parts:

1. Selection Process of the LP Region:

For the candidate regions, each column will be checked one by one. If the column blackness ratio exceeds 50%, then the current column belongs to the LP region; thus, this column will be replaced by a vertical black line in the result image. Hence, each column is checked by the condition that, if BlckPix ≥ 0.5 ×colmnHght, then the current column is an element of the LP region. Here, the "BlckPix" represents the total number of black pixels per each column in the current candidate region, and the "colmnHght" represents the column height of the of the candidate region. This condition with a fixed value (0.5) is used with nonblurry images. However, some pixels of the candidate regions will not be detected in case the ratio of blackness to the total length (height) of the candidate region is greater than 50%. Therefore, the condition is changed to be less than 50%, according to the ratio of the blurry level or the deformation of the LP. The condition will be modified and store in P_{RS} . The BlckPix ≥ P_{RS} × colmnHght, where P_{RS} represents the P_{RS} factor. TheP_{RS} value is reduced when the blurry level is high to highlight more important details, and it is increased when the blurry level is less. Therefore, the mathematical representation for selecting the LP region can be formulated as follows:

$$C_{region} = \begin{cases} 0, & \text{BlckPix} \geq P_{RS} \times \text{colmnHght} \\ 255, & \text{otherwise} \end{cases} \tag{4}$$

where C_{region} represents the output value for the current pixel of the currently processed candidate region. If C_{region} = 0, consider the checked pixel as an element of the LP region; otherwise, consider it as background. The value of P_{RS} is automatically determined.

2. Making a Vote:

The columns whose top and bottom neighbors have high ratios of blackness details are given one vote. This process is done for all candidate regions. Hence, the candidate region that has the highest vote values will be the selected region as the true LP. By tracking the black vertical lines, the plate area will be detected and extracted.

3. CONCLUSION

This paper present a new and fast algorithm for detecting vertical edges, in which its performance is faster than the other existing edge detection operators. Before applying the method the image is binarized using adaptive thresholding, which makes the license plate clear. The advantages of this method is that it concentrates black pixel inside the plate and have the ability to work with blurred images for detecting vertical edges. This advantage helps to detect license plate details easily and faster than other existing methods.

REFERENCES:

[1] S. N. Huda, K. Marzuki, Y. Rubiyah, and O. Khairuddin, "Comparison of feature extractors in license plate recognition," in *Proc. 1st IEEE AMS*, Phuket, Thailand, 2007, pp. 502–506.

[2] E. R. Lee, K. K. Pyeoung, and J. K. Hang, "Automatic recognition of a car license plate using color image processing," in *Proc. IEEE Int. Conf.Image Process.*, 1994, pp. 301–305.

[3] H. Bai and C. Liu, "A hybrid license plate extraction method based on edge statistics and morphology," in *Proc. 17th Int. Conf. Pattern Recognit.*, Cambridge, U.K., 2004, pp. 831–834.

[4] J.-W. Hsieh, S.-H. Yu, and Y. S. Chen, "Morphology-based license plate detection from complex scenes," in *Proc. 16th Int. Conf. Pattern Recognit.*, Quebec City, QC, Canada, 2002, pp. 176–179.

[5] A. A. Lensky, J. Kang-Hyun, and V. V. Gubarev, "Vehicle license plate detection using local fractal dimension and morphological analysis," in *Proc. IEEE 1st Int. Forum Strategic Technol.*, Ulsan, Korea, 2006, pp. 47–50.38 IEEE Transactions On Vehicular Technology, Vol. 62, No. 1, January 2013.

[6] H. Zhang, W. Jia, X. He, and Q. Wu, "A fast algorithm for license plate detection in various conditions," in *Proc. IEEE Int. Conf. Syst., Man, Cybern.*, Taipei, Taiwan, 2006, pp. 2420–2425.

[7] S. Kim, D. Kim, Y. Ryu, and G. Kim, "A robust license-plate extraction method under complex image conditions," in *Proc. 16th Int. Conf. Pattern Recognit.*, Quebec City, QC, Canada, 2002, pp. 216–219.

[8] Z.-X. Chen, Y.-L. Cheng, F.-L. Chang, and G.-Y. Wang, "Automatic license-plate location and recognition based on feature salience," *IEEE Trans. Veh. Technol.*, vol. 58, no. 7, pp. 3781–3785, Sep. 2009.

[9] H. Caner, H. S. Gecim, and A. Z. Alkar, "Efficient embedded neuralnetwork- based license plate recognition system," *IEEE Trans. Veh. Technol.*, vol. 57, no. 5, pp. 2675–2683, Sep. 2008.

[10] S. Rovetta and R. Zunino, "License-plate localization by using vector quantization," in *Proc. Int. Conf. Acous., Speech, Signal Process.*, 1999, pp. 1113–1116.

[11] W. Jia, H. Zhang, and X. He, "Region-based license plate detection," *J. Netw. Comput. Appl.*, vol. 30, no. 4, pp. 1324–1333, Nov. 2007.

[12] H. Bai, J. Zhu, and C. Liu, "A fast license plate extraction method on complex background," in *Proc. IEEE Int. Conf. Intell. Transp. Syst.*, 2003, pp. 985–987.

[13] S. Thanongsak and C. Kosin, "The recognition of car license plate for automatic parking system," in *Proc. 5th Int. Symp. Signal Process. Appl.*, Brisbane, QLD, Australia, 1999, pp. 455–457.

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

[15] P. D. Wellner, "Adaptive thresholding for the DigitalDesk," Rank Xerox Ltd., Birmingham, U.K., Tech. Rep. EPC-1993-110, 1993.

[16] F. Shafait, D. Keysers, and T. M. Breuel, "Efficient implementation of local adaptive thresholding techniques using integral images," in *Proc. Doc. Recognit. Retriev.*, 2007, pp. 681510-1–681510-6.