iJRASET
International Journal For Research in
Applied Science and Engineering Technology

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 2 **Issue:** V **Month of publication:** May 2014

DOI:

ISSN: 2321-9653

INTERNATIONAL JOURNAL FOR RESEARCH IN APPLIED SCIENCE AND ENGINEERING TECHNOLOGY (IJRASET)

A Review of Automatic License Plate Detection Using Edge Detection Methods

THOMAS K T*, VAIJAYANTHIMALA J**

*Student of ME Computer Science & Engineering, Mahendra Engineering College, Namakkal, TN

**Assistant Professor, Department of computer science and engineering, Mahendra Engineering College, Namakkal, TN

Abstract—Vehicle plate recognition is a valuable image processing system employed for detecting and recognizing license plates thereby categorizing and identifying vehicles. Of late several researchers around the globe have initiated exploring license plate recognition using various methods. ALPR-Automatic License Plate Recognition systems have become a central part in various security and traffic management systems across various countries. Automatic toll management, Speed control and access control etc are some areas where ALPR is found very successful. In this paper we are comparing three popular methods suggested for vehicle license plate detection (LPD algorithms).The key focal point of these schemes is on proposing a non complex yet robust solution to get to the bottom of the common issues in license plate recognition. Edge detection methods are popularly used in license plate detection algorithms. In this paper we are presenting a review of mechanisms using Canny operator, Sobel Operator and a comparatively new algorithm called VEDA- Vertical Edge detection Algorithm.

Index Terms- Image Processing, Edge Detection, License Plate Detection, License Plate Recognition.

I. Introduction

License plate Detection (LPD) is an important function in many areas such as traffic control, Toll Processing, Crime investigation, parking management etcetera. A number of algorithms are now available for LPD some of them show very high accuracy than others. Some methods use computationally intensive algorithms. One of the difficult task the industry faces is the selection of one of them based on trade off over execution time, optimum memory usage, Algorithm complexity and accuracy in various environmental conditions etc.

In this paper we are trying to study some of the popular LPD algorithms and compare them in terms of execution time and accuracy

The LPR is used mostly in real-time systems; A good system shall provide accuracy as well as acceptable response time . The majority of the LPR systems are based on image processing methods and character recognition techniques [1, 2,]. The LPR systems generally consists of three fundamental sections namely, image acquisition, License Plate Detection (LPD), and Optical Character Recognition (OCR). The image acquisition system uses a camera with sufficiently high shutter speed to capture an image of vehicle part which contains the license plate. The higher the shutter speeds the lesser will be the motion blur.

The LPD portion of the system analyzes the captured image to localize the license plate by finding where the alphanumeric characters are. Certain algorithms locate the license plate based on features such as the shape, color, height-to-width ratio etc. The performance of these algorithms depends heavily up on changes in environmental situations such as lighting condition or current weather scenario that affect the

ISSN: 2321-9653

INTERNATIONAL JOURNAL FOR RESEARCH IN APPLIED SCIENCE AND ENGINEERING TECHNOLOGY (IJRASET)

quality of image features. The final part of the system segments the characters and uses an OCR program to recognize the segmented characters that is present on the plate area [2].

We should note that Lighting conditions such as cloudy weather, night working hours, reflecting sunlight or car's taillight or front light, Complex background, Damaged or dirty license plates, Varying camera angles and changing distance between camera and the object etc may affect the success of the method used.

This paper is organized as follows. In section II we discuss the general procedures we follow in edge detection based algorithms. In section III we discuss about the Canny operator, which is one of the most popular edge detection operator and go on to analyze it against the comparatively newer VEDA in section IV and finally, conclusions are drawn in Section V.

II. Overview of LPR using Edge Detection

The common steps for almost all the edge based license plate detection are as given below:

1) Acquisition of image
2) Image preprocessing
3) Morphological operations if any
4) License Plate Detection
5) License Plate Extraction
6) Character Recognition

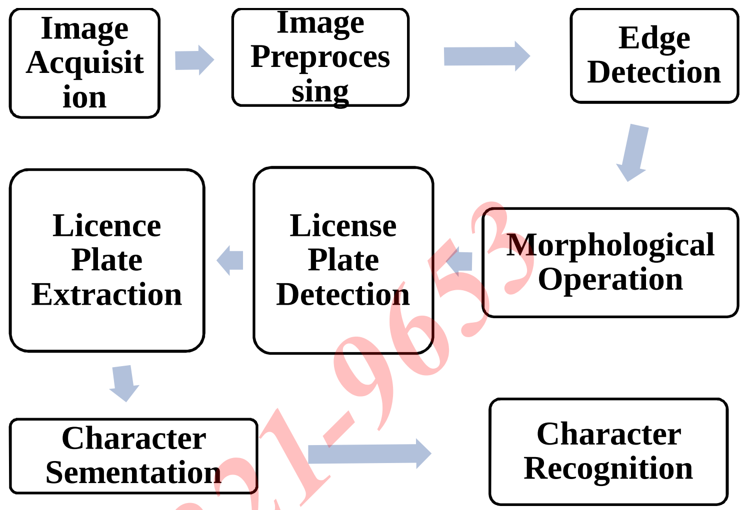

Fig.1 Generic Steps in Image Processing as part of LPR

The third step given in the above figure is the point of focus of this paper .This can be carried out using popular operators like Canny or Sobel or comparatively novel algorithms like VEDA.

III. Using the Canny Operator for LPR

Edge detection is a basic tool which has widespread us in processing images. It is applied in a variety of applications such as determining the object which could detect and identify certain objects of an image very clearly.

A lot of edge-detection methods are widely used based on several possible optimization mechanisms. As an example,error minimization, fuzzy logic, morphology, genetic algorithms, neural network and Bayesian approach. A number of edge detection methods perform to wavering degrees of quality within altered conditions. Therefore, it is advisable to apply multiple edge-detection algorithm.

The Canny Edge-Detector is an operator which uses a multistage algorithm to find out a wide range of edges images that contain a lot of noice, as follows

Vol. 2 Issue V, May 2014

ISSN: 2321-9653

INTERNATIONAL JOURNAL FOR RESEARCH IN APPLIED SCIENCE AND ENGINEERING TECHNOLOGY (IJRASET)

1. Use the Gaussian filter $G\sigma\,(m,n)$ to smooth out the image $f(m,n)$. This will reduce noise or unwanted details and textures as given in Eq.1

$$g(m,n) = G\sigma\,(m,n) * f(m,n) \qquad (1)$$

Where $G\sigma\,(m,n)$ is given as

$$G\sigma\,(m,n)=\frac{1}{\sqrt{2\pi\sigma^2}}exp[-\frac{m^2+n^2}{2\sigma^2}] \qquad (2)$$

2. Compute gradient of $g(m,n)$ using any of the gradient operators to reach:

$$M(m,n)=\sqrt{g_m^2(m,n)+g_n^2(m,n)} \qquad (3)$$

and

$$\theta(m,n) = tan^{-1}[g_n(m,n)/g_m(m,n)] \qquad (4)$$

3. Threshold M is given as:

$$M_T(m,n)=\begin{cases} M(m,n) & if\ M(m,n) > T \\ 0 & otherwise \end{cases} \qquad (5)$$

Where T is chosen, such that all edge elements are kept, most of the noise is suppressed.

4. Suppress non-maxima pixels in the edges of M_T obtained above to thin the edge ridges. To do so, check to see whether each non-zero $M_T(m,n)$ is greater than its two neighbors along the gradient direction $\theta(m,n)$. If so, keep $M_T(m,n)$ unchanged, otherwise, set it to zero.
5. Threshold the previous result by two different thresholds τ_1 and τ_2 (where $\tau_1 < \tau_2$) to obtain two binary images T_1 and T_2. This T_2 has less noise and fewer false edges than T_1 but it also has larger gaps between edge segments.
6. Link edges segments in T_2 to form continuous edges. To do so, trace each segment in T_2 to its end and then search its neighbors in T_1 to find any edge segment in T_1 to bridge the gap until reaching another edge segment in T_2.

THE ALGORITHM FOR DETECTING THE LP

The main objective of this study is to explore the efficiency of capturing and recognizing plate numbers of vehicles using a suitably fixed camera. The image would then be kept for further processing or may be compared against specific databases, such that if a plate number is matched to one already existing, then the system is enabled. The algorithm is based on clear and simple image processing tools and is summarized as follows:

1. Scanning and re dimensioning of the acquired image

2. Converting the image to a gray scale image using suitable algorithm

3. Obtain the complement of the image and deneterm the edges

4. Split the image into objects

6. Produce the recognized LP as output.

Every step of the algorithm procedure has its responsibility, priority, towards having a quality algorithm. This causes a better plate recognition software application. Scanning the overall image is an initial point for constructing impression of the plate. Dimensioning image to match a systematic analysis approach is formed. Several edge-detection approaches may be used to obtain the plate of a vehicle [2,3,4].

So using suitable filters, unwanted items are easily picked and essential objects of the image are retained. All of these are split in a way which a plate is identified, to its appropriate place.

IV. VERTICAL EDGE DETECTION These ALGORITHM

Abbas M. Al-Ghaili et al. suggests a new algorithm named VEDA- Vertical Edge Detection Algorithm. In their paper[] titled Vertical Edges Based Car License Plate Detection Method, they suggests the use of the new algorithm together with Adaptive Thresholding and ULEA- Unwanted Line Elimination Algorithm.

Adaptive Thresholding

This technique used is just a simple extension of Bradley's [5] and Wellner's methods [7]. The idea in Wellner's algorithm is that the pixel is compared to an average of neighboring pixels. Specifically, an approximate moving average of the last S pixels seen is calculated while reading the photograph. If the value of the current pixel is Th 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 contrasting rays and leaves out soft gradient disparities. The major advantage of this 3 technique is that only a single pass through the imimage file suffice. Wellner uses 1/8th of the image width for the value of S and 0.15 for the value of Th in order to yield the best results for

INTERNATIONAL JOURNAL FOR RESEARCH IN APPLIED SCIENCE AND ENGINEERING TECHNOLOGY (IJRASET)

a variety of images. The value of That may be slightly modified from the proposed value by Wellner depending on the used images whereas it should be in the range 0.1 < Th ,< 0.2: in the new 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. So, using the integral image in [5] has solved this problem

Vertical Edge Detection based license plate recognition

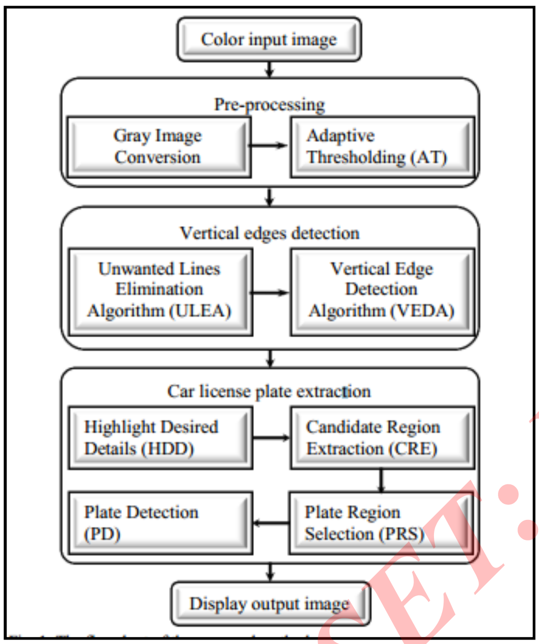

Fig.2 *Overall flow of processes in IPR using VEDA*

The overall steps in the VEDA is given in the flow chart given in fig 2.

Unwanted Line Elimiination Algorithm

After binarizing the image using Adaptive Thresholding algorithm. ULEA is applied on the binarized image. The masks given in Fig 3 tell us how the eliminatable pixels are determined

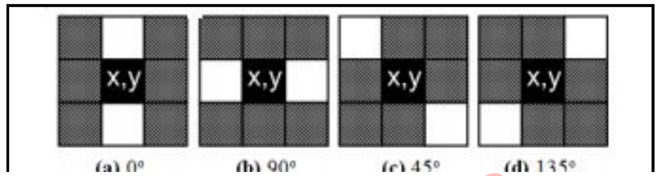

Fig. *3. The 4-cases for converting center pixel to background. (a) Horizontal. (b) Vertical. (c) Right-inclined. (d) Left-inclined*

Vertical Edge Detection Algorithm

The m aiadvantage of VEDA is to distinguish number plate details region, especially, the beginning and the end of each character. Therefore, the plate details will be easily detected and character recognition process will be done faster. After thresholding and ULEA processes, the image will only have black and white regions, and VEDA is processing these regions. The idea of VEDA concentrates on intersections of black-white. See figures. A mask 2×4 is proposed for this process as shown in Fig.4, where x and y represent rows and columns of the image. The center pixel of the mask is located at point (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 case of white-black regions will be kept.

x , y-1	**x, y**	**x, y+1**	**x, y+2**
x+1, y-1	**x+1, y**	**x+1, y+ 1**	**x+1, y+2**

Fig 4. The design of the mask

The mask should have the size of 2×4 which shall fulfill a couple of criteria.

1)In this sort of mask, it is split into three sub-masks: the first one is the left mask '2×2', the second one is the center '2×1', and the third one is the right mask '2×1', as marked in Fig. 4. Simply, after each two pixels are checked at once, the first sub-mask is applied; so that 2-pixel width 'because two column are processed' can be considered for detecting. This process is specified to detect the vertical edges at the intersection of black and white regions. Likewise the third sub-mask is affected on the intersections of white-black regions. Thus, the detected vertical edge has the property of 1-pixel width.

2) The number '2' sorts out the quantity of rows that are inspected at once. The used time in this situation m be twice lesser in case each row is singly checked. The selection of the column at the locations (0,1) and (1,1) to be a center of the proposed mask, is to retain two pixels and one pixel in case of black-white and white-black regions, respectively.

This is carried out equally to both of the edges, left and right sides of the object. The 2 × 4 mask starts reading from top to bottom and from left to right. If the four pixels at locations (0,1), (0,2), (1,1), and (1,2) are black , then the other mask values are invetstigated to check if they are black or not. If all the values are black, then the two positions at (0,1) and (1,1) should be converted to white. Otherwise, if the column 1 and any other column have dissimilar values, the pixel worth of the column 1 will then be recorded. These procedures are done again with the whole pixels in the photograph file.

V. CONCLUSION

The new algorithm suggested by Abbas M. Al-Ghaili et al a new and is fast algorithm for vertical edge detection (VEDA) which gives good performance compared to Sobel and Canny operators.

Though it gives lesser accuracy compared to canny operator in low resolutions, its overall performance is good. Dataset captured using a webcam can also be applied upon by VEDA and will give more than 90% accurate output.

VI. ACKNOWLEDGMENT

Authors wish to acknowledge the support provided by the members of department of computer science and engineering, Mahendra Engineering College, Namakkal, TN.

REFERENCES

.

[1] Liao Yu, Research on Edge Detecton in License Plate Recognition, *The 2nd International Conference on Computern Application and System Modeling (2012)*

[2] Hamid Mahini, Shohreh Kasaei, An Efficient Features–Based License Plate Localization Method 0-7695-2521-0/06/$20.00 (c) 2006 IEEE

[3] Bharat Raju Dandu, Abhinav Chopra, "Vehicular Number Plate Recognition Using Edge Detection and Characteristic Analyisis of National Number Plates", International Journal Of Computational Engineering Research/ ISSN: 2250–300

[4] Serkan Ozbay, and Ergun Ercelebi, Automatic Vehicle Identification by Plate Recognition., World Academy of Science, Engineering and Technology 9 2007

[5] D. Bradley and G. Roth, "Adaptive Thresholding using the Integral Image," Journal of Graphics Tools, vol. 12, pp. 13–21, 2007.

[6] F. Shafait, et al., "Efficient Implementation of Local Adaptive Thresholding Techniques Using Integral Images," in Document Recognition and Retrieval, USA, 2007.

[7] P. D. Wellner, "Adaptive Thresholding for the DigitalDesk," Technical Report EPC-1993-110, 1993.

[8] Abbas M. Al-Ghaili, Syamsiah Mashohor, Abdul Rahman Ramli and Alyani Ismail ,Vertical Edges Based Car License Plate Detecton Method, IEEE January 2013.

AUTHORS

First Author – Vaijayanthimala J, ME, Mahendra engineering college, Namakkaal, TN

Second Author – Thomas K T, MCA,

Correspondence Author – Thomas K. T. k.t.thomas76@gmail.com , ktthomas@india.com