ISSN:0975-9646

A Robust Automated Process for Vehicle Number Plate Recognition

Dr. Khalid Nazim S. A.[#1], Mr. Adarsh N.[#2]

[#1] *Professor & Head, Department of CS&E, VVIET, Mysore, Karnataka, India.*
[#2] *Department of CS&E, VVIET, Mysore, Karnataka, India.*

***Abstract*— A moving/parked car embeds mainly speed and momentum, and is surrounded by its environment. Both the cases are extremely unpredictable so, dynamically both the situations are handled. Any image captured is processed with edge detecting algorithm to recognize the number plate and simultaneously solve the problem of redundancy and speed issues in processing the image keeping visual quality.**

***Keywords*— Edge detection, Filtering License Plate Localization, Masking, Morphological closing.**

I. Introduction

A moving car embeds mainly speed and momentum, while a parking car is surrounded by its environment. Both the cases are extremely unpredictable so, dynamically both the situations are handled. Any image captured is processed with edge detection algorithm in recognizing the number plate and finally the recognized licensed plate is automated. In the case of moving car the clarity for the image and the time complexity in recognizing the exact number plate are the major tasks. In the case of parked car extraction of the number plate itself becomes a herculean task as the car may be parked in different backgrounds[3][7].

In processing the image the number plate is extracted along with its co-ordinates. Depending on the recognized number plate the number plate is automated by sending the contents of the detected number plate automatically to the information center.

Once the number plate is sent to the information center it can be cross verified with the constraints provided by the Regional Transport Office and get the details of the vehicle owner through the recognized number plate number.

II. Related Works

An easy way to comply with the conference paper formatting requirements is to use this document as a template and simply type your text into it. Java ANPR is a recognition techniques that has been developed earlier and has been widely used in various traffic and security applications, such as parking, access and border control, tracking of stolen cars [6].

In parking, number plates are used to calculate duration of the parking. ANPR systems can be used in access control. For example as shown in Fig. 1, this technology is used in many companies to grant access only to authorized vehicles. In some countries, ANPR systems are installed on country borders to automatically detect and monitor the border crossings. Each vehicle can be registered in a central database and compared to black list of stolen vehicles.in traffic control, vehicles can be directed to different lanes for a better congestion control in busy urban communications during the rush hours[2][4].

III. Constraints Of Anpr

ANPR though is one the effective image processing techniques. However, there are a number of flows which will take away the accuracy factor of theoretical concepts when implemented practically. This system is confined only to times new roman font. But, the number plates today are available in different font styles. The system totally fails when the car has a complex background having alpha numerical characters. The image resolution has to be high and the blurriness' in the image should be low for effective car plate recognition which is quite not a regular case when we practically implement.

The flows in ANPR system are as follows:

- Identifies only the characters of specific font style.
- Fails to recognize plate with poor image resolution.
- Fails to segment the car plate with alphanumeric background.
- Inefficient in recognizing blur images.

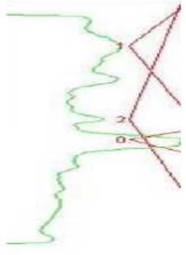
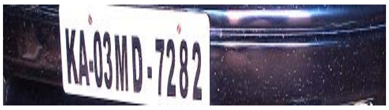
	
Car.jpg	Width: 809 X 486px
	
Detected Band	Width 728 X 436 px

Fig. 1:Image processing by Java ANPR

IV. Proposed System

In our paper we propose the basic image processing technique which involves capturing the image and converting it into gray scale image as shown in Fig. 2 followed by focusing on variety of precise filters and edge detection methodologies.

Recognizing the ROI with at most accuracy and further processing the image to segment and finally conclude in extracting the car number plate while keeping the image quality issues has been a prime concern.

A. Plate Localization module

In the first step, the features of characters are used to find the probable characters locations. In the second step, the features of number plates are used to find the probable number plate locations. This approach enables the localization of number plates in widely varying illumination conditions with relevance to the commonly found types of Indian number plates with a success rate of 87%.

1) Sobel Edge Detection

For a (3×3) neighbourhood, each simple central gradient estimate is a vector sum of a pair of orthogonal vectors. Each orthogonal vector is a directional derivative estimate multiplied by a unit vector specifying the derivative's direction. The vector sum of these simple gradient estimates amounts to a vector sum of the 8 directional derivative vectors. Thus a point on Cartesian grid and its eight neighbouring density values as shown below:

a	b	c
d	e	f
g	h	i

The directional derivative estimate vector *G* was defined such as a density difference /distance to the neighbour. This vector is determined such that the direction of *G* will be given by the unit vector to the approximate neighbour [1]. Note that, the neighbours group into antipodal pairs: (a,i), (b,h), (c,g), (f,d). The vector sum for this gradient estimate is as given below:

$$G = \frac{(c-g)}{R} \cdot \frac{[1,1]}{R} + \frac{(a-i)}{R} \cdot \frac{[-1,1]}{R} + (b-h) \cdot [0,1] + (f-d) \cdot [1,0] \qquad - (1)$$

where, $R = \sqrt{2}$ This vector is obtained as

$$G = \frac{(c-g-a+i)}{2} + f - d, \frac{(c-g+a-i)}{2} + b - h \qquad - (2)$$

:

Here, this vector is multiplied by 2 because of replacing the divide by 2. The resultant formula of this vector is as given as follows:

$$G' = 2G = [(c-g-a+i) + 2(f-d), (c-g+a-i) + 2(b-h)] \qquad - (3)$$

-1	0	+1
-2	0	+2
-1	0	+1

Gx

+1	+2	+1
-2	0	+2
-1	0	+1

Gy

The operator consists of a pair of 3×3 convolution masks as shown above. These masks are designed to respond maximally to edges running vertically and horizontally relative to the pixel grid as shown in Fig. 4 and Fig. 5.

Erosion is one of two fundamental operations in morphological image processing from which all other morphological operations are based. It was originally defined for binary images, later has been extended to images, and subsequently to complete lattices [11][12].

In grayscale morphology, images are functions mapping a Euclidean space or grid E into $R \cup \{\infty, -\infty\}$ wher R is the set of reals, ∞ is an element larger than any real number, and -∞ is an element smaller than any real number.

Denoting an image by *f(x)* and the grayscale structuring element by *b(x)*, where B is the space that b(x) is defined, the grayscale erosion of *f* by *b* is given by:

$$(f \ominus b)(x) = \inf_{y \in B}[f(y) - b(y-x)], \qquad - (4)$$

where "inf" denotes the infimum.

The input image is as shown in Figure 2

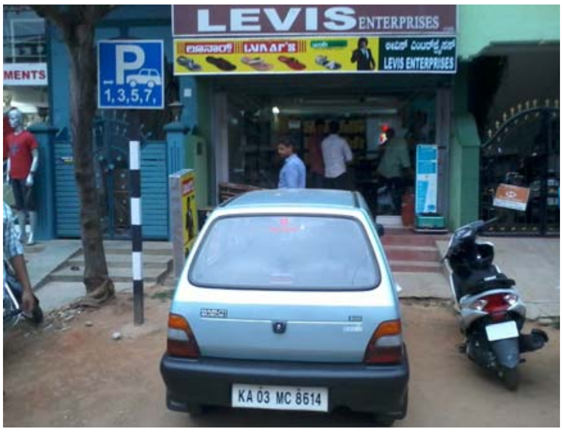


Fig. 2: Car Image with a complex background

This source image to be converted into gray scale image as shown in Fig. 3**.**

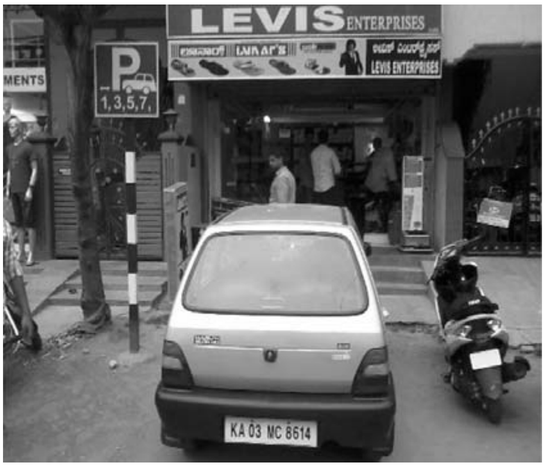


Fig. 3: Grayscale Image

This grayscale image can be processed with filtering and edge detection methodology of sobel operation is shown in Fig. 4 and Fig. 5.

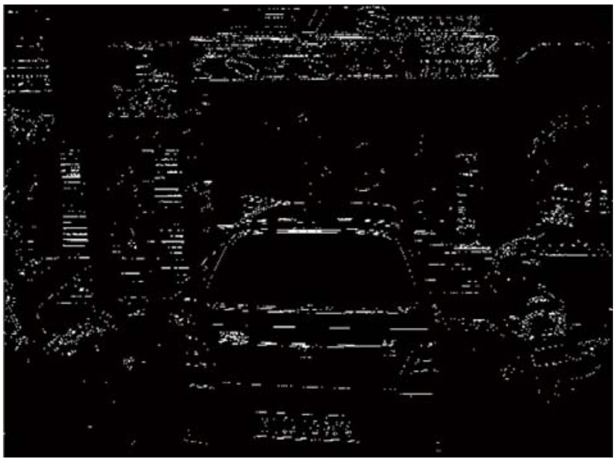


Fig. 4: Horizontal edge detection

The width of the horizontally oriented rank filter matrix is much larger than the height of the matrix ($w \gg h$), and vice versa for the vertical rank filter ($w \ll h$).

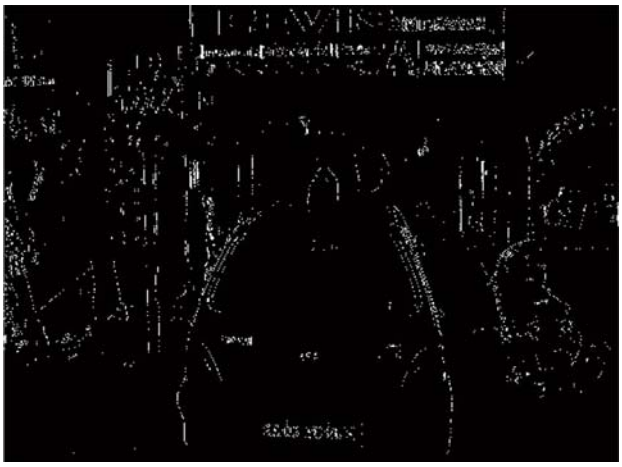


Fig. 5: Vertical edge detection

A mask is a black and white image of the same dimensions as the original image (or the region of interest you are working on) Fig. 6. Each of the pixels in the mask can have therefore a value of 0 (black) or 1 (white).

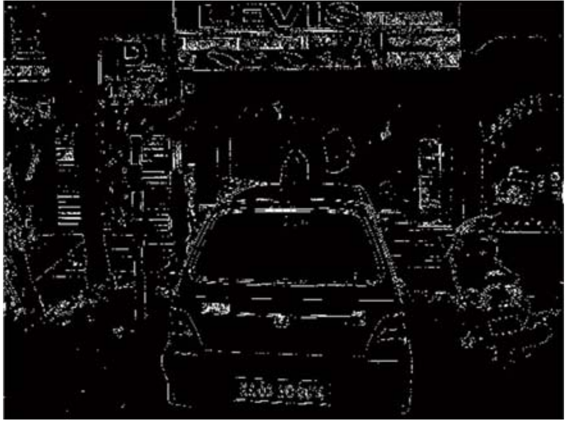


Fig. 6: Result of masking.

A binary image is viewed in mathematical morphology as a subset of a Euclidean space $\boldsymbol{R}^d$ or the integer grid $\boldsymbol{Z}^d$, for some dimension d. Let E be a Euclidean space or an integer grid, A a binary image in E, and B a structuring element[12].

$$A \ominus B = \{z \in E | B_z \subseteq A\}, \quad - (5)$$

where $\boldsymbol{B_z}$ is the translation of B by the vector z,

$$B_z = \{b + z | b \in B\}, \forall z \in E. \quad - (6)$$

When the structuring element B has a center (In our case it is rectangle. Since, most of the number plates are in rectangular geometry Fig. 7), and this center is located on the origin of E, then the erosion of A by B can be understood as the locus of points reached by the center of B when B moves inside A. For example, the erosion of a square of side 10, centered at the origin, by a disc of radius 2, also centered at the origin, is a square of side 6 centered at the origin.

The erosion of A by B is also given by the expression:

$$A \ominus B = \bigcap_{b \in B} A_{-b}. \quad - (7)$$

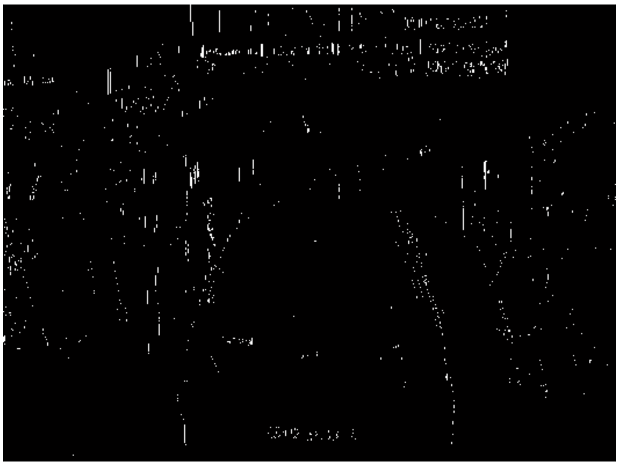


Fig. 7: Morphologically eroding the image with rectangle as structural element.

Dilation is one of the basic operations in mathematical morphology. Originally developed for binary images, it has been expanded first to grayscale images, and then to complete lattices. The dilation operation usually uses a structuring element for probing and expanding the shapes contained in the input image.

In binary morphology, dilation is a shift-invariant (translation invariant) operator[1][7].

A binary image is viewed in mathematical morphology as a subset of a Euclidean space $\mathbf{R}^{\mathbf{d}}$ or the integer grid $\mathbf{Z}^{\mathbf{d}}$, for some dimension d. Let E be a Euclidean space or an integer grid, A a binary image in E, and B a structuring element. The dilation of A by B is defined by:

$$A \oplus B = \bigcup_{b \in B} A_b. \quad - (8)$$

The dilation is commutative, also given by:

$$A \oplus B = B \oplus A = \bigcup_{a \in A} B_a \qquad \text{- (9)}$$

The dilation is commutative, also given by: If B has a center on the origin, then the dilation of A by B can be understood as the locus of the points covered by B when the center of B moves inside A. The dilation of a square of side 10, centered at the origin, by a disk of radius 2, also centered at the origin, is a square of side 14, with rounded corners, centered at the origin. The radius of the rounded corners is 2[11].

Image morphologically is closed doing both erosion and dilation as in Fig. 8. Precisely the rectangle remains the structural element for both. Hence, the possible number plate rectangular geometric shapes can be obtained.

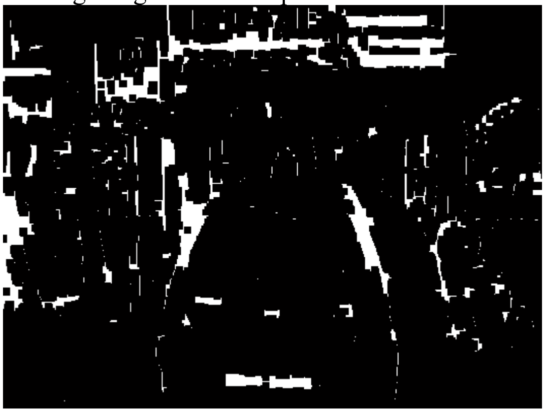


Fig. 8: Morphologically closing of the image.

Before the image is further processed for segmentation of the plate via canny edge detection we remove the unwanted small information which is typically the fragments.

2) Canny Edge Detection

Recognition of Plate is done using canny edge detection which is a multi-step process, which can be implemented as a sequence of filters [2][3]. The Canny Edge Detector is one of the most commonly used image processing tools that detects edges in a very robust manner as it tries to find the local Maxima and filter out unnecessary information as shown in Fig. 9.

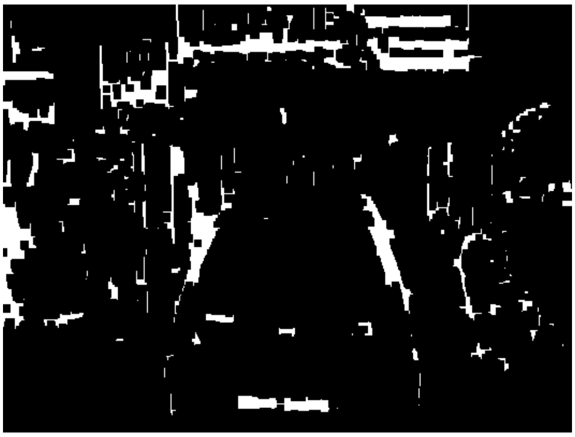


Fig. 9: Filtered image after closing.

The various steps that are involved in morphological edge detection using canny edge detection algorithm is as given below:

Step 0: Convert to Grayscale.
Step 1: Noise Reduction
Step 2: Compute Gradient Magnitude and Angle
Step 3: Non-Maximum Suppression

The recognized number plate as shown in Fig.10 after morphological edge detection and morphological operations and effective projections in both horizontal and vertical direction and canny edge detection.

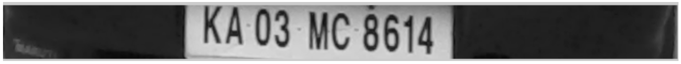


Fig. 10: Recognized number plate

B. *Automation Of The Number Plate module*

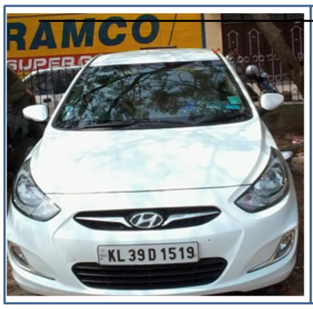
The automation of the number plate can be achieved by sending the number plate recognized to the concerned information centre automatically. By setting the domain server as the default server and accessing the permission from the particular domain we could send the mail to the prescribed domain once the plate is recognised.

C. *Validation Of The Number Plate module*

This is done by validating the database with the recognized number plate image. The information of the vehicle owner can be easily then accessed by the vehicle number. The Number plate recognized is cross checked with the constraints provided by the RTO.

V. Experimental Results

Case1: Image with multiple text in background



Input Image

Image after Morphological closing

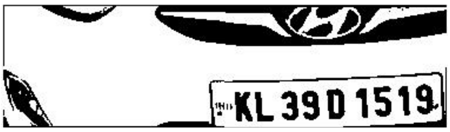
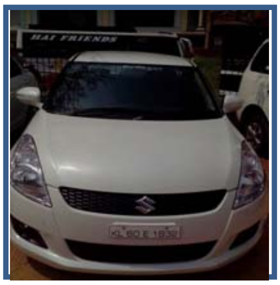


Image after canny edge detection

Time taken is 1.288770 seconds.

Case2: Text Information On Multiple Vehicles (Foreground and Background)



Input Image

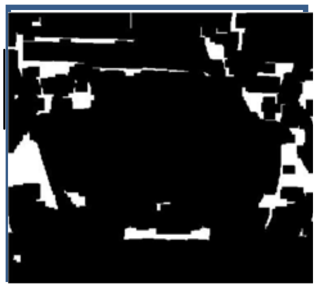


Image after Morphological closing

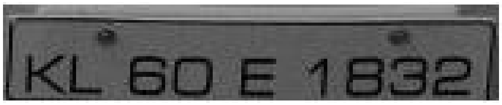
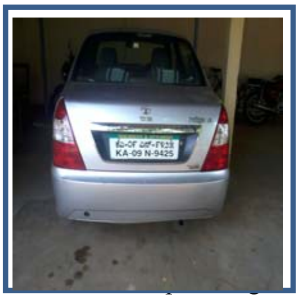


Image after canny edge detection

Time taken is 1.181035 seconds

Case3: Multiple Language Number Plate With Poor Visibility



Input Image

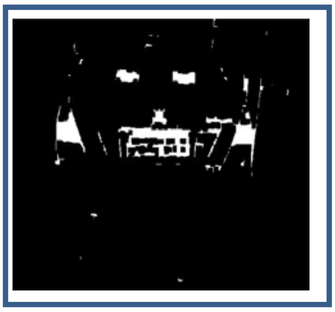


Image after Morphological closing

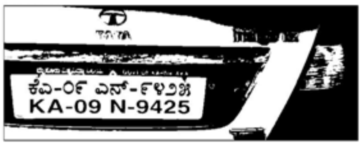
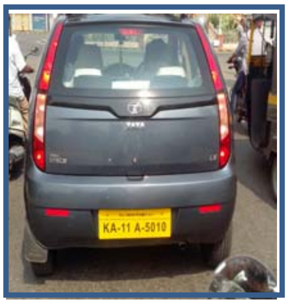


Image after canny edge detection

Time taken is 1.550398 seconds

Case4:Moving Car With Yellow Board and Different Background.



Input Image

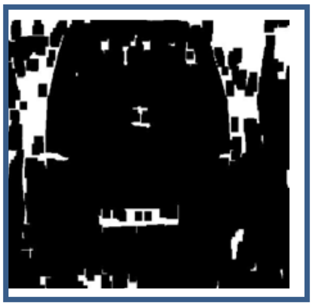


Image after Morphological closing

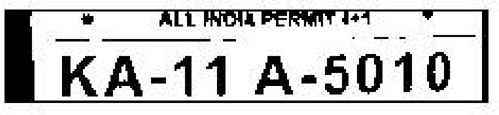
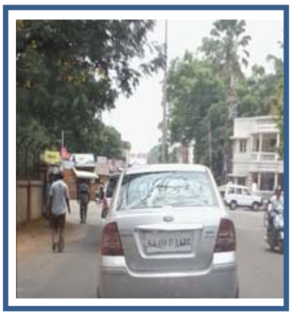


Image after canny edge detection

Time taken is 1.101353 seconds.

Case5: Moving Car With Different Geometrical Shaped Number Plate



Input Image

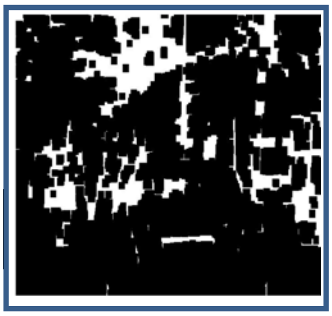


Image after Morphological closing

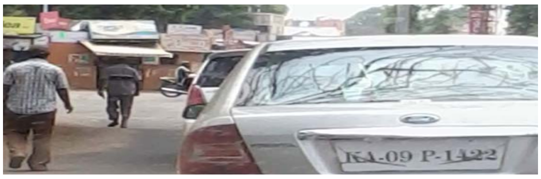


Image after canny edge detection

Time taken is 1.480692 seconds.

VI. Conclusion

A new robust method has been implemented for Automatic car number plate recognition. The input image is from the car with its plate under consideration for processing. The Captured Image in Jpeg format to its equivalent grayscale format. Further by applying sobel operator we detect the edges in both horizontal and vertical directions. The images obtained are further masked and eroded using rectangle as the structural element. Our Experimental results have proven the success rate of the proposed concepts high and hence this can be implemented to improve the security in our surrounding.

In our paper we have also taken steps to refine the edges and calculate the overall time taken in processing and refinement of the captured image. Our paper can be further enhanced to improve the OCR concept apart from primitive implementations for a car image that is captured dynamically.

References

[1] +Sobel Edge Detection Algorithm Samta Gupta1, Susmita Ghosh Mazumdar2 1M. Tech Student, Department of Electronics & Telecom, RCET, CSVTU Bhilai, India, 2Reader, Department of Electronics & Telecom, RCET, CSVTUBhilai, India. Vol 2 Issue 2 February 2013.

[2] Kuldeepak , Monika kaushik , Munish Vashishath, "License Plate Recognition System based on Image Processing "International Journal of Electronics Communication and Computer Technology (IJECCT) Volume 2 Issue 4 July 2012.

[3] Elizabeth Dekain and Songju kim, "Journal of Transportation Technologies". 13-21 http://dx.doi.org/10.4236/jtts.Published Online January 2012.

[4] Kumar Parasuraman and P.Vasantha Kumar,"An Efficient Method for Indian Vehicle License Plate Extraction and Character Segmentation", IEEE International Conference on Computational Intelligence and Computing Research 2010.

[5] Rafael C.Gonzalez , Richard E.Woods ,Digital Image Processing(India, PEARSON publications,2008).

[6] Milan Sanka, Vaclav Hlavac, Roger Boyle,Digital Image Processing and Computer Vision(New Delhi, CENGAGE Learning Publications, 2008)

[7] Algotithmic and Mathematical Principles of Automatic Number Plate Recognition System ,Ondrejmarinsky,Brno 2007

[8] http://javaanpr.sourceforge.net/anpr.pdf

[9] http://Automatic%20License%20Plate%20Recognition

[10] http://www.police.uk/information-and-advice/automatic-number-plate-recognition/

[11] http://www.ibtta.org/files/PDFs/Kennedy_Jim1.pdf

[12] http://www.platerecognition.info/1102.htm