A METHOD FOR SEGMENTING AND RECOGNIZING A VEHICLE LICENCE PLATE FROM A ROAD IMAGE

Abdelhalim Boutarfa, Mahfoud Hamada
Advanced Electronics Laboratory (LEA), Chahid Boukhlouf Road, University of Batna, 05000, Algeria

Emptoz Hubert
RFV Laboratory, LIRIS, 20 avenue Albert Einstein, 69621 Villeurbanne Cedex, France

Keywords: Intelligent Transportation Systems, License plate, Feature extraction, Segmentation, Pattern Recognition.

Abstract: To solve the problems of heavy traffic, due to the increase in the number of vehicles, modern cities need to establish effectively automatic systems for traffic monitoring and management. One of the most useful systems is the License-Plate Recognition System which captures images of vehicles and reads the plate's registration numbers automatically. Our method in this paper presents a robust algorithm for segmenting and recognizing a vehicle license plate area from a road image. As preprocessing steps, we statistically analyze the features of some sample plate images, and compute thresholds for each feature to decide whether a pixel is inside a plate or we cannot decide it. Our methodology starts from constructing the binary version of a road image according to the thresholds. Then, we select at most three strong candidate areas by searching the binary image with a moving window. The plate area is selected among the candidates with simple heuristics. Our algorithm is stable and robust against the cases of plate transformation and/or decolorization. The experimental results show 98.05% of successful plate recognition for 256 input images.

1 INTRODUCTION

Automatic vehicle identification is an essential stage in intelligent traffic systems. Nowadays vehicles play a very big role in transportation. Also the use of vehicles has been increasing because of population growth and human needs in recent years. Therefore, control of vehicles is becoming a big problem and much more difficult to solve. License plate recognition is a form of automatic vehicle identification. It is an image processing technology used to identify vehicles by only their license plates. LPR has many applications like highway electronic toll collection, red light violation enforcement (Bailey, 2002) border and customs checkpoints and speed limit enforcement. Since every vehicle carries a unique license plate, no external cards, tags or transmitters need to be recognizable, only license plate.

2 PREVIOUS WORK

The vehicle recognition from road images is one of the basic operations for traffic monitoring, detecting illegal vehicles, and automatic fee collection systems in parking lots or highways.

Especially for ITS (Intelligent Transportation System), vehicle recognition is one of the most important problems which should be solved. One of the problems directly related to the vehicle recognition is recognizing license plates (Naito, 2000), (Rahman, 2003).

Generally, vehicle license plate recognition problem consists of two parts:

1. Detecting and segmenting the plate area.
2. Recognizing the characters inside the plate.

(Chang & al, 2004) proposed a robust license plate detection algorithm using color edge and fuzzy disciplines. However, their algorithm can only detect the license plates with specific colors. In (Matas & al, 2005) the authors proposed an algorithm to detect license plate and road sign. They

Boutarfa A., Hamada M. and Hubert E. (2010).
A METHOD FOR SEGMENTING AND RECOGNIZING A VEHICLE LICENCE PLATE FROM A ROAD IMAGE.
In *Proceedings of the International Conference on Computer Vision Theory and Applications*, pages 413-419
DOI: 10.5220/0002840004130419

used character regions as basic units of license plate, which makes the algorithm hardly distinguish interference characters from the true license plates.

There have been many attempts for the plate area segmentation based on various techniques including a binary image processing (Takatoo, 1987), neural networks and Markov random field (Chui, 1997), the mathematical morphology (Hsieh, 2002), the color and edge data from the plate and inner characters (Xu, 2004) and so on.

Gao and Yang applied image processing techniques for the plate area to have high feature values, to finally detect candidate areas for the plate. (Gao, 2000) use the observation that there is a big difference between the contrast of the plate background and inner characters. They also use the observation that the average gradient value of the plate area is high, but the variance is low.

(Yang, 2005) computes histograms of the vertical and horizontal gradients and then applies mathematical morphology to select candidate areas for the plate.

In this paper, we present an algorithm to detect and segment for recognition the plate areas from road images. The input images are still images with 1280×960 resolutions that are captured from the CCD camera which is installed on a support pole above a road. In a preprocessing step, we analyze the features of the sample plate images. Then, we compute thresholds to distinguish a plate area from remaining areas. Those thresholds are used to binaries the input image later.

Given an input image, our algorithm applies image processing techniques, such as the transformation of its color model and applying Sobel edge operator. Then, we binarize input images to emphasize the plate area. In the next stage, we use a fixed-size window to search the plate area over the whole binary image. Through moving the window over the binary image, we can find at most three non-overlapping plate candidate areas according to the accumulation of pixel values inside the window. We finally choose the final plate area from the candidates based on simple heuristics. The advantages of our method can be summarized as follows:

1. The algorithm robustly detects and segments the plate area even for the cases when the plate in the image is inclined or transformed.
2. The algorithm is not affected by the changes of illumination, camera exposure, or the decolorization of plates.
3. In spite of relatively simple preprocessings with a small number of sample images, experimental results show high success rates.

In this paper, we restricted the input and sample images as those captured in the People's Republic of China. However, our algorithm is not limited to be applied only for Chinese license plates (see figure 7). The license plates from other countries also have similar features such that the usage of numbers and characters and some specific pair of colors for the plate. By varying the parametric values and the thresholds, we can more generalize our plate segmentation algorithm. This paper is organized as follows:

In Section 3, we analyze the feature of license plates, and construct a preprocessing stage based on the statistical analysis. In the next Section, we present an algorithm for segmenting and recognizing a license plate and its validation. The experimental results are shown in Section 5. We conclude this paper in Section 6.

3 FEATURES EXTRACTION

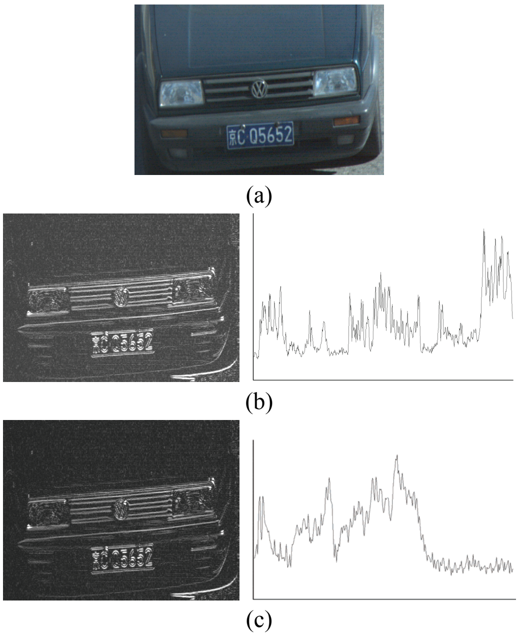
To isolate the license plate area from road images, we need distinctive features to distinguish it from other areas. This paper uses road images captured in the People's Republic of China to demonstrate our method of isolating license plate areas. Thus, we focus on the Chinese license plate and consider the following three aspects:

1. Edge features due to the characters used in the license plate.
2. Chromatic features due to the colors used in the license plate.
3. Geometric features due to the shape of the license plate.

Our license plate area extraction algorithm is derived from these distinctive features. Analyses and pre-processing steps for them are followed.

3.1 Edge Features from the Characters

License plates used in the People's Republic of China generally consist of a single Chinese character, a single alphabet and five alphanumeric characters in a row. Due to their geometric shapes, alphanumeric characters contain relatively large quantities of vertical edges. Thus, we can expect that the license plate area will have more vertical edges compared to other areas.



(a) original road image
(b) horizontal edges and their histogram
(c) vertical edges and their histogram

Figure 1: Characteristics of the Sobel's edge detection operator applied to the front side of vehicle.

Figure 1 shows the result of horizontal and vertical edge extraction through applying Sobel edge operators to the grayscale image containing a license plate. Figure (1a) is the original road image. Figures (1b) and (1c) are horizontal and vertical edge extraction results and their corresponding histograms, respectively.

As shown in figure (1b), the lower body of a vehicle contains many horizontal edges in several areas and we met difficulties in using horizontal edges to find the license area. In contrast, as shown in figure (1c), the license plate area contains more vertical edges in comparison with other areas. Due to the side areas or some models of radiator grills of the vehicle, the vertical edge feature alone cannot identify the license plate area.

To calculate the characteristics of vertical edges in the license plate area, we use a set of sample images. Letting n be the number of sample images, we extract minimal rectangular area R_i, $1 \le i \le n$ containing the license plate area, for each sample image (Fig. 2a).

When p_{xy} is the pixel at (x,y) in the input image, the rectangular area R_i, can be defined as a set of pixels as follows:

$$R_i = \{p_{xy} \mid x_{\min}(R_i) \le x \le x_{\max}(R_i) \text{ and } y_{\min}(R_i) \le y \le y_{\max}(R_i)\}$$

where $x_{min}(R_i)$, $x_{max}(R_i)$, $y_{min}(R_i)$, and $y_{min}(R_i)$ are the minimum and the maximum values of x and y coordinates for R_i, respectively.

We first convert each pixel P_{xy} in R_i to grayscale values. Then apply Sobel edge operator to calculate the vertical edge value $Edge^{vertical}(p_{xy})$ (Fig. 2b).

After calculating $Edge^{vertical}(p_{xy})$ values, we accumulate them along the vertical direction, to construct histograms as shown in Figures (b). In the histogram, the more vertical edge portions due to the characters in the license plate are isolated from the less vertical edge portions corresponding to the blank areas, and we can easily distinguish the characteristics of the vertical edges.

In this paper, we use 25 sample images to extract minimal rectangular area R_i,s and construct corresponding histograms. After analyzing the vertical edge portions and inverse mapping the result to $Edge^{vertical}(p_{xy})$ values, the $Edge^{vertical}(p_{xy})$ values show a distribution with the mean of $\mu_{edge}=52.196$ and the standard deviation of $\sigma_{edge}=24.342$.

As shown in the example histogram of figure (1b), the character portions and the blank portions show large variations. From the statistics point of view, we use only uppermost 10% of edge detection values for the final decision step of the algorithm that will be presented in section 4.

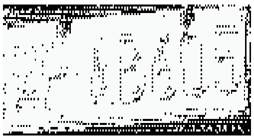
After calculating $Edge^{vertical}(p_{xy})$,s the pixels with values larger than $\mu_{edge}+1.33\ \sigma_{edge}$ are considered as the candidates for the license plate area pixels. This corresponds to the uppermost 10.02% of vertical edge values, when interpreted as a normal distribution.

3.2 Chromatic Features

Since the license plate has a unique background color and another distinctive character color, we can use this chromatic feature to find the license plate area. Original input images use the *RGB* color model, which do not distinguish color and intensity information. Thus, to normalize the intensity and camera exposures at the capture time, we convert the *RGB* color values into the *HSV* (hue-saturation-value) color values. In the *HSV* model, intensity information is isolated in the *V(value)* channel and we can get the intensity-normalized result using only the *H(hue)* and/or *S(saturation)* channels (Shyang-Lih, 2004).

To extract chromatic features, we select the most widely used license plate type, which is usually used for general passenger cars and small cargo trucks in China, whose background color is dark blue and characters are painted in white color (see figure 2).

(a)



(b) (c)

(a)

(b) (c)

(a) rectangular region R_i from original images
(b) extraction of vertical edges
(c) binary image due to the hue value processing

Figure 2: Image processing results.

Our experimental results show that the *S(saturation)* channel does not show distinctive characteristics, while the distribution in the *H(hue)* channel does. To analyze the distribution of hue values, we use a set of sample images. In this paper, we use 25 sample plate images S_i, $1 \le i \le 25$, which are extracted from 25 road images and aligned along the x and y axis. Each sample image can be considered as a set of pixels as follows:

$$S_i = \{p_{xy} \mid x_{\min}(S_i) \le x \le x_{\max}(S_i) \text{ and } y_{\min}(S_i) \le y \le y_{\max}(S_i)\}_{\text{W}}$$

here $x_{min}(S_i)$, $x_{max}(S_i)$, $y_{min}(S_i)$, and $y_{min}(S_i)$ are the minimum and maximum of x and y coordinates of the rectangular area covered by S_i, whose size varies for each sample image.

For more simple and efficient processing, the hue value $hue(p_{xy})$ for a pixel p_{xy} is normalized into integer values between 0 and 255.

To get distributions of hue values for S_i,*s* we use the following algorithm:

Algorithm 1

```
const int numImages = 25
array hueCount[i][h] : for the i-th image, contain no. of pixels with hue value h
for i =1 to numImages do
for each pixel pxy ∈ Si do
h ← hue(pxy) // normalized hue value: 0 to 255
 hueCount[i][h] = hueCount[i][h] + 1
 end
end
for each possible hue value h do
 averageHue[h] = average of hueCount[i][h]
end
```

Figure 3 shows the final distribution of hue values. From this discrete distribution, we get the mean of $\mu_{hue}=149.992$ and the standard deviation of $\sigma_{hue}=13.794$. A normal distribution corresponding to these values in figure 3 are drawn in red color.

In our paper, we select the confidence interval of ($\mu_{hue} - \sigma_{hue}$, $\mu_{hue} + \sigma_{hue}$) corresponding to 72.36% of confidence. Thus, in our experiments, the pixels between $H_{min}= \mu_{hue} - \sigma_{hu} =136.198$ and $H_{max}= \mu_{hue} + \sigma_{hu}=163.786$ (marked as gray area in figure 3) are considered as the license plate area pixels. When assigning 1,s to the pixels with $\mu_{hue} - \sigma_{hue} \le hue(p_{xy}) \le \mu_{hue} + \sigma_{hue}$ and 0,s to others, we can get a binary image reflecting this chromatic feature, as shown in figure 2(c).

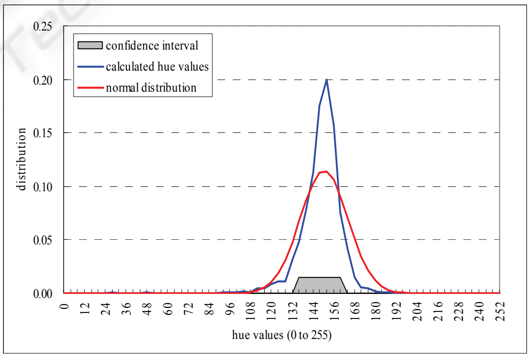


Figure 3: Distribution of hue values for the sample license plates.

3.3 Geometric Features

In China and other countries, the size of a license plate is specified in a law. Thus, we can calculate the expected size of the license plate in a road image, from the plate size and camera configurations. In this paper, we use 25 images to get the expected size of *wxh* pixels.

We use the value of $w=185$ and $h=75$ which contains some extra pixels for efficient implementation of our algorithm shown in the section 4. Our experiments with 256 road images show that this plate size can sufficiently contain the plate area for all finally successful segmentations.

4 SEGMENTATION ALGORITHM

In this section, we present a license plate area segmentation algorithm, based on the features explained in the previous sections. For each pixel p_{xy}, we choose the threshold value of $\mu_{edge} + 1.33\sigma_{edge}$ to find uppermost 10.02% of Sobel vertical edge detection results, as explained in section 3.1. To reflect chromatic characteristics of the license plate area, an interval of hue values ($\mu_{hue} - \sigma_{hue}$, $\mu_{hue} + \sigma_{hue}$) will be used as the threshold values. Additionally, from the size of the license plate and camera configurations, we calculated the expected window size *wxh* for searching the plate area in an input image.

Figure 4 presents the block diagram of our license plate area segmentation algorithm, using all these features.

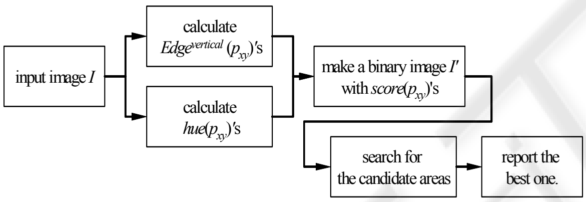


Figure 4: Block diagram of our license plate area segmentation system.

To simplify the area search process, we first construct a binary image I' from an input image I. An input image represented in the *RGB* color model is converted into a *HSV* color model-based image. In this *HSV* representation, we can get the grayscale image of the original input image through extracting the V(value) channel only. As explained in section 3.1, we apply the Sobel vertical edge operator to calculate $Edge^{vertical}(p_{xy})$ values.

The *H(hue)* values of the *HSV* model are directly used as $hue(p_{xy})$,s.

For the $Edge^{vertical}(p_{xy})$ values, only the uppermost 10.02% of them are considered as the license plate area pixels, using the threshold of $Edge^{vertical}(p_{xy}) \geq \mu_{edge} + 1.33\,\sigma_{edge}$, based on a statistical approach.

Similarly, the pixels with hue values of $\mu_{hue} - \sigma_{hue} \leq hue(p_{xy}) \leq \mu_{hue} + \sigma_{hue}$ are considered as the license plate area pixels, as explained in section 3.2.

Combining these two restrictions, the final score function $score(p_{xy})$ for a pixel (p_{xy}) is calculated as follows:

$$score(p_{xy}) = \begin{cases} 1 & \text{if } Edge^{vertical}(p_{xy}) \geq \mu_{edge} + 1.33\,\sigma_{edge} \\ & \text{and } \mu_{hue} - \sigma_{hue} \leq hue(p_{xy}) \leq \mu_{hue} + \sigma_{hue} \\ 0 & \text{otherwise} \end{cases}$$

where μ_{edge}, σ_{edge}, μ_{hue} and σ_{hue} are pre-calculated values from a set of sample images.

Calculating $score(p_{xy})$,s for all pixels in the input image, we can construct a binary image I' as shown in Figure 5.

Figure 5: Binary image of $score(p_{xy})$.

In this binary image, we can easily find that the pixels with $score(p_{xy})=1$ are certainly gathered in the license plate area.

In Section 3.3, we show that the size of a license plate area is always contained in *wxh* pixels, for our camera configurations and resulting road images. Thus, a rectangular area *R* with the size of *wxh* are moved over the whole input image and the total score of $score(R)=\sum_{p_{xy} \in R} score\ (p_{xy})$ are used to finally find the license plate area. Conceptually, we calculate all *score(R)* values for all possible *wxh* size rectangular area *R,s* and sort the *score(R)* values in a non-decreasing order. When some areas with high *score(R)* values are overlapped, only the maximum one is remained and others are removed from the sorted list.

Our experimental results show that headlight lamp areas and/or radiator grill areas can also show high *score(R)* values in some cases for several specific vehicle models and/or some specific vehicle colors.

To remove these unexpected areas, at most three areas with high *score(R)* values in the sorted list are considered as strong candidates. Since the license plates are usually located in the lower-center portion, we prefer the one with lower location, and additionally with high *score(R)* value, among the at most three candidates. The overall skeleton of our algorithm is shown as follows:

Algorithm 2

```
    input:  a road image I (with a specific
camera      configuration)
    output: the license plate area
begin
    list candidate = ∅
    // calculate score function for each
pixel
    for each pixel p_xy ∈ I do
            calculate   Edge^vertical(p_xy)   and
            hue(p_xy)
            set score(p_xy)
    // search for the candidate areas
    for each possible location (x,y) in I do
            set the rectangular region R at
(x,y)
            calculate score(R) = Σ_{p_xy ∈ R} score(p_xy)

            candidate = candidate ∪ score(R)
    remove overlapping areas in the candidate
list
    sort the candidate list
    // pick the best three candidates and
    report the area
    for i = 1 to 3 do
            if the best candidate is located
            in the acceptable area
   or i = 3 then
            report it as the number plate
area
    else
            discard the best candidate and
retry
    end if
end
```

5 EXPERIMENTAL RESULTS

The algorithm represented in Section 3 was implemented as a program using Visual Cpp. We used 256 input images with 1280×960 resolutions, which are captured by a CCD camera from a road.

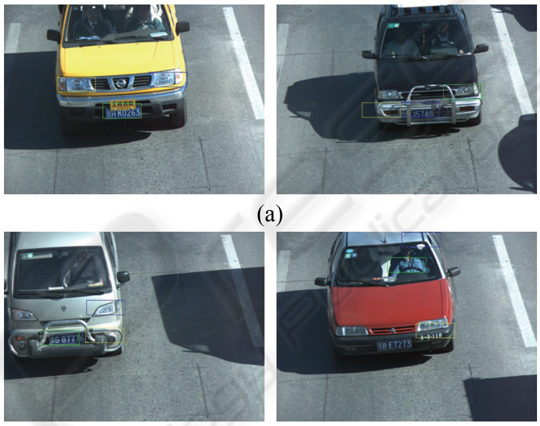
In the preprocessing stage, we used 25 sample images. Table 1 shows the experimental results.

Table 1: Experimental Results.

Classification		Number	Rate
Success		223	98.05%
Failure	Partial Segmentation	3	1.17%
	Wrong Segmentation	2	0.78%
Total		228	100.00%

For the successful cases, the area fully including the plate is segmented as the result. Our experiments show 98.05% of success rate. The input images include the cases: i) the plates are bent or decolorized, ii) there are complicated equipments around the plate, and iii) there are characters outside the plate. Our algorithm successfully segmented the plate areas of those cases (Fig. 6a).

We classified the cases when the segmentation result does not include the whole plate but the detection of plate position was right as a partial segmentation. There was 1.17% of partial segmentation (Fig. 6b). We had only 0.78% of failure rate that failed to both of detecting and segmenting the plate (Fig. 6c).



(a)

(b) (c)

(a) Success

(b) Partial segmentation

(c) Failure case

Figure 6: Segmentation results.

Other experiments have been performed to test the proposed system and to measure his accuracy. The test images were taken under various illumination conditions (see Fig.7).

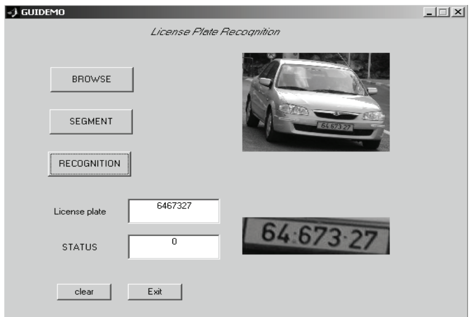


Figure 7: Recognition results for vehicle license plate.

6 CONCLUSIONS

In this paper, we presented an algorithm for segmenting and recognizing the license plate area from a road image. It has ability to correctly recognize all license plates located in the picture, in a short time, even if they are dirty or containing small mechanical damages. We analyzed the feature of plates by considering the distribution of vertical edges inside the plate, the distribution of hue values from the color of plates, and the geometric shape of the plates. Based on those features, we constructed a preprocessing stage that statistically analyzes the sample plate images.

Given a road image, our algorithm computes its binary image by using the thresholds derived in the preprocessing stage. By moving a fixed-size window over the binary image, we search candidate areas for the plate, which has the local maximum accumulation of pixel values. Our algorithm successfully detects and segments the plate area for 98.05% cases from 256 input images.

The algorithm robustly detects and segments the plate area even for the cases when the plate in the image is inclined or transformed. It is also stable to the changes of illumination, camera exposure, or the decolorization of plates. In spite of relatively simple preprocessing with a small number of sample images, the experiments show high success rates.

According to the proved experimental results, one can conclude that our method in comparison with previous works on subject (Sulehria, 2007), (Arlazarov, 2008), (Ispas, 2008) is effective and fast to be employed with the practical applications.

It arises from it the direct advantages as follows

1. The algorithm implementation area is remarkably reduced.
2. The approximations leading to area reduction do not cause significant sacrifices since any required precision may be recovered when switching regularly to the usual algorithm.

An important extension of this work is to implement a new algorithm using hybrid process based on neural network and Hough Transform to analyze the geometric defaults obtained in edges of images in the process. Further work is needed within the proposed framework to improve provide flexible bandwidth adaptation and robustness.

REFERENCES

D.G. Bailey, D. Irecki, B.K. Lim and L. Yang, 2002. Test bed for number plate recognition applications. *Proceedings of 1st IEEE International Workshop on Electronic Design, Test and Applications, Delta'02, IEEE Computer Society.*

T. Naito, T. Tsukada, K. Yamada, K. Kozuka, 2000. Robust license plate recognition method for passing vehicles under outside environment. *IEEE Transactions Vehicular Technology, 49 (6), pp. 2309-2319.*

C. Rahman, W. Badawy, A. Radmanesh, 2003. A real time vehicle's license plate recognition system. *Proc. of IEEE Conference on Advanced Video and Signal Based Surveillance, pp. 163-166.*

S.L, Chang YC, Chen Chung and SW Chen, 2004. Automatic License Plate Recognition. *IEEE Transactions on Intelligent Transportation Systems, 5(1), pp. 42-53.*

J. Matas, K. Zimmermann, 2005. Unconstrained License Plate and Text Localization and Recognition. *Proc. of International Conference on Intelligent Transportation System, pp. 572-577.*

M Takatoo, M. Kanasaki, T. Mishima, T. Shibata, H. Ota, 1987. Gray scale image processing technology applied to vehicle license number recognition system. *Proc. of IEEE International Workshop Industrial Applications of Machine Vision and Machine Intelligence, pp. 76-79.*

Y Chui, Q. Huang, 1997. Automatic license extraction from moving vehicles. *Proc. of International Conference Image Processing, pp. 126-129.*

J.W. Hsieh, S.H. Yu, Y.S. Chen, 2002. Morphology based license plate detection from complex scenes. *Proc. of 16th International Conference on Pattern Recognition, Vol. 3, 176-179.*

J.F. Xu, S.F Li, M.S. Yu, 2004. Car license plate extraction using color and edge information. *Proc. of IEEE 3rd International Conference on Machine Learning and Cybernetics, pp. 3904-3907.*

D.S. Gao, J. Zhou, 2000. Car license plates detection from complex scene. *Proc. of 5th International Conference on Signal Processing, pp. 1409-1414.*

F.Yang, Z. Ma, 2005. Vehicle license plate location based on histogramming and mathematical morphology. *Proc. of 4th IEEE Workshop on Automatic Identification Advanced Technologies, pp. 89-94.*

Chang Shyang-Lih, Chen Li-Shien, Wan Sei-Chen. 2004. Automatic License Plate Recognition Intelligent Transportation Systems, *IEEE Transactions 5(1), pp. 42-45.*

GA666-2006, People's Republic of China regulations for vehicles number plates (in Chinese).

H.K. Sulehria, Y. Zhang, D. Irfan, 2007. Mathematical morphology methodology for extraction of vehicle number plates. *International Journal of Computers, WSEAS, 1 (3), pp. 312-319.*

V.L. Arlazarov, M. Kazanov, 2008. Segmentation of small objects in color images. *Programming and Computing Software, 34 (3), pp. 173-182.*

I. Ispas, E. Franti, F. Lazo, 2008. The complexity of the algorithms for the image recognition and classification (IRC). *Proceedings of the WSEAS International Conference on Applied Computing Conference, Istanbul, Turkey, pp. 160-164.*