An Efficient Features–Based License Plate Localization Method

Hamid Mahini, Shohreh Kasaei
Computer Engineering Department
Sharif University of Technology
P.O. Box 11365–9517, Tehran, Iran
mahini,skasaei@ce.sharif.edu

Faezeh Dorri, Fatemeh Dorri
Electrical Engineering Department
Amirkabir University of Technology(Tehran Polytechnic)
P.O. Box 15875–4413, Tehran, Iran
fdorri,f_dorri@dena.aut.ac.ir

Abstract

This paper presents a feature-based license plate localization algorithm that copes with multi-object problem in different image capturing conditions. The proposed algorithm is robust against illumination, shadow, scale, rotation, and weather condition. It extracts license plate candidates using edge statistics and morphological operations and removes the incorrect candidates according to the determined features of license plates. We have formed a rather complete database of 269 *images in different conditions. The proposed algorithm successfully detecteds the accurate location of the license plates in* 96.5% *cases, which outperforms the other available approaches in the literature.*

1 Introduction

The recent developments in automatic object tracking and the importance of traffic monitoring has risen the attention to *intelligent transport systems* (ITSs). As such, the vehicle *license plate recognition* (LPR) has turned out to be an important research field. The LPR system can be used for parking management systems, traffic control, ticketing, detecting stolen vehicles, and the forth.

Usually, an LPR system consist of three main parts: license plate detection, character segmentation, and character recognition. Among these the *license plate detection* (LPD) is the most important stage and also the most difficult part. This is mostly because during this stage the algorithm has to overcome various undesired input image conditions such as out of focus (blur) images, undesired illumination conditions, small size plates, rotations, shadows, and different weather conditions. Different LPD techniques are introduced in the literature. This includes different approaches base on edge statistics analysis [1][3][9], morphological filtering [3][5][8], hough and radon transform-based [6][10], neural networks [7], and combination of plate features [2][4][7]. As expected, using a combination of these techniques can yield better results. For example just considering edge information is not a good approach in complex images with small license plates (as there might exist a number of undesired edges).

In this paper, we present an accurate LPD algorithm that is robust to above mentioned problems. The proposed algorithm can detect plates of different sizes (the height of plate can be in the range of $\frac{1}{2}$ to $\frac{1}{50}$ of image height), different illumination conditions, rotations, scales, shadows, and the real world noise. We have applied our approach on Iranian license plates which contains arabic numbers but it can also be used for other plates with only a few changes.

The rest of the paper is organized as follows. Section 2 describes some definition, in Section 3 the proposed methods of finding license plates candidates and omitting the incorrect ones are presented. The experimental results are discussed in Section 4, and finally Section 5 concludes the paper.

2 Terminology and Definitions

First of all for better understanding of our proposed algorithm we briefly explain our terminology and some morphological operations.

Assume that $S_{m\times n}$ is a structuring element with size $m\times n$ with all its entries equal to one, I refers to a gray-scale image, and $\odot$ and $\oplus$ denote the erosion and dilation operation in the morphological filtering, respectively. There are some definitions of other morphological operations which are useful in describing our algorithm:

Close operation: $I \bullet S_{m\times n} = (I \oplus S_{m\times n}) \odot S_{m\times n}$
Open operation: $I \circ S_{m\times n} = (I \odot S_{m\times n}) \oplus S_{m\times n}$
Bothat operation: $I \bigtriangledown S_{m\times n} = I \bullet S_{m\times n} - I$
Tophat operation: $I \bigtriangleup S_{m\times n} = I - I \circ S_{m\times n}$

3 Plate Localization

Because a wide range of plate sizes may appear in real scenes we should be able to resize images whenever needed.

0-7695-2521-0/06/$20.00 (c) 2006 IEEE

We start with a resized image of width 100 to find the candidate regions where its height is proportional to its real size. In each step, in the verification process if a plate candidate is found then we stop, otherwise we resized the image to a larger one.

As our mask and structuring element are constant and have predefined sizes, in order to find large/ small plates we can apply the algorithm on small/ large size images. This approach helps to find plate candidates with plate to image height ratio of 2%.

3.1 Finding plate candidates

As a plate area has many vertical edges, the dark characters are located on a light background, and the intensity of each pixel of the plate is almost the same for all three attributes (R, G, and B) resulting in a gray looking pixels, we use these characteristics to find the plate candidates.

3.1.1 Finding regions with many vertical edges

As the bothat operation emphasize dark character on light background, it is more beneficial to extract edges from the output of the bothat operator.

$$I_0 = I \bigtriangledown S_{2\times 8}. \tag{1}$$

To find the vertical edges we use the sobel vertical mask on I_0 and I_1 would be obtained . We take the absolute of edge values and scale them linearly between $0-255$, to prevent from negative and out of range values.

$$\begin{aligned} minI_1 &= \min_{i,j} Abs(I_1(i,j)) \\ maxI_1 &= \max_{i,j} Abs(I_1(i,j)) \\ I_2 &= 255 \times \frac{Abs(I_1) - minI_1}{maxI_1 - minI_1}. \end{aligned} \tag{2}$$

Since we may have some unwilling sharp edge values we smooth the edge values by using:

$$\begin{aligned} count(k) &= \sum_{I_2(i,j)=k} 1 \\ I_3(i,j) &= I_2(i,j) \times \sqrt{\log(count(I_2(i,j)))} \end{aligned} \tag{3}$$

and linear scaling of I_3 is crucial to obtain I_4. Finally we use a 5×5 median filter to remove salt and pepper type of noise.

$$Edge = median(I_4, S_{5\times 5}). \tag{4}$$

Figure 1(b) shows the result of finding vertical edges on images shown in Figure 1(a).

After finding vertical edges, we convolve it with a 3×30 identity mask. The convolved image is shown in Figure 2(a).

$$C_1 = Edge * S_{3\times 30}. \tag{5}$$

(a) (b)

Figure 1. Finding vertical edges: (a) input gray-scale image, (b) vertical edges.

3.1.2 Finding gray looking regions

The $C_2(i,j)$ is equal to 1 if the intensity of (i,j)'th pixel of the input image has the same value in R, G, and B components resulting in a gray looking pixel and otherwise is set to 0.

3.1.3 Finding the light background regions

To find the locations of the light backgrounds, apply the morphological close operation on the input image with the structural element $S_{3\times 3}$:

$$C_3 = I \bullet S_{3\times 3} \tag{6}$$

3.1.4 Merging results

Now, we merge C_1, C_2, and C_3 to obtain the candidate licence plate locations, by:

$$\begin{aligned} Candidate_1(i,j) &= C_1(i,j) \times C_2(i,j) \\ &\times C_3(i,j) \times (C_3(i,j) > 50) \\ Candidate_2 &= Candidate_1 \bigtriangleup S_{3\times 8}. \end{aligned} \tag{7}$$

Finally, we obtain the binary mask of the candidate licence plate using soft thresholding :

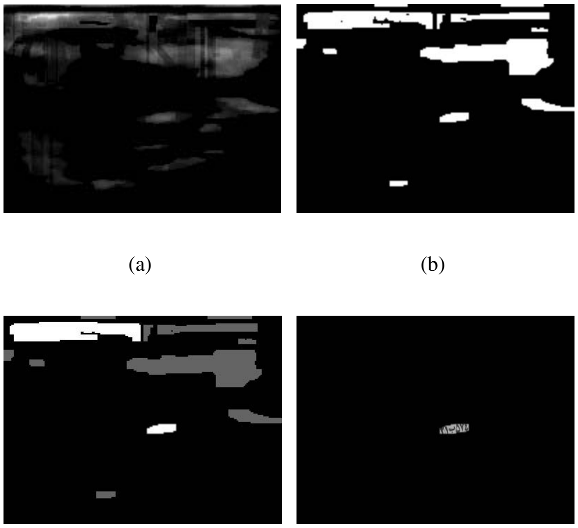
$$BinaryCandidate = Candidate_2 > Threshold \tag{8}$$

Now, each component of the binary mask is supposed to be a candidate of the plate region which is sent to the next stage. Figure 2(b) shows the plate candidate region of the image shown in Figure 1(a).

3.2 Plate Verification

Plate candidates are eliminated from the set of obtained candidates if they do not satisfying the following:

- plate region should not be very small.
- plate region shape should be similar to a rectangular and its width to height ratio must be between 2 and 10.



(a) (b)

(c) (d)

Figure 2. Obtained plate candidate regions: (a) plate edges after convolution, (b) plate candidates, (c) processes plate candidates, (d) plate location.

- plate region average intensity must be light enough.
- plate region should not be connected to the image margin .
- plate regions direction, using *principal component analysis* (PCA), should be almost horizontal (up to $35°$).

The above mentioned rules are applied on Figure 2(b) and the result is shown in Figure 2(c).

- We use the Otsu threshold to binarize the candidate regions and count the connected components which are almost perpendicular to the dominant plate direction. Note that we apply the otsu algorithm on each candidate separately to obtain better results.
- Let Y_l denotes the summation of pixel values belonging to a line of ls perpendicular to the first component of the PCA of the candidate region. Histogram of Y_l must have $4 - 10$ peaks if it is a correct plate region (about 6 for new Iranian "yellow" licence plates and about 8 for the new Iranian "blue" licence plates shown in Figure 3).

In Figure 2(d) we can see the result of our verification stage on candidates shown in Figure 2(b).

Figure 3. Samples of some Iranian plates.

Figure 4. Some sample images.

4 Experimental Results

Our database consists of 800×600 JPEG formatted color images. The images are taken from real scenes of any distance (scale) and viewing angle (rotation)that may be crowded, contain multi-objects, different weather conditions(cloudy, sunny, snowy) and Different illuminations. We have implemented the algorithm using C^{++} language on a Pentium IV PC with 512 Megabyte of RAM.

When evaluating the performance of other available approaches, it seems that the size of the plates in some techniques are almost fixed or is more than a ratio of the image. For instance the approach introduced in [1] can just detect plates which their heights are more than 5% of height of the image. Also, it seems that other approaches do not mention this point and cannot support detecting very small size plates either. But, in our algorithm we support detecting plates which their height is smaller than $\frac{1}{20}$ (up to $\frac{1}{50}$) of the image height. We also need that the plate height only includes about 10 pixels.

The proposed algorithm can detect the plates in our (complex) database, some of which are shown in Figure 4, even those in which the half of plate is in shadow.

Some other algorithms have not overcome the problems associated with uneven lightening, small size plates, and scenes with multi-objects. They are usually optimized for parking management or ticket passing systems. Our algorithm can handle wider applications in various conditions.

To have a better comparison among the performance of different algorithms, we need to run them on the same databases. Since access to different used databases is usually impossible, a precise comparison is hard to achieve. Furthermore, the exact image conditions and database fea-

Table 1. Performance of different algorithms.

Algorithms	Images#	Correct	No Plate	Extra region
[10]	256	57	-	199
[1]	269	53	156	60
Proposed	269	260	2	7
IR Images	2483	2466	3	14

tures are not mentioned in the related literature. By referring to the published sample images we can see that many approaches assume that the plate image sizes are fixed or somehow large. Since many approaches are designed to be used in parking management and ticketing systems, they assume that the lightening and weather conditions are under control. As such, in comparison, the proposed algorithm can cope with very complex situations in the real word and thus can be used in a variety of applications.

Our database contains 269 images of the new Iranian plates (Figure 3), which 13 of them contain no plate. Our algorithm successfully detects no plate in these 13 images. Other 256 images are taken from real scenes. Note that the algorithm can run for images with different resolution. The total average time elapsed by our algorithm to run 800×600 color JPEG formatted images is about 300 milliseconds with a high accuracy. Also, we have not optimized our algorithm implementation yet and thus its speed can increase by a better implementation. Also, in order to show the robustness of our algorithm we run it on 2483 infrared images and a correct detection rate of 99.3% was achieved. Result of our algorithm is shown in Table 1.

As the experimental results show, the proposed algorithm can detect above 96.5% of plate region correctly.

The result of implementing the approach introduced in [10] (based on our best understanding) when running on our database is shown in Table 1. That approach showed many problems with small size plates and multi-object scenarios which has many edges other that the plate edges. We also have run the approach introduced in [1] on our database. The results are given in Table 1. Because of different plate sizes, different illumination conditions, and existence of multi-objects in our database, those approaches have shown low performance when dealing with more complex databases. The performance comparison has also shown the ability of our method to handle difficult situations.

5 Conclusion

A robust approach which considers different features of license plate to deal with more complex situations in real world is presented. We first find the license plate candidates based on vertical edges, morphological operation, and color analysis of the images. Then, by eliminating the incorrect candidate regions based on images features the correct license plate regions are obtained. As opposed to the available algorithms reported in the literature, the proposed algorithm can be successfully run on our complex database. As shown in the experimental results section, the proposed algorithm is robust against different lightening conditions, shadows, small size images, rotations, and multi-object scenes.

Acknowledgment

This work was in part supported by YSC collaboration of researchers and Tehran municipality. The first author wishes to thank Mr. M. Lesani, M. Goodarzy and E. B. Ghader for their helps in implementing the algorithm.

References

[1] S. A. Boroujeni. Design and implementation of automatic system of image recording of vehicles license plate. Master's thesis, Amirkabir University of Technology(Tehran Polytechnic), 2000.

[2] Y. Cui and Q. Huang. Automatic license extraction from moving vehicles. *International Conference on Image Processing(ICIP97)*, 3:126–129, 1997.

[3] B. Hongliang and L. Changping. A hybrid license plate extraction method based on edge statistics and morphology. *17th International Conference On Pattern Recognition(ICPR'04)*, 2:831–834, 2004.

[4] C. T. Hsieh, Y. S. Juan, and K. M. Hung. Multiple license plate detection for complex background. *19th International Conference on Advanced Information Networking and Applications (AINA'05)*, 2:389–392, 2005.

[5] J. W. Hsieh, S. H. Yu, and Y. S. Chen. Morphology-based license plate detection from complex scenes. *16th International Conference On Pattern Recognition(ICPR'02)*, 3:179–179, 2002.

[6] V. Kamat and S. Ganesan. An efficient implementation of hough transform for detecting vehicle license plate using dsp's. *1st IEEE Real-Time Technology and Applications Symposium*, pages 58–59, 1995.

[7] S. Kim, D. Kim, Y. Ryu, and G. Kim. A robust license-plate extraction method under complex image conditions. *16th International Conference On Pattern Recognition(ICPR'02)*, 3:216–219, 2002.

[8] F. Martín, M. García, and J. L. Alba. New methods for automatic reading of vlps (vehicle license plates). *Signal Processing Patten Recognition and Application*, 2002.

[9] M. Sarfraz, M. J. Ahmed, and S. A. Ghazi. Saudi arabian license plate recognition system. *Proceedings of the 2003 International Conference on Geometric Modeling and Graphics(GMAG'03)*, pages 36–41, 2003.

[10] V. Shapiro, D. Dimov, S. Bonchev, V. Velichkov, and G. Gluhchev. Adaptive license plate image extraction. *International Conference on Computer Systems and Technologies*, 2003.