MobLP: A CC-based approach to Vehicle license plate number segmentation from images acquired with a mobile phone camera

Abhishek Sharma
Dept. of Electrical and Electronics Engineering
Birla Institute of Technology, Mesra, India
sharma.abhishek@ieee.org

Amey Dharwadker
Dept. of Electronics and Communication Engineering
National Institute of Technology, Tiruchirappalli, India
ameydhar@gmail.com

Thotreingam Kasar
MILE Lab, Dept. of Electrical Engineering
Indian Institute of Science, Bangalore, India
tkasar@ee.iisc.ernet.in

***Abstract*— Several License Plate Recognition systems have been developed in the past. Our objective is to design a system implemented on a standard camera-equipped mobile phone, capable of recognising vehicle license number. As a first step towards it we propose a license plate text segmentation approach that is robust to various lighting conditions, complex background owing to dirty or rusted LP and non-convential fonts. In the Indian scenario, some vehicle owners choose to write their vehicle number plates in regional languages. Since our method does not rely on language-specific features, it is therefore capable of segmenting license number written in different languages. Using color connected component labeling, stroke width and text heuristics we perform the task of accurately segmenting the number from the license plate.**

Experiments carried out on Indian vehicle license plate (LP) images acquired using a camera-equipped cellphone shows that our system peforms well on different LP images some with different types of degradations. OCR evaluation on the extracted LP number text with the proposed method has an accuracy of 98.86%.

Keywords- Vehicle License Number Segmentation, Text Segmentation, Mobile Phone based systems, Color Connected Component Labeling

I. Introduction

Automatic License Number Plate Systems have existed for several years now. Our aim however, is to work towards a system implemented on a standard camera equipped cellphone. Such a system may allow the traffic police to enforce road laws more easily. Significant cost reductions can be made due to the elimination of specialized hardware. A complete system may work as follows: The user needs to just click the scene with the vehicle in it. The software on the cellphone proceeds to establish a connection with a central server using the internet and sends the image for processing. The software on the remote server first locates the license plate(LP) using any standard plate-detection algorithm. Subsequently, it segments the characters using our proposed method and recognises the characters. The number can then be stored in a database or sent back to the client cellphone for other uses. This improves traffic law enforcement and helps achieve a paperless system where all information is processed and stored electronically on a central server. An alternative may also be to implement the entire processing software on the cellphone itself, eliminating the need for any connection to a remote server. An extension to this use can be smart mobile phone software that can read text from signboards or a restaurant menu, etc. and thereby assist blind users or foreigners to understand the written content by use of appropriate text-to-speech or translation software.

Thus, as a first step towards such a system, we focus on robust license number segmentation. Our method works well even on degraded images, low resolution mobile camera acquired images; images with poor lighting, complex license plate background, non-conventional character fonts and different languages. Thus, it can also be extended to segmentation of text in any kind of natural scene image.

Most other techniques use low-level techniques like morphology [1], histogram projection [2] and the hough transform [3] for segmenting numbers on the license plate. They make use of the structural features of the LP characters including regularity of character spacing, number of characters, and fixed direction. Some work also make use of geometric and spatial information. In [4] use of a two-layer Markov network has been made to integrate segmentation and recognition into one system. However, most of them assume text with high contrast on the license plate. Further, very few, if any, studies are available regarding segmentation of characters from LP images acquired with a cellphone camera.

In the context of the utility of robust segmentation and extraction of license number text for character recognition purposes, a large number of methods have been presented to identify the segmented characters. Approaches include those using neural networks [5] and template matching[6]. High accuracy rates were reported by them if they receive well segmented character images as input.

We propose a novel method, based on color connected component labeling, character stroke width and color similarity to segment the LP text. It is primarily based on a novel method for natural scene text segmentation method proposed in one of our previous works[7]. The clear benefit of this method over others is its' robustness in segmenting characters from degraded LP images, low contrast, rusted or damaged LPs etc.

Further, the results on images acquired using a cellphone camera yields very promising results, thus proving the efficacy of the system for possible implementation at a later stage on a standard mobile device.

II. System Description

In India, vehicle owners often use non-convential fonts and plate colors for their license plates. Although Indian vehicle ownership laws specify a particular format for the license plate text, the rule is not strictly followed nor enforced. We may find license plates with both one or two lines of text. Secondly, due to natural degradations like lighting conditions while acquisition of the LP image, poor physical condition of the LP, etc. the LP number extraction process becomes challenging.

We acquire images using two mobile phones equipped with a 2MP camera. Further details about the LP images dataset is given in section III. The license plate region in each image is detected using a standard plate detection algorithm [8] and the detected region is cropped for license number segmentation.

A. License Number segmentation

Our license number segmentation method consists of two stages. In the first stage, color segmentation is performed at the pixel-level using a novel color connected component labeling technique. In the second stage, we employ stroke and text color information to identify the license number text components.

1) Color connected component labeling: The most common RGB color format is not suitable for color grouping tasks because it is not expressed in the way perceived by humans. The color difference between two points with the same Euclidean distance in the RGB color space does not reflect the same change in the perceived color. So, we use a uniform color space, namely CIE L*a*b*, in which similar changes in color distance also correspond to similar recognizable changes in the perceived color. Equations for conversion from RGB to this format can be found in [9]

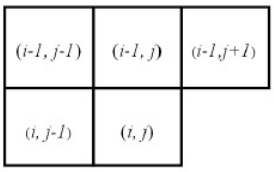


Fig. 1. The four neighbors of the pixel at (i; j) used in color connected component labeling

The CC labeling algorithm scans an image and groups the pixels into components based on pixel connectivity, i.e. all pixels in connected component share similar pixel intensity values and are in some way connected with each other. The image is scanned pixel by pixel from left to right and top to bottom. Let (i; j) denote the pixel at any step in the scanning process. We consider its two upper diagonal neighbors and the left and upper neighbors as shown in Fig. 1. The nature of the scanning process is such that these neighbors have already been processed by the time the procedure gets to pixel (i; j). The color distances of the four neighbors from the pixel at (i; j) is computed and our labeling algorithm proceeds as follows:

Algorithm 1 Color CC Labeling Algorithm

1. If none of the neighbors have a color distance smaller than a predefined threshold value T_c, assign a new label to pixel (i; j).
2. If only one of the neighbors has a color distance smaller than T_c, assign its label to pixel (i; j).
3. (a)If two or more neighbors have a color distance smaller than T_c, the pixel at (i; j) is assigned the label of the one that has the least color distance.
(b)All the pixels in the image that have the same label as that of the other neighbor pixels are then reassigned with the same label given to pixel (i; j).

The distance between the colors $C_1 = (L_1^*; a_1^*; b_1^*)$ is computed using the Euclidean distance as follows:

$$Dist(C_1; C_2) = \sqrt{(L_1^* - L_2^*)^2 + (a_1^* - a_2^*)^2 + (b_1^* - b_2^*)^2} \quad (1)$$

The threshold parameter T_c decides the similarity between two colors and is set to a value 8 after experimentation. It is observed that in some license plate (LP) images that have poor contrast, the color transition at the edges are not sharp and hence the neighbor pixels belonging to different labels can satisfy the color threshold. This condition leads to merging of previously correctly labeled CCs with the background CC. To avoid this problem, we check whether the current pixel is an edge pixel or not. The re-alignment step (step 3(b)) of the algorithm is skipped whenever the current pixel belongs to the edge image. Unlike conventional CC labeling methods that store equivalent label pairs into equivalent classes and perform re-assignment through a second scan, replacing each label by the label assigned to its equivalent class, our approach requires only a single scan through the image. Therefore, our method depends only on the license number plate image size and not on the number of CCs.

2) Stroke Width Estimation: Characters in a license plate usually have uniform stroke width and color. Based on the observation that the dominant stroke width in the entire cropped license plate image normally corresponds to the character stroke width, we seek to estimate the dominant stroke width in the labeled image. Treating all components whose stroke widths are reasonable close to the estimated dominant stroke width as possible text candidates, we use the mean color of each CC for final character extraction step.

The stroke width of each CC is estimated using the edge information. At each pixel (i; j) in the edge map, the gradient angle is first computed. It may be observed here that the orientation of the stroke will generally be in a direction roughly perpendicular to the gradient angle. Let this angle be denoted by $\theta(i; j)$. Also, to accommodate the case of darker text on brighter neighboring background and viceversa the stroke direction could also be in the direction $-\theta(i; j)$. Subsequently, we start traversal in the direction $\theta(i; j)$ until we reach another edge pixel. When an edge pixel is found, we check for an angle condition to establish the validity of the traversed stroke segment. A valid stroke-ending edge pixel should have a gradient direction roughly opposite to that of the stroke-starting edge pixel. Thus, we treat the computed stroke segment as valid only if the gradient angle of the stroke-ending pixel is within the range of $\pm\pi/8$ radians from that of the stroke-starting pixel. The range is fixed based on extensive experiments performed. If this condition is satisfied, the length of the stroke segment is assumed to be the stroke width of the component. Now, all pixels in the stroke segment are assigned this particular stroke width, unless it had been previously assigned a smaller stroke width. This process is repeated in the direction $-\theta(i;j)$.

(a) The output of color CC labeling (b) Edge image (c) Stroke width image from which the dominant stroke width is computed

Fig. 2. Stroke Width Estimation Overview

This opposite angle condition is normally valid in text CC since they have closed contours and is effective in removing spurious stroke segments. However, after the first pass, complex areas of the character pixels, especially at the corners may not contain the true stroke width. Thus, we make a second pass where process of stroke traversal is repeated again. However, this time we compute the median stroke over all the pixels of each valid stroke segment; and replace each pixel of that segment with this median value unless it already contains a smaller stroke width value. A few spurious stroke widths may still be present but since we make use of the dominant stroke width of the components, our results are not affected by such minor inconsistencies. Figure 2 shows the corresponding stroke width image and edge map for a particular component. Our observation is that this process results in a robust estimation of the stroke width of every component.

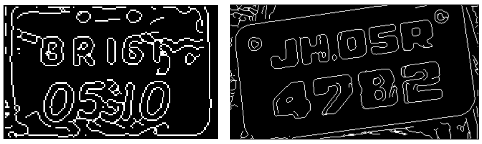
3) Final License Number Extraction: For LP number text components, we make use of the fact that the stroke width will be uniform to a large extent. Once the stroke width of the the labeled CC is obtained, we compute a dominant stroke width, SW_{dom} over all those components.

$$SW_{dom} = median(SW) \tag{2}$$

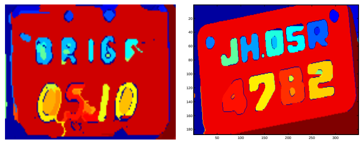
where SW is the set of valid stroke widths. Since, we are dealing with only LP images, the global dominant stroke width will be very close to that of a LP text component. However, the individual stroke widths may differ slightly across different characters. Thus, all the stroke widths having a frequency of occurrence greater than 50% of the global dominant stroke frequency are considered. The standard deviation, SW_{dev} of these selected set of stroke widths is computed. Potential LP text components are identified as those CCs whose stroke width lie within ($SW_{dom} \pm 0.5 \times SW_{dev}$) since the stroke width of the characters in a LP is regular.

We compute the distance D_{color} of the mean color of these potential LP text CCs from the mean color of the pixels whose stroke width is equal to SW_{dom}. Assuming that the characters are of uniform color, we filter out CCs having a value of D_{color} greater than a threshold which is set to 20 after experimentation. The final segmented LP text CCs are those that remain after the stroke width and color-based filtering.

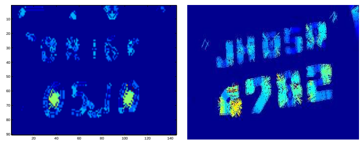
(a) Original LP Image



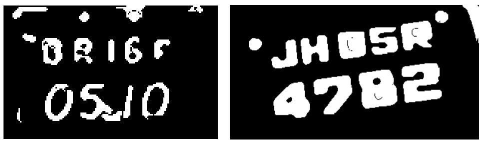
(b) Corresponding Edge image



(c) Output of color-based CC labeling



(d) Stroke width of CCs computed from the labeled images



(e) Potential license number text CCs based on stroke width information

(f) Final extracted LP number text CCs after color filtering

Fig. 3. Results of the intermediate processing steps.

III. Experiments and Results

Images of vehicle license plates were acquired using a LG KP500 and Nokia 5233 phone both equipped with a 2MP camera. We used 85 images randomly collected from parked vehicles at different locations clicked under unconstrained conditions.

TABLE I

PERFORMANCE EVALUATION OF PROPOSED METHOD BY CONDUCTING OCR EVALUATION ON RAW LP IMAGES AND OUTPUTS FROM THE PROPOSED LP NUMBER TEXT EXTRACTION METHOD.

	Total Number of LP Characters Tested	Levenshtein distance (LD) on the raw input LP images	LD on the output from the proposed method
	790	469	9
Percentage Accuracy		40.63	98.86
The proposed method is implemented using MATLAB v. 7.9 and tested on a laptop PC with an Intel Centrino Duo processor-1.5GHz CPU/4 GB DDR2 RAM			

These images include some with various kinds of degradations such as uneven lighting conditions, complex backgrounds owing to rusting of LP, dirty/muddy LP or shadows, etc., variable fonts, and one or two lined text.

These images contain a total of 790 LP number characters. Small images are resized using bi-cubic interpolation so that it has a height of at least 70 pixels while maintaining the aspect ratio. This is done so as to ensure that the edges are properly detected for low resolution LP text.

The results of color-based CC labeling and subsequent license number text CC extraction from the stroke width and color information are shown in Figure 3.

Some example outputs of the proposed LP number text extraction method are shown in Figure 4.

Fig. 4. Outputs from the proposed method

To illustrate the efficacy of our method, we evaluate the segmented LP number text using Nuance Omnipage Professional 16 (trial version) OCR software. Since, LP number recognition is beyond the scope of the present work we used the OCR software to test the efficiency of successfully segmenting the LP number text with our method.

To quantify the OCR results, the Levenhstein distance [10] between the ground truth (raw LP images) and the OCRed text is computed and listed in Table 1. The lower value of LD implies less number of OCR errors. The results of OCR on the raw input images without using the proposed method is 40.63% which markedly increases to 98.86% when it is fed with the processed outputs from our method.

IV. CONCLUSIONS

In this paper, we primarily deal with the segmentation and extraction of license plate characters using a novel technique. Using color connected component labeling, stroke width and color information we achieve promising results.

We evaluate the proposed algorithm on real-world license plate images captured using mobile phone cameras. The experiments show promising results even on degraded images where connected and distorted characters, rusted or muddy/dirty LPs, shadows or uneven lighting conditions are present. Since, we do not make use of any language specific or country specific LP information for our method, the work can be extended to LPs from other countries or in non-English LPs as demonstrated in one of the output (see vehicle license plate with Tamil characters in Figure 4).

A. *Contribution towards Green Computing*

This work is a first step towards a system fully implemented on a cellphone that can be used for traffic management and traffic law enforcement. Traffic policemen will not need to use paper for noting down LP numbers of law-breaking vehicles. A centralized server can store the LP numbers electronically removing the need for paper on a larger scale. Other possible uses of such a system in contributing towards a greener world are possible and only limited by our imagination.

B. *Future Work*

In our future work we intend to implement this system using Symbian C++ on a Symbian OS based mobile phone. Thereafter, we will design a complete system that will acquire, process and recognise vehicle LP numbers using just a cellphone and our software installed on it.

REFERENCES

[1] S. Nomura, K. Yamanaka, O. Katai, H. Kawakami, and T. Shiose, “A novel adaptive morphological approach for degraded character image segmentation,” *Pattern Recognition*, pp. 1961–1975, 2005.

[2] S. Zhang, M. Zhang, and X. Ye, “Car plate character extraction under complicated environment,” in *Proc. of 2004 IEEE Intl. Conf. on Systems, Man and Cybernetics*, 2004, p. 47224726.

[3] Y. Zhang and C. Zhang, “A new algorithm for character segmentation of license plate.”

[4] X. Fan, G. Fan, and D. Liang, “Joint segmentation and recognition of license plate characters.”

[5] S. L. Chang, L. S. Chen, Y. C. Chuang, and S. W. Chen, in *Automatic license plate recognition*, 2004, pp. 42–53.

[6] S. Belongie, J. Malik, and J. Puzicha, in *Shape matching and object recognition using shape contexts*, 2002, pp. 509–522.

[7] T. Kasar, Amey Dharwadker, Abhishek Sharma, and A. G. Ramakrishnan, “Textracc: Robust text extraction using nonlinear color smoothing and stroke width,” 2010.

[8] V. Abolghasemi and A. Ahmadyfard, “An edge-based color-aided method for license plate detection,” *Image and Vision Computing*, pp. 1134 – 1142, 2009.

[9] A. Koschan and M. Abidi, *Digital Color Image Processing*, 2008.

[10] V. I. Levenshtein, “Binary codes capable of correcting deletions, insertions, and reversals.” in *Soviet Physics Doklady*, 1966.