ICIC Express Letters
Volume **10**, Number **4**, April **2016**

ICIC International ©2016 ISSN 1881-803X
pp. 883–887

A PROJECTION-BASED FEATURE EXTRACTION ALGORITHM FOR LICENSE PLATE RECOGNITION

Cheng-Yu Yeh and Yi-Cheng Chen

Department of Electrical Engineering
National Chin-Yi University of Technology
No. 57, Sec. 2, Zhongshan Rd., Taiping Dist., Taichung 41170, Taiwan
cy.yeh@ncut.edu.tw; a54917@yahoo.com.tw

Received October 2015; accepted January 2016

Abstract. *License plate recognition (LPR) system mainly involves a license plate localization technique, a character segmentation strategy and a plate character recognition technique, among which, the character recognition plays a key role to a high recognition rate for LPR system. This paper presents an efficient feature extraction approach in an attempt not merely to well maintain a high recognition rate but also to reduce the computational load considerably. Both the horizontal and the vertical projections of a divided character are treated as the feature parameters in this work. For a character divided into $M \times N$ blocks, the dimension of a feature vector is reduced from $M \times N$ in a conventional approach to $M + N$ in this work. As it turns out, there is a 91% computational load reduction using a combination of a k-means clustering algorithm and this proposal in the case of ($M = 36$, $N = 16$), and a comparable recognition rate is demonstrated as well. In other words, the use of this algorithm enables a handheld device to meet the energy saving requirement for extended operations.*

Keywords: Character recognition, License plate recognition, Feature extraction, k-means clustering algorithm

1. **Introduction.** As the name indicates, character recognition refers to the recognition of printed text characters by a computer, and it is commonly applied to license plate recognition (LPR) [1,2] accordingly. Today, broad applications of LPR techniques can be found in intelligent transportation systems [3,4], parking space and vehicle access control systems, automated toll collection systems, street monitoring systems, and many more.

Most LPR techniques are three-stage algorithms, involving a license plate localization [5] followed by a character segmentation [6] and then a character recognition [3,7-10]. License plate localization refers to an image processing that is able to accurately locate a license plate from a complicated background, character segmentation refers to an image processing that segments the entire plate into individual characters, and finally character recognition refers to an image processing that converts individual character images into text characters based on extracted features. An LPR is done accordingly.

Character recognition is found to be a key factor in the recognition rate of LPR. Most character recognition algorithms can be categorized into two types in terms of the way it is performed, i.e., template matching-based [7] and classifier-based [3,8-10] algorithms. In template matching-based algorithms, a segmented character must be resized to the size of a template. Subsequently, template matching is performed between a piece of raw data, or pixel values, and each of the templates. In classifier-based algorithms, a set of feature parameters, often referred to as a feature vector, of a segmented character is extracted firstly, following which classifier-based character recognition is performed using neural networks [8] or hidden Markov model (HMM) [10]. Though character recognition can be performed in an efficient manner using a typical template matching-based algorithm, a major disadvantage is the absence of robustness, meaning that merely characters of specific type can be recognized. In contrast, a high robustness and an improved recognition rate are provided by a typical classifier-based algorithm, due to which it is treated as the most common approach to character recognition.

Other than a required high recognition rate, it is requested that LPR is performed in an efficient manner for the sake of real-time operations, particularly when performed on embedded systems or handheld devices. Hence, it is mandatory that the computational complexity is reduced to meet the energy saving requirement for extended operation of handheld devices.

In short, this paper presents a novel and particularly time efficient feature extraction algorithm. In other words, it requires a much smaller number of extracted features than a conventional counterpart, namely a significant computational load reduction, while a high recognition rate is well maintained. With extracted features, LPR is done using a k-means clustering algorithm, and the performance superiority is well validated by experimental means at the end of this work.

The rest of this paper is outlined as follows. Section 2 refers to a conventional feature extraction algorithm, Section 3 details the novel feature extraction approach, Section 4 discusses experimental results, and Section 5 draws a conclusion in the end.

2. **Conventional Feature Extraction Algorithm.** Typically, the input image of character recognition is a binarization but different sized image. As illustrated in Figure 1, a conventional LPR process [9] segments a character into $M \times N$ blocks, where M and N both are integers. Subsequently, feature extraction is performed corresponding to each block in turn, and then a feature vector is composed of all the individual features, based on which a character recognition can be done. In this manner, a feature vector of dimension $M \times N$ is yielded regardless of the size of characters, meaning the size problem can be fully removed.

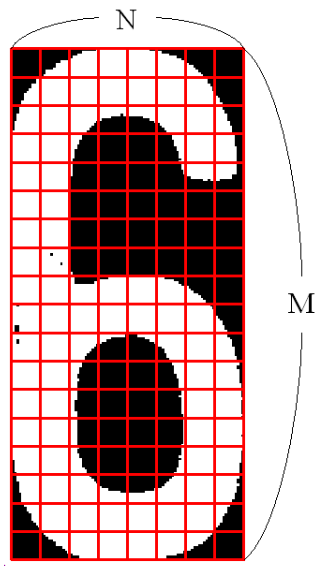

FIGURE 1. An illustration of conventional character feature extraction

In conventional approaches, the feature corresponding to a block is defined as the mean of the pixel values contained therein, that is,

$$\boldsymbol{V}(l) = \frac{1}{255 \times \frac{H}{M} \times \frac{W}{N}} \sum_{i=0}^{H-1} \sum_{j=0}^{W-1} Image(i,j) \Bigg|_{i \in b_l, j \in b_l}, \quad 0 \leq l < M \times N \tag{1}$$

where $\mathbf{V}(l)$ represents the feature corresponding to block l, namely the l-th component of the feature vector, H and W respectively denote the height and the width of the character, M and N respectively denote the number of equal blocks along the horizontal and vertical directions, $Image(i, j)$ symbolizes the value of the pixel in the i-th row and the j-th column, and both i, j belong to block b_l. The mean value is evaluated as the sum of the pixel values over b_l divided by the block area $(H/M) \times (W/N)$, and is further divided by a factor of 255 for normalization. Consequently, the normalized mean value lies between 0 and 1.

3. **Proposed Feature Extraction Algorithm.** This proposal is essentially a simplified version of conventional approaches in terms of computational load required in feature extractions. As stated previously, the mean value of pixels contained in each block must be evaluated in a conventional approach, while instead this proposal merely requires to evaluate the mean value in each row or column. In other words, the feature in each column/row is defined as the ratio of the number of pixels occupied by a character and the total number of pixels contained in the column/row, i.e., the column/row projection.

The concept is illustrated in Figure 2. Just as in Figure 1, a character is equally divided into M segments in the vertical direction. Subsequently, a feature extraction is performed on a row, say row 4 highlighted in boldface in Figure 2. Indicated on the left side of the character is a plot of the extracted feature versus block No. The row-based feature $\mathbf{V}_h(l)$ corresponding to b_l is defined as

$$\boldsymbol{V}_h(l) = \frac{1}{255 \times \frac{H}{M} \times W} \sum_{i=0}^{H-1} \sum_{j=0}^{W-1} Image(i, j) \Big|_{i \in b_l}, \quad 0 \leq l < M \tag{2}$$

As can be seen, the block b_l, highlighted in boldface, is a rectangle measuring (H/M) and W. Therefore, it is divided by $(H/M \times W)$, the area of b_l, and further divided by a factor of 255 for normalization.

Similarly, the character in Figure 2 is equally divided into N blocks along the horizontal direction. Subsequently, column-based feature extraction, say block 3 highlighted in boldface, is performed in turn, and the result is plotted versus block No. beneath the character. As demonstrated, there are a total of N features. The column-based feature

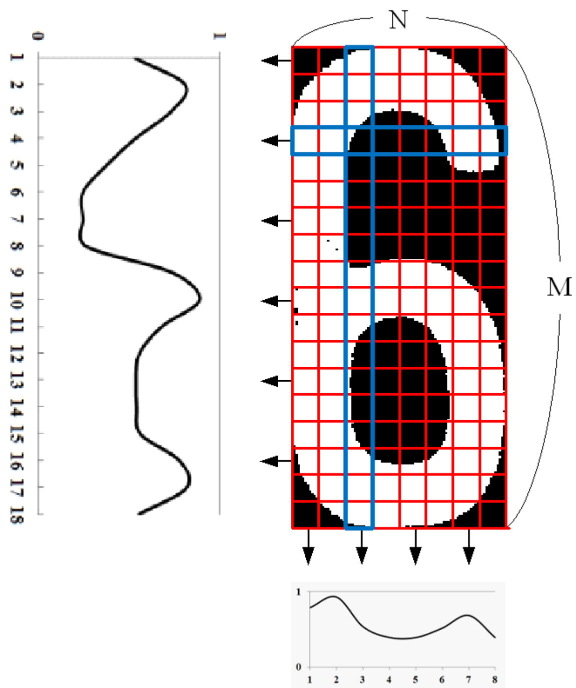

Figure 2. An illustration of projection-based character feature extraction

$\mathbf{V}_v(l)$ corresponding to b_l is defined as

$$\boldsymbol{V}_v(l) = \frac{1}{255 \times H \times \frac{W}{N}} \sum_{i=0}^{H-1} \sum_{j=0}^{W-1} Image(i,j)\Bigg|_{j \in b_l}, \quad 0 \leq l < N \tag{3}$$

Unlike in the row-based case, the block b_l, highlighted in boldface, is a rectangle measuring H and (W/N). Hence, it is divided by the block area $(H \times W/N)$ for the pixel mean value, and is further divided by a factor of 255 for normalization.

In this proposal, a combined collection of the normalized column and row-based features are treated as the resultant feature vector, a vector of length $M + N$ for a character. As stated previously, a conventional approach requires a feature vector of dimension $M \times N$, meaning that the aim of computational load reduction can be reached, particularly for large values of M and N when a high recognition rate is required.

4. **Experimental Results.** Table 1 gives complexity comparison, including addition and multiplication according to (1)-(3), between this proposal and a conventional counterpart, when feature extractions are performed on a character. As in previous equations, M and N represent the number of divisions that occur along the height and the width of a character, respectively. It is clearly seen that a tremendous reduction in multiplication complexity is achieved by the presented algorithm as intended.

Table 1. Comparison on multiplication and addition complexities between this proposal and a conventional counterpart

Feature extraction method	Multiplication	Addition
Conventional method (dimension = $M \times N$)	$M \times N$	$H \times W - M \times N$
Proposed method (dimension = $M + N$)	$M + N$	$H \times W + M \times N - (M + N)$

Furthermore, a k-means clustering algorithm is employed for character recognition, and then Table 2 gives a comparison on the computational load with K representing the number of clusters. A k-means clustering algorithm is performed based on Euclidean distance, defined as the distance between an input character image and each cluster center. For illustration purposes, the performance superiority is demonstrated by two cases ($M = 18$, $N = 8$) and ($M = 36$, $N = 16$). In the case of ($M = 18$, $N = 8$), this proposal requires as low as 18.06% of the multiplication complexity required in a conventional counterpart, that is, a 81.94% computational saving, while a computational saving up to 90.97% is provided by this presented algorithm in the case of ($M = 36$, $N = 16$). Thus, this proposal is experimentally validated as a superior feature extraction algorithm in terms of computational load, particularly for large values of M and N.

Table 2. Complexity comparison using a k-means clustering algorithm based on the results listed in Table 1

Feature extraction method		Multiplication	Addition
Conventional method		$M \times N \times K$	$(2 \times M \times N - 1) \times K$
Proposed method		$(M + N) \times K$	$(2 \times (M + N) - 1) \times K$
Computational saving	$M = 18$, $N = 8$	81.94%	82.23%
	$M = 36$, $N = 16$	90.97%	91.05%

Table 3 gives a comparison on the recognition rate among various combinations of M, N and k values in a k-means clustering algorithm. The comparison is made with 2146 non-equal sized character images, 1786 of which are chosen as the training samples, and

Table 3. Recognition rate comparison among various cases

Block No.	K-means clustering with different number of clusters	Recognition rate (%)	
		Conventional method	Proposed method
$M = 18$, $N = 8$	$K = 1$	99.49	98.11
	$K = 2$	99.54	98.29
	$K = 4$	99.56	98.33
$M = 36$, $N = 16$	$K = 1$	99.72	98.12
	$K = 2$	99.82	98.29
	$K = 4$	99.82	98.36

the remaining 360 are chosen as the test samples. It is found that a comparable as well as high recognition rate can be maintained by this proposal among all the testing cases. In short, the aim of significant computational load reduction is achieved at the cost of marginal drop in the recognition rate. As presumed, large values of (M, N) lead to an improved recognition rate at the cost of a low rise in the computational load, a clear advantage of this proposal over conventional feature extraction algorithms.

5. **Conclusion.** This paper presents a projection-based feature extraction algorithm. A combined use of this novel algorithm and a k-means clustering algorithm provides an up to 91% computational load reduction relative to a conventional counterpart in the case of $(M = 36, N = 16)$, while a high recognition rate is well maintained meanwhile. In other words, the use of this algorithm enables a handheld device to meet the energy saving requirement for extended operations. As scheduled, a continuous effort will be made on the robust recognition for a variety of harsh environment.

REFERENCES

[1] S. Du, M. Ibrahim, M. Shehata and W. Badawy, Automatic license plate recognition (ALPR): A state-of-the-art review, *IEEE Trans. Circuits Syst. Video Tech.*, vol.23, no.2, pp.311-325, 2013.

[2] S. L. Chang, L. S. Chen, Y. C. Chung and S. W. Chen, Automatic license plate recognition, *IEEE Trans. Intell. Transp. Syst.*, vol.5, no.1, pp.42-53, 2004.

[3] H. Peng, X. Wang, H. Wang and W. Yang, Recognition of low-resolution logos in vehicle images based on statistical random sparse distribution, *IEEE Trans. Intell. Trans. Syst.*, vol.16, no.2, pp.681-691, 2015.

[4] J. Pirc, G. Turk and M. Zura, Using the robust statistics for travel time estimation on highways, *IET Intell. Trans. Syst.*, vol.9, no.4, pp.442-452, 2015.

[5] S. Yu, B. Li, Q. Zhang, C. Liu and M. Q. H. Meng, A novel license plate location method based on wavelet transform and EMD analysis, *Pattern Recognition*, vol.48, no.1, pp.114-125, 2015.

[6] J. M. Guo and Y. F. Liu, License plate localization and character segmentation with feedback self-learning and hybrid binarization techniques, *IEEE Trans. Veh. Tech.*, vol.57, no.3, pp.1417-1424, 2008.

[7] P. Comelli, P. Ferragina, M. N. Granieri and F. Stabile, Optical recognition of motor vehicle license plates, *IEEE Trans. Veh. Tech.*, vol.44, no.4, pp.790-799, 1995.

[8] H. Caner, H. S. Gecim and A. Z. Alkar, Efficient embedded neural-network-based license plate recognition system, *IEEE Trans. Veh. Tech.*, vol.57, no.5, pp.2675-2683, 2008.

[9] F. Aghdasi and H. Ndungo, Automatic license plate recognition system, *Proc. of IEEE AFRICON Conf. in Africa*, Gaborone, Botswana, pp.45-50, 2004.

[10] D. Llorens, A. Marzal, V. Palazon and J. M. Vilar, Car license plates extraction and recognition based on connected components analysis and HMM decoding, *Lecture Notes Comput. Sci.*, vol.3522, pp.571-578, 2005.