Pattern Recognition 42 (2009) 358–369

Contents lists available at ScienceDirect

Pattern Recognition

journal homepage: www.elsevier.com/locate/pr

A configurable method for multi-style license plate recognition

Jianbin Jiao, Qixiang Ye*, Qingming Huang

Graduate University of Chinese Academy of Sciences, No. 19A, Yuquan Road, Beijing 100049, PR China

ARTICLE INFO

Article history:
Received 12 July 2007
Received in revised form 7 July 2008
Accepted 13 August 2008

Keywords:
License plate recognition
Multi-style license plate
Configurable method
Character segmentation
Character recognition

ABSTRACT

Despite the success of license plate recognition (LPR) methods in the past decades, few of them can process multi-style license plates (LPs), especially LPs from different nations, effectively. In this paper, we propose a new method for multi-style LP recognition by representing the styles with quantitative parameters, i.e., plate rotation angle, plate line number, character type and format. In the recognition procedure these four parameters are managed by relevant algorithms, i.e., plate rotation, plate line segmentation, character recognition and format matching algorithm, respectively. To recognize special style LPs, users can configure the method by defining corresponding parameter values, which will be processed by the relevant algorithms. In addition, the probabilities of the occurrence of every LP style are calculated based on the previous LPR results, which will result in a faster and more precise recognition. Various LP images were used to test the proposed method and the results proved its effectiveness.

1. Introduction

Automatic license plate recognition (LPR) has been a practical technique in the past decades [1]. Numerous applications, such as automatic toll collection [1], criminal pursuit [2] and traffic law enforcement [3], have been benefited from it [1–10]. Although some novel techniques, for example RFID (radio frequency identification), WSN (wireless sensor network), etc., have been proposed for car ID identification, LPR on image data is still an indispensable technique in current intelligent transportation systems for its convenience and low cost.

LPR is generally divided into three steps: license plate detection, character segmentation and character recognition. The detection step roughly classifies LP and non-LP regions, the segmentation step separates the symbols/characters from each other in one LP so that only accurate outline of each image block of characters is left for the recognition, and the recognition step finally converts grey-level image block into characters/symbols by pre-defined recognition models.

Although LPR technique has a long research history, it is still driven forward by various arising demands, the most frequent one of which is the variation of LP styles, for example:

(1) Appearance variation caused by the change of image capturing conditions.
(2) Style variation from one nation to another.
(3) Style variation when the government releases new LP format.

We summed them up into four factors, namely rotation angle, line number, character type and format, after comprehensive analyses of multi-style LP characteristics on real data. Generally speaking, any change of the above four factors can result in the change of LP style or appearance and then affect the detection, segmentation or recognition algorithms. If one LP has a large rotation angle, the segmentation and recognition algorithms for horizontal LP may not work. If there are more than one character lines in one LP, additional line separation algorithm is needed before a segmentation process. With the variation of character types when we apply the method from one nation to another, the ability to re-define the recognition models is needed. What is more, the change of LP styles requires the method to adjust by itself so that the segmented and recognized character candidates can match best with an LP format.

Several methods have been proposed for multi-national LPs or multi-format LPs in the past years [11,12] while few of them comprehensively address the style adaptation problem in terms of the above-mentioned factors. Some of them only claim the ability of processing multi-national LPs by re-defining the detection and segmentation rules or recognition models.

In this paper, we propose a configurable LPR method which is adaptable from one style to another, particularly from one nation to another, by defining the four factors as parameters. Users can constrain the scope of a parameter and at the same time the method will adjust itself so that the recognition can be faster and more accurate. Similar to existing LPR techniques, we also provide details of LP detection, segmentation and recognition algorithms. The difference is

* Corresponding author. Tel.: +86 10 88256968.
E-mail addresses: jiaojb@gucas.ac.cn (J.Jiao), qxye@gucas.ac.cn (Q. Ye), qmhuang@gucas.ac.cn (Q. Huang).

0031-3203/$ - see front matter © 2008 Published by Elsevier Ltd.
doi:10.1016/j.patcog.2008.08.016

that we emphasize on the configurable framework for LPR and the extensibility of the proposed method for multi-style LPs instead of the performance of each algorithm.

The rest of this paper is organized as follows. Related work is reviewed in Section 2. The configurable method is proposed in Section 3. Implementation of the configurable method is presented in Section 4. Experimental results are provided and discussed in Section 5 and conclusions in Section 6.

2. Related works

In the past decades, many methods have been proposed for LPR that contains detection, segmentation and recognition algorithms. In the following paragraphs, these algorithms and LPR methods based on them are briefly reviewed.

LP detection algorithms can be mainly classified into three classes according to the features used, namely edge-based algorithms, color-based algorithms and texture-based algorithms. The most commonly used method for LP detection is certainly the combinations of edge detection and mathematical morphology [4–7]. In these methods, gradient (edges) is first extracted from the image and then a spatial analysis by morphology is applied to connect the edges into LP regions. Another way is counting edges on the image rows to find out regions of dense edges [8] or to describe the dense edges in LP regions by a Hough transformation [9]. Edge analysis is the most straightforward method with low computation complexity and good extensibility. Compared with edge-based algorithms, color-based algorithms depend more on the application conditions. Since LPs in a nation often have several pre-defined colors, researchers have defined color models to segment region of interests as the LP regions [10,13,14]. This kind of method can be affected a lot by lighting conditions. To win both high recall and low false positive rates, texture classification has been used for LP detection. In Ref. [15], Kim et al. used an SVM to train texture classifiers to detect image block that contains LP pixels. In Ref. [16], the authors used Gabor filters to extract texture features in multi-scales and multi-orientations to describe the texture properties of LP regions. In Ref. [17], Zhang used *X* and *Y* derivative features, grey-value variance and Adaboost classifier to classify LP and non-LP regions in an image. In Refs. [18,19], wavelet feature analysis is applied to identify LP regions. Despite the good performance of these methods the computation complexity will limit their usability [20]. In addition, texture-based algorithms may be affected by multi-lingual factors.

Multi-line LP segmentation algorithms can also be classified into three classes, namely algorithms based on projection [21–23], binarization [24–27] and global optimization [28]. In the projection algorithms, gradient or color projection on vertical orientation will be calculated at first. The "valleys" on the projection result are regarded as the space between characters and used to segment characters from each other. Segmented regions are further processed by vertical projection to obtain precise bounding boxes of the LP characters. Since simple segmentation methods are easily affected by the rotation of LP, segmenting the skewed LP becomes a key issue to be solved. In the binarization algorithms, global or local methods are often used to obtain foreground from background and then region connection operation is used to obtain character regions. In the most recent work, local threshold determination and slide window technique are developed to improve the segmentation performance [29]. In the global optimization algorithms, the goal is not to obtain good segmentation result for independent characters but to obtain a compromise of character spatial arrangement and single character recognition result. Hidden Markov chain has been used to formulate the dynamic segmentation of characters in LP [28]. The advantage of the algorithm is that the global optimization will improve the robustness to noise. And the disadvantage is that precise format definition is necessary before a segmentation process.

Character and symbol recognition algorithms in LPR can be categorized into learning-based ones and template matching ones. For the former one, artificial neural network (ANN) is the mostly used method [30–32] since it is proved to be able to obtain very good recognition result given a large training set. An important factor in training an ANN recognition model for LP is to build reasonable network structure with good features. SVM-based method is also adopted in LPR [15] to obtain good recognition performance with even few training samples. Recently, cascade classifier method is also used for LP recognition [2]. Template matching is another widely used algorithm [31,32]. Generally, researchers need to build template images by hand for the LP characters and symbols. They can assign larger weights for the important points, for example, the corner points, in the template to emphasize the different characteristics of the characters [29]. Invariance of feature points is also considered in the template matching method to improve the robustness. The disadvantage is that it is difficult to define new template by the users who have no professional knowledge on pattern recognition, which will restrict the application of the algorithm.

Based on the above-mentioned algorithms, lots of LPR methods have been developed. However, these methods are mainly developed for specific nation or special LP formats. In Ref. [1], the authors focus on recognizing Greek LPs by proposing new segmentation and recognition algorithms. The characters on LPs are alphanumerics with several fixed formats. In Ref. [2], Zhang et al. developed a learning-based method for LP detection and character recognition. Their method is mainly for LPs of Korean styles. In Ref. [14], optical character recognition (OCR) technique are integrated into LPR to develop general LPR method, while the performance of OCR may drop when facing LPs of poor image quality since it is difficult to discriminate real character from candidates without format supervision. This method can only select candidates of best recognition results as LP characters without recovery process. Wang et al. [10] developed a method to recognize LPR with various viewing angles. Skew factor is considered in their method. In Ref. [20] the authors proposed an automatic LPR method which can treat the cases of changes of illumination, vehicle speed, routes and backgrounds, which was realized by developing new detection and segmentation algorithms with robustness to the illumination and image blurring. The performance of the method is encouraging while the authors do not present the recognition result in multi-nation or multi-style conditions. In Ref. [11], the authors propose an LPR method in multi-national environment with character segmentation and format independent recognition. Since no recognition information is used in character segmentation, false segmented characters from background noise may be produced. What is more, the recognition method is not a learning-based method, which will limit its extensibility. In Ref. [12], Mecocci et al. propose a generative recognition method. Generative models (GM) are proposed to produce many synthetic characters whose statistical variability is equivalent (for each class) to that showed by real samples. Thus a suitable statistical description of a large set of characters can be obtained by using only a limited set of images [12]. As a result, the extension ability of character recognition is improved. This method mainly concerns the character recognition extensibility instead of whole LPR method.

From the review we can see that LPR method in multi-style LPR with multi-national application is not fully considered. Lots of existing LPR methods can work very well in a special application condition while the performance will drop sharply when they are extended from one condition to another, or from several styles to others.

3. Configurable LPR method

In this section, a configurable framework for LPR is proposed by defining the factors mentioned above as parameters. Then the methodology on how to process the parameters in order is presented.

3.1. Configurable framework with parameter definition

Since any change of the factors, namely rotation angle, line number, character type and character format, can cause the change of LP style or appearance, we define four parameters corresponding to these factors as $\Theta = \{\theta_0, \theta_1, \theta_2, \theta_3\}$.

3.1.1. Plate rotation angle (θ_0)

In some applications, the captured LPs can be skewed if the camera axis is not perpendicular to the license plate plane or keeps an angle with horizontal orientation. This angle is the rotation (angle between horizontal baseline and up boundary of an LP) of the license plate and does not encapsulate the slant or tilt of the plate. A rotation angle parameter is defined so that the captured skewed LP can be restored. LP rotation angle can take discrete values as $\theta_0 \in \{-5S, \ldots, 0, 5, \ldots, 5S\}$, with the step set to 5 degrees and maximum angle as $5S$ degrees. S is the step number. Generally speaking, $S = 6$ is enough for most of the applications.

3.1.2. Character line number (θ_1)

In some conditions, LPs are formatted as single line character group, while in others as multiple line characters. For example, the truck LPs in China usually contain two or three character lines. Therefore, we define $\theta_1 \in \{1, 2, 3, \ldots\}$ to represent the character line number. In an LPR process, if $\theta_1 > 1$, an additional line separation procedure will be called. In experiments we consider LPs of three lines at most.

3.1.3. Recognition models (θ_2)

Almost all LPs come with digits, alphabets, and sometimes, special symbols. LPs of different styles can contain different content. For example, some LPs have symbols and alphabets, and others may have digits, alphabets and special symbols. In different nations, symbols may be different. Thus we define a parameter θ_2 corresponding to specific recognition models:

$$\theta_2 = \begin{cases} 0 & \text{alphabet only} \\ 1 & \text{alphabet and digit} \\ 2 & \text{alphabet, digit and symbols} \end{cases} \tag{1}$$

If there are symbols in LPs, users can train the recognition models by themselves with the ANN training algorithm (which is developed into a training tool) in Section 4.2. This parameter can be either defined by a user or inferred from the following character format parameter.

3.1.4. Character formats (θ_3)

An LP is made up of characters (alphanumerics or symbols) that can form various formats by changing their arrangement. If we use "D" to represent a digit, "A" an alphabet and "S" a symbol, we can represent the jth format as $\theta_3^j = \{C_1 C_2, \ldots, C_i, \ldots, C_n\}$, where C_i represents the character type at ith position, which can be S, A, or D. For example, an LP with content "京CB6836" can be formatted as "SAADDDD". In one nation, there are usually several (generally less than 10) kinds of character formats defined by its government. Given a defined format then it can be decided what kind of recognition model should be used to recognize the character at the ith position exactly. This will avoid testing all of the possible recognition models and then improve the recognition efficiency and precision.

As a matter of fact, there are lots of factors that can decide various types of plate styles except for the above ones, which can be categorized into "internal" and "external" ones. For example, number of lines, character types, and formats, etc., are internal factors that are intrinsic to an LP. On the other hand, LP image capturing condition, i.e., illumination, rotation angle, LP size and location in an image, etc., are external factors that change with applications. All of "internal" and "external" elements can be considered in a configurable framework given support of implementation algorithms. In our present system, we consider only some of the most important ones since the support algorithms are limited. Number of lines, character types and formats are three most important internal elements that should be considered. Rotation angle is the most important factor that affects LP character segmentation and recognition. Other factors, like illumination, LP size, etc., can be easily solved by most of the existing LPR methods, and therefore, they are not considered in our system. At present we just proposed a framework for configuration. Factors and parameters can be extended in the future research.

Given defined parameters we propose the configurable framework for LPR as shown in Fig. 1. Like the traditional LP recognition method, there are three main steps, namely detection, segmentation and recognition in our method. These three steps can be decomposed into candidate detection, rotation, segmentation, recognition, format matching algorithms. In each LPR recognition process, we will coarsely detect some LP candidates in an image. Each candidate will be processed as the framework in Fig. 1 and the format matching finally classify it into an LP or a false positive.

In the LPR framework, the method should determine if the plate is skewed or not to make sure that if a restoration procedure is necessary. For the segmentation algorithm, the parameter of character line number will determine if a row separation procedure should be performed. And for image block containing single line LP characters a morphological operation on color quantification result will be taken out to obtain character candidates which is called a character group. For recognition, corresponding models will be selected for each position in the format. In the format matching a character group will be tested on all of the defined formats to find the best result. To sum up, the LPR will adjust itself so that the defined styles will be well recognized.

3.2. Processing configured styles in order

By defining the above-mentioned parameters, $\Theta = \{\theta_0, \theta_1, \theta_2, \theta_3\}$, we can describe the variation of LP styles as the parameter value changes, and then use these parameters to configure the recognition flow. It can be seen that the combination of four parameters can produce lots of LP styles. Given a located LP candidate we need to test it on all of the possible styles one by one if it is still unknown what the real style is.

In the four algorithms, testing all of the possible parameter values will decrease the performance of the method. For example, if a false format parameter is tested firstly in the format matching process, correct format may lose the chance to be tested, which will bring about false result. An intuitional way is to process the parameter values of high probability earlier and low probability later. Thus for parameter θ_i, $i = 0, \ldots, 3$, we can sort their possible values by occurrence probability descendently as $\{\theta_i^1, \theta_i^2, \ldots, \theta_i^{Ni}\}$ where N_i represents the value number of θ_i.

The probability of each parameter value can be obtained on statistics of history recognition results. Given the recognized M LPs as ground truth, the probability $P(\theta_i^j)$ for jth of parameter θ_i is

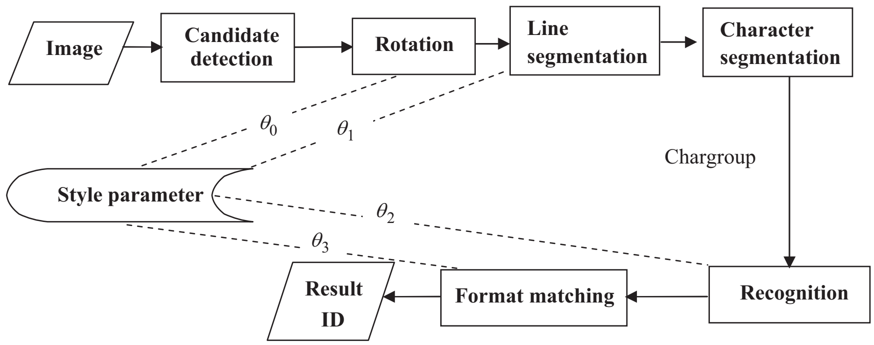


Fig. 1. LP recognition framework.

calculated by a non-parametric probability estimation method as

$$P(\theta_i^j) = \frac{1}{M}\sum_{m=1}^{M} K\left(\frac{\theta_i^m - \theta_i^j}{\sigma_{\theta_i}}\right) \quad (2)$$

where $K(x) = e^{-x^2}$, θ_i^m is the ith parameter value of sample m in the recognized results, σ_{θ_i} is the variance of $K(x)$ whose value is determined by experiments.

When the probability of each parameter is estimated, the probability of styles can be obtained by multiplying probability of each parameter together. The styles are then sorted by their probability descendently and in a recognition style of high probability are tested preferentially. Given one detected LP candidate the method will test few parameter values averagely to recognize it.

4. Implementation of the method

In this part, we present the algorithms for implementing the configurable LPR method. It should be mentioned that these algorithms are independent modules and can be substituted by similar other algorithms.

4.1. Detection, rotation and line segmentation

For a captured image with license plate, we assume that a region with dense vertical edges should be segmented as a plate candidate which is called a ROI (region of interest). In the location procedure, vertical Sobel edge features with spatial reduction are extracted first. Then a skeleton extraction algorithm [33] on edge map is carried out to ensure that single edge is extracted at one object boundary as shown in Fig. 2c.

It can be seen in Fig. 2c that in an LP region the edge is dense and repetitive, that is to say, a plate region is composed of a "cluster" of edge pixels. None but "dense" edge pixels can construct a text region and the isolated edge pixels are often noises. Morphological operation such as "close" operation is often used to connect text pixels into plate regions in the previous work. In the operation, all of the pixels close to each other will be connected despite whether they form a "cluster" of edge pixels or not. In this paper, we use our "density-based" region growing method [34] to fulfill this task.

A pixel P will be a seed pixel if the percentage of candidate pixels in its neighborhood is larger than a threshold T_D. The neighborhood is a $R \times R$ template determined by the size of LP. Generally speaking, given an image size, R can be set as 10–20% the height of the LP image. T_D can be adjusted in terms of image size, which is set as 0.10 empirically. A pixel P' is considered to be density connected with pixel P if P' is within the neighborhood of P and P is a seed pixel. By these definitions, the region growing method is described as follows:

(1) Search the unlabeled candidate pixels to find a seed pixel.
(2) If a seed pixel P is found, a new region is created. Then, we iteratively collect the unlabeled edge pixels that are density connected with P, and assign these pixels to the same region label.
(3) If there are still seed pixels, go to 1.
(4) Label each found region as a ROI.

Fig. 2d is an example of the text regions found by the "density-based" region growing algorithm. In these located regions there are lots of false positives that contain dense edges like a real LP region. To eliminate false positives we use the recognition and format matching result in the following sections.

In a located ROI a projection profile operation is used to separate multi-line LP characters. A horizontal projection profile is defined as the sum of the edge pixels over image rows as shown in Fig. 3a. For two character lines in an LP, we use a "valley" analysis method on the profile (as shown in Fig. 3a) to separate multi-line characters. On the profile we will find horizontal location where the profile value is less than a threshold T_P and then separate the lines at the "valley". T_P can be calculated by

$$T_P = (Avg_{profile} + Min_{profile})/2.0 \quad (3)$$

where $Min_{profile}$ and $Avg_{profile}$ are the minimum value and average value of the profile, respectively. This scheme can also be extended to character separation of more than two lines.

For skewed LP, an algorithm is developed to restore it into horizontal orientation, which will be performed in all the candidate regions. For an LP region, we can firstly calculate its height H by valid image row analysis. Image rows in which projection profile values larger than T_P are valid rows. By observation, we find that an LP will have least valid row number (H) if it is horizontal. Thus given a parameter $\theta_1 > 0$ we can rotate the original ROI step by step to find out an angle $\tilde{\theta}_1$ at which the valid row is minimized as shown in Fig. 3c, which can be used to restore the ROI.

4.2. Character segmentation and recognition

4.2.1. Character segmentation

In all of the ROIs we need to firstly obtain the precise bounding boxing of each character before recognition (as shown in Fig. 4b). This is character segmentation.

Firstly grey-level quantization and morphology analysis algorithm are used to obtain candidate characters. In this process, a 256 grey levels ROI image is linearly quantized into Q grey levels. Then a morphology "close operation" [33] is performed on the quantized

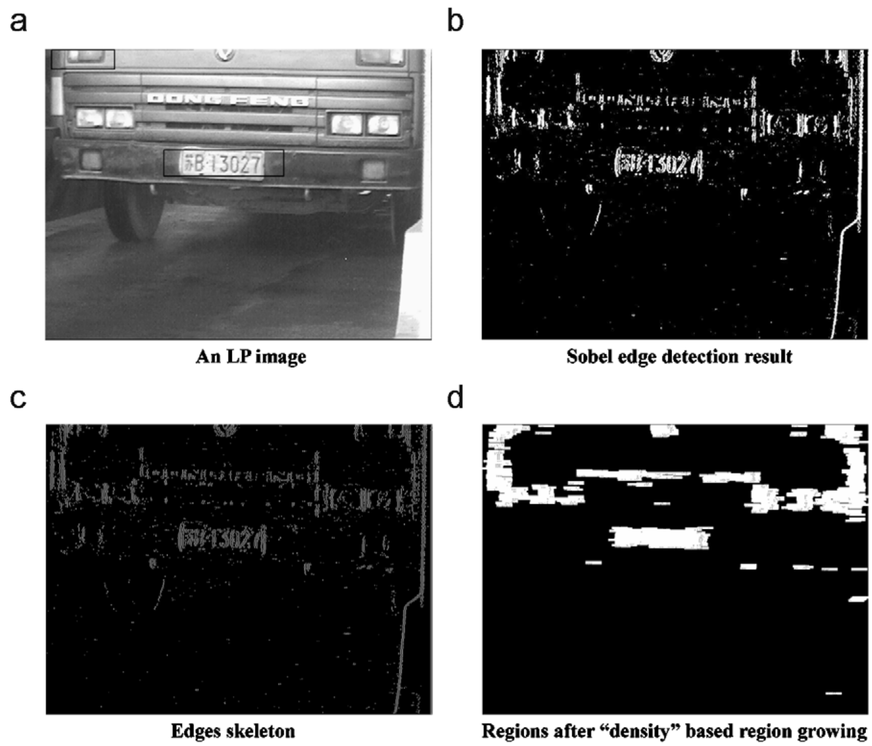


Fig. 2. Illustration of ROI location: (a) an LP image, (b) Sobel edge detection result, (c) edges skeleton, and (d) regions after "density" based region growing.

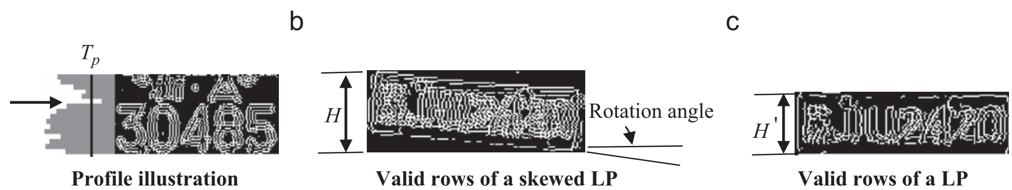


Fig. 3. Profile and skewed LP rotation illustration: (a) profile illustration, (b) valid rows of a skewed LP, and (c) valid rows of an LP.

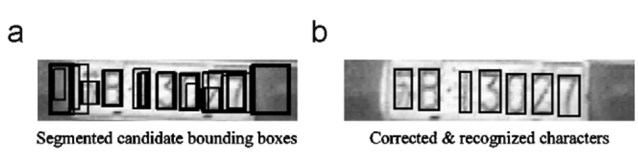


Fig. 4. Candidate character segmentation illustration: (a) segmented candidate bounding boxes and (b) corrected and recognized characters.

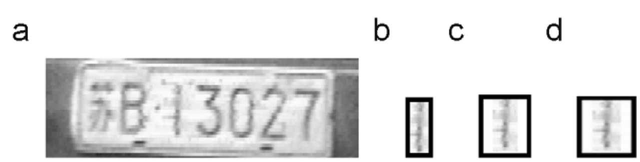


Fig. 5. Extend block boundary illustration: (a) a license plate, (b) segmented digit "1", (c) extend boundary of the character block, and (d) normalized character block.

images to obtain connected regions. The size of "close" operator is set as $(1, 0.2 \times H)$ pixels in X and Y orientations, respectively, where H represents the height of the ROI. To obtain as many candidates as possible, we adjust the parameter Q from 5 to 10 and then calculate all possible characters. The outlines of the final candidates are shown as Fig. 4a. To improve the segmentation performance and avoid missing characters we can also combine local binarization method [27] with the method mentioned above.

4.2.2. Character recognition

To convert segmented grey-value image blocks into characters, recognition models (digit recognition model, alphabet recognition model, and symbol recognition model) need to be built corresponding to the parameter θ_2. ANN is employed to train recognition models. The performance of ANN, given larger training set, has been proved by lots of pattern recognition applications. Another merit of ANN is that all users can train the recognition models for their special applications given in a training tool.

Each segmented character boundary is adjusted firstly if the width/height ratio of character block is not close to a constant (0.8 in our experiments), exactly, width/height is smaller than 0.5. We will extend the boundary of the block so that its weight/height ratio is equal to the constant. Fig. 5(b–d) is an example for segmented digit "1" and its extended block boundary. After this operation, segmented characters are firstly normalized into 16×16 image blocks by a linear interpolation algorithm.

Pixels' grey values of normalized block are extracted as the features. Fig. 6a–d are image blocks of 16×16 (in pixels). To emphasize

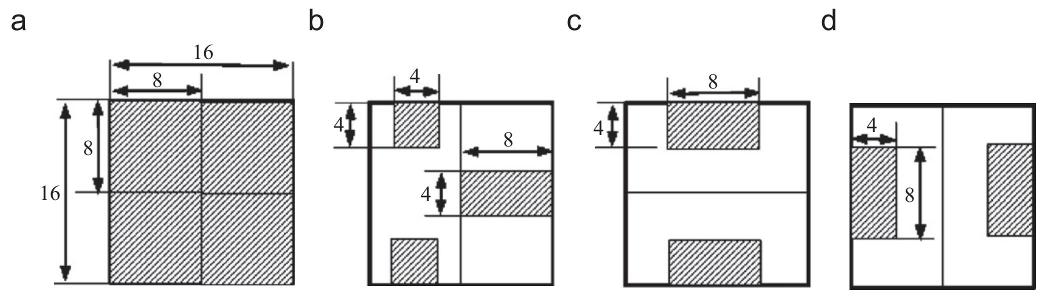


Fig. 6. Sub-blocks for recognition feature extraction.

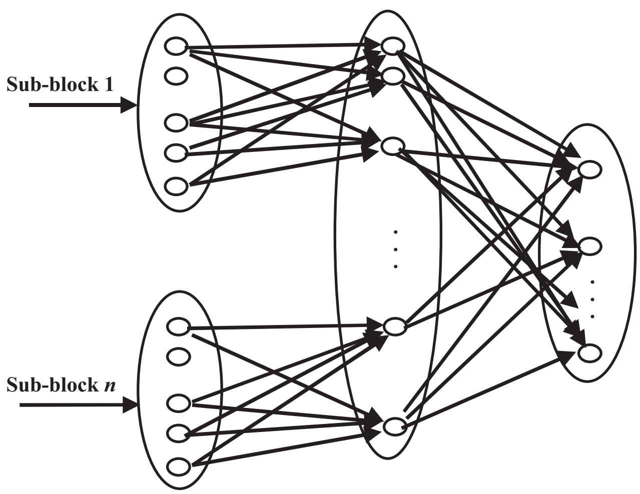


Fig. 7. ANN structure.

the characteristics of each character, 11 kinds of sub-blocks are defined as the shadow areas in Fig. 6a–d. In Fig. 6a, there are four 8×8 (in pixels) sub-blocks. In Fig. 6b, there are two 4×4 and one of 8×4 sub-blocks (in pixels). In Fig. 6c, there are two 8×4 sub-blocks. And in Fig. 6d, there are two 4×8 sub-blocks. The spatial locations of these sub-blocks are also shown in the figure. For each of the sub-block, pixel grey values, which are normalized to (0.0–1.0), are extracted as input nodes of the three-layer ANN. Input nodes from the same sub-block build connections with the same nodes in the hidden (middle) layer as shown in Fig. 7. This ANN structure can ensure that for a character some special sub-blocks could be emphasized by obtaining larger weights in the training process.

There are totally 448 input nodes in the network, constructed by the grey values of pixels in shadow areas shown in Fig. 6a–d. The node number of the middle layer is 1/10–1/5 of the input node number, which is determined by experiments. The output node number is equal to the class number, that is to say, for digit recognition model, the output number is 10, for alphabet recognition model, the output number is 26. For symbol recognition model, the output node number can be defined by the users.

When training the models, samples are collected by hand. For each alphanumeric, we marked more than 3000 samples to ensure the recognition performance. A back-propagation algorithm is used to train the ANN and a Sigmoid function [35] is selected as the transformation function between different ANN layers which is proved to perform best in experiments. The formula for the three layer ANN is described as

$$G_k(X) = f\left(\sum_{j=1}^{n} w_{kj} f\left(\sum_{i=1}^{d} w_{ji}x_i + w_{j0}\right) + w_{k0}\right) \tag{4}$$

where X is the input pixel value vector for classification. $G_k(X)$ is the final recognition result, f is the transformation function between nodes of different net layers, w represents connection weights, w_{ji} is a weight from input node to hidden node while w_{kj} is a weight from hidden node to output node. When there is no connection between two nodes, w is zero.

4.3. Format matching

Given an LP of format parameter θ_3, the method needs to match segmented and recognized candidate characters with the defined format. The matching is a compromise between single character recognition and character's spatial arrangement, which is formulated as a dynamic programming problem.

Character recognition outputs (response) from ANN are distributed from 0.0 to 1.0. For each of the candidate character we can calculate the probability to judge whether it is a real character by

$$P(G(X_i(t))) = \frac{1}{\sqrt{2\pi}\sigma}\exp\left(-\frac{(G(X_i(t)) - 1.0)^2}{2\sigma^2}\right) \tag{5}$$

where $G(X_i(t))$ is the response (ANN recognition result) of candidate i at the tth lattice of a format, $X_i(t)$ represents its spatial position as shown in Fig. 8a, σ is a standard variation determined by experiments. To formulate the spatial layout characters, we define the "transformation" probability of candidate i at time t and j at time $t+1$ by

$$P(X_j(t+1)|X_i(t)) = \frac{1}{\sqrt{2\pi|\Sigma_{tr}|}}\exp\left(-\frac{(X_j(t) - X_i'(t+1))\Sigma_{tr}^{-1}(X_j(t) - X_i'(t+1))^{\mathrm{T}}}{2|\Sigma_{tr}|}\right) \tag{6}$$

where $X_j(t+1)$ represents the spatial position of candidate j at $(t+1)$th lattice, $X_i'(t+1)$ is the predicted location of candidate i at $(t+1)$th lattice. $X_i'(t+1)$ is calculated on the size of candidate i, whose value on vertical orientation is equal to that of $X_i(t)$ and value on horizontal orientation is $X_i(t)+sW_i$, where W_i is the extended width (as shown in Fig. 6) of candidate i and s is the space factor which can be evaluated by averaging the space of all of the adjacent candidates. In our research, if two candidates are not overlapped and have a distance smaller than the larger width of them, they are defined as adjacent ones. If the candidate chain is broken because of the segmentation failure of some characters, we will insert some candidates randomly in the broken position.

After calculating the probability, we constructed a graph as shown in Fig. 8b in which each node represents a candidate character in the LP image. To find the best global matching between the candidates and a defined format, we obtain a real character in each lattice.

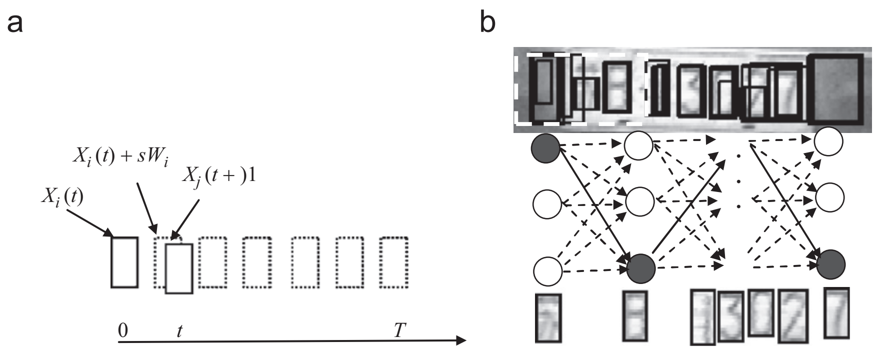


Fig. 8. Optimized format matching by dynamic programming: (a) location and predicted location of LP characters and (b) matching by dynamic programming, each node in the graph represents a candidate in LP image. A black node means a real character.

This is calculated by the following equation:

$$P(V^T) = \max_{\{i(t),j(t)\}} \left(\prod_{t=1}^{T-1} P(X_j(t+1)|X_i(t))P(X_i(t)) \right) \tag{7}$$

in all lattices. The solution for this formula can be obtained by a "back-forward" searching algorithm of dynamic programming [35].

To determine whether the matching result is a real LP or a false positive, we compare the average response of final selected characters with a threshold

$$\frac{1}{T}\sum_{t=1}^{T} G(X(t)) > T_r \tag{8}$$

If the average ANN response is larger than the threshold T_r, the image is classified into an LP and otherwise a false positive. Experiments show that T_r can range from 0.75 to 0.85. Higher value of T_r means higher precise with lower recall rate and vice versa.

5. Experimental result

In order to validate the proposed method and demonstrate its advantages, experiments on multiple style LPs are carried out with comparisons.

5.1. Test set with multi-style LPs

There are in total 14,200 LP images and 2600 non-LP images in the prepared training and testing sets. The images are all stored in RGB color format and are converted into grey images before recognition. There are two kinds of the image size: 720 (width)×576 (height) and 1920×1280 in pixels.

To evaluate the proposed method for real applications, the LP images in the test set consist of a variety of cases, including LPs of different font size, font color, rotation angles, line numbers and various formats from five nations. The LP images are captured in both daytime and night at long and short distance, etc. Fig. 9 illustrates some LP images in the test set.

We illustrate LPs of different styles from five nations in Table 1 and six styles from a same nation in Table 2. In the last column of these tables, LP example of each style is given.

5.2. Performance

Since there are LPs from five nations, we perform the configuration process five times in experiments. Each time $\Theta = \{\theta_0, \theta_1, \theta_2, \theta_3\}$ is configured by defining maximal line number, rotation angle, recognition model and possible formats. For example, Singapore LP styles are configured as $\{\theta_0 \in \{-10, \ldots, 10\}, \theta_1 \in \{1\}, \theta_2 \in \{1, 2\}, \theta_3 \in \{AAAADDDDA, AAADDDDDD, \ldots, AAAADDDAA\}\}$. After defining these parameters we can use relevant algorithms to test a ROI on these values one by one. At the same time, the occurrence probability of each value is calculated as the method in Section 3.2. For a new ROI input, it will be tested on the ordered styles (Section 3.2) by the method.

Ground truth is marked manually for performance evaluation. There are in total 5026 valid LPs in the test set. To evaluate the method, LDR (license plate detection rate), DAR (digit–alphabet recognition rate) and OVR (overall character recognition rate) are defined. DAR is an important criterion for performance evaluation. In lots of applications, if all of the alphanumerics are correctly recognized while the symbols are falsely recognized, the LP ID can still be determined with a very high probability. In the experiments the higher the three rates are, the better the performance is. The three rates can be calculated as follows:

$$\mathrm{LDR} = \frac{\text{Number of correctly located LPs}}{\text{Number of all LPs}}$$

$$\mathrm{DAR} = \frac{\text{Number of LPs in which all digits alphabets are correctly recognized}}{\text{Number of all LPs}}$$

$$\mathrm{OVR} = \frac{\text{Number of LPs in which all digits, alphabets and symbols are correctly recognized}}{\text{Number of all LPs}}$$

In the detection process, high false positive rate is not critical since there are segmentation and recognition procedures after the detection to finally determine whether a candidate is an LP or not. DAR is important for the reason that if all of the digits and alphabets are correctly recognized but the symbols are falsely recognized, the LP ID can still be identified with a very high probability.

Given the marked ground truth and the detected result, LDR, DAR and OVR can be calculated. In experiments, on average 95.9% LDR, 92.3% DAR and 90% OVR are reported, which are shown in the last row of Table 3. By comparing the performance in the table row by row, we can see that the performance is relatively steady when the method is facing LPs of various styles from different nations. In Table 4, we can see that when the style changes in the same nation, the performance also keeps steady.

In application systems, speed is another important factor to evaluate an LPR method. During the initial process, the detection speed is very low, about 1 s per image, since almost all of the parameter

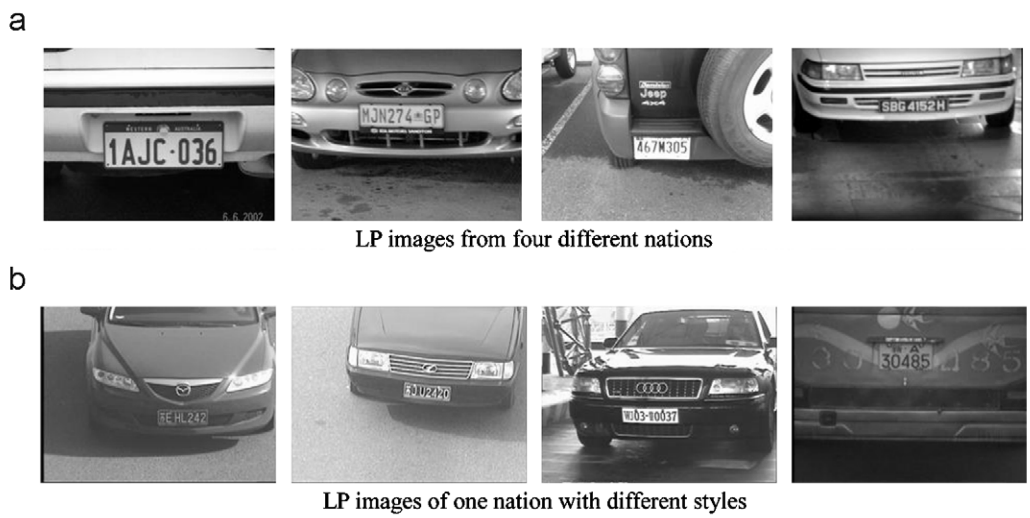


Fig. 9. Examples of LP images: (a) LP images from four different nations and (b) LP images of one nation with different styles.

Table 1
Selected styles of five nations in our test set

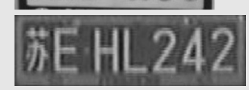
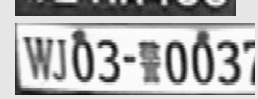
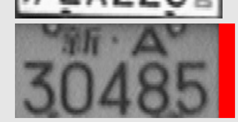
Parameters styles by nation	Line number	Max rotation angle (degree)	Recognition model number	Format number	LP image example
US	1 or 2	0	2	3	
China	1 or 2	± 15	3	6	
Singapore	1	± 10	2	5	
Australia	1	0	2	5	
South Africa	1 or 2	± 10	3	3	

Table 2
Styles of one nation

Parameters styles by nation	Line number	Max rotation angle (degree)	Recognition model number	Layout format	LP image example
Chinese car1	1	± 10	3	SADDDDD	
Chinese car2	1	± 15	3	SAADDDD	
Chinese car3	1	± 10	3	SAAADDD	
Chinese police car1	1	± 10	3	CCDD-SDDD	
Chinese police car2	1	± 10	3	SAADDDS	
Chinese cargo	2	± 10	3	SADDDDD	

values will be tested to recognize an LP. This is negligible since we need only some recognition results (about hundreds of LPs) to calculate the occurrence probability of the parameter values. We have found that in one special application there are usually 1–2 values of high probability for parameter $\theta_0, \theta_1, \theta_2$, and 5–10 values for θ_3. The product of these values will produce top 10–20 LP styles that

Table 3
Performance on multi-national LPs

Performance method	LDR (plate detection rate) (%)	DAR (digit–alphabet recognition rate) (%)	OVR (overall success rate) (%)	Speed (CPU/images per second)
US	94.3	91.4	87.6	Pentium IV 3.0 GHz/9.1
China	96.8	91.1	90.1	Pentium IV 3.0 GHz/8.9
Singapore	96.7	93.5	92.5	Pentium IV 3.0 GHz/10.6
Australia	94.2	92.3	91.3	Pentium IV 3.0 GHz/7.2
South Africa	97.5	89.9	88.9	Pentium IV 3.0 GHz/9.4
Average	95.9	92.3	90.1	Pentium IV 3.0 GHz/8.9

Table 4
Performance on multi-style LPs

Performance method	LDR (plate detection rate) (%)	DAR (digit recognition rate) (%)	OVR (overall success rate) (%)	Speed (CPU Speed/images per second)
Chinese car1	97.2	92.4	91.6	Pentium IV 3.0 GHz/9.1
Chinese car2	97.5	92.3	91.4	Pentium IV 3.0 GHz/8.2
Chinese car3	97.1	91.5	91.2	Pentium IV 3.0 GHz/10.6
Chinese police car1	96.2	91.0	89.3	Pentium IV 3.0 GHz/7.2
Chinese police car2	97.5	89.9	88.9	Pentium IV 3.0 GHz/7.9
Chinese cargo	95.5	89.6	89.2	Pentium IV 3.0 GHz/8.0
Average	96.8	91.1	90.1	Pentium IV 3.0 GHz/8.9

Table 5
Performance of different LP-image weight ratio

Performance	LDR (plate detection rate) (%)	DAR (digit–alphabet recognition rate) (%)	OVR (overall success rate) (%)
LP width/image width			
0.05–0.1	91.1	86.3	82.6
0.1–0.5	95.9	92.3	90.1
0.5–0.75	94.2	91.3	87.7

Table 6
Performance comparison

Performance method	OVR (%)	OVRD (%)	Speed (images per second)	Description of the method, P: plate detection method, S: segmentation method, R: recognition method
Method in this paper	90.1	1.6	8.0	P: edge analysis, S: optimized matching, R: feedforward ANN
Method of [1]	89.1	8.9	4.2	P: color and grey scale analysis, S: local binarization, R: ANN
Method of [13]	90	6.5	3.5	P: grey scale processing, S: profile analysis, R: ANN

frequently appear. These styles can cover 90–95% of all LPs. In experiments we have found that on average 5.75 styles will be tested to correctly recognize an LP. After the initial process the method has an average detection speed of more than eight images (720×576 by pixels) per second on a Pentium IV 3.0 GHz CPU, which means that LPs can be detected in real time even for captured video sequence. For image of higher resolution, the speed will decrease linearly.

In these images, camera–plate distance is constrained in a scope, usually 1.0–100.0 m, to get sufficient resolution of the LPs. Cameras are at height of 1.5–12.0 m to the ground. The focal length of cameras ranges 3–450 mm. When the camera–LP distance is larger than 20 m, the engineers will use a camera of long focus as extra equipment. In this condition, they should adjust the camera focus so that the ratio between the width of LP area and the width of the whole image fall into (0.1–0.5). Experiments have shown that LPs of this width ratio have the best recognition rate. Larger or smaller ratio will decrease the performance as shown in Table 5.

In order to provide an idea about the advantages of the configurable method, we compare the OVRs of several methods with/without configurable parameters. The results are presented in Table 6. We select the methods in Refs. [1,13] for comparison. Since we cannot obtain the original programs of these methods, we implement the two methods on the ideas of the papers. In the last column of Table 6, simple description of the method is listed. Our goal of the comparison is not to compare the performance of these two methods with our method for fixed style LPs, but to compare the performance drops when we extend the method from one LP style to another. Let us use OVRD to represent the drops of OVR when we extend the method from LPs of several styles to other styles, it can be seen in the table that the OVRD of unconfigurable algorithms are about 7–8% and the average OVR of configurable methods is only 1.6%. This indicates that the configurable method keeps stable performance when the style changes. So we can apply it to various styles of LPRs by just configuring their parameters. It also can be seen that the proposed method has a higher speed compared with the unconfigurable ones (Table 6).

5.3. Recognition examples

We present examples of detection results in Fig. 10. It can be seen that the proposed method performs robustly on a majority of the test images. Fig. 10c and d show skewed LPs and Fig. 10h illustrates multi-line LPs. In almost all of the images there are false positives which are usually caused by the image regions of densely distributed edges. Most of these false positives can be eliminated after format matching procedures.

In Fig. 11, we show some examples of the final recognition results. It can be seen that from Fig. 11a to h, all LPs are overall correctly recognized in spite of the variation of LP styles (character layout, multi-line, skew). Fig. 11g–i are skewed LPs that are correctly restored and the result is attached to the right of the original LP regions. In the results, there are also some failed examples as shown in Fig. 11h and i. The ground truth of Fig. 11h is "沪DB1628", which was falsely recognized as "沪DBL628". The digit "1" is falsely recognized as character "L". The ground truth of Fig. 11i is "苏BE3700" which was falsely recognized as "苏EE3760". The character "B" is falsely recognized as the character "E" and digit "0" as "6".

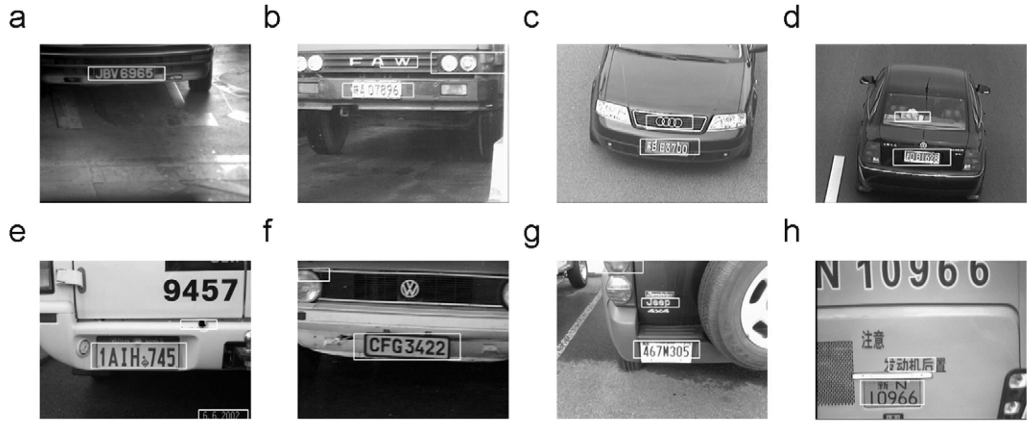


Fig. 10. Examples of initial location results.

Fig. 11. Segmentation and recognition examples (with failure examples).

By observation, we can see that the segmentation results (marked outline rectangles) of the false examples are not very accurate, which will affect the recognition algorithm and then induce false recognition result. These characters are initially located in the candidate segmentation process. Therefore, more robust candidate character segmentation algorithm should be developed in the future work.

6. Conclusion

The configurable adaptability of LPR is very important to extend the LPR techniques to new LP formats in one nation or multiple nations. In this paper, we proposed a novel multi-style LP recognition framework and provide an implementation of the detection, segmentation and recognition algorithms. On the experimental results of LPs from five nations with various styles, we conclude that the configurable method can be extended to other nations by just adjusting the parameters according to the characteristics of the LPs. By defining precisely formats of the LPs, the algorithm can perform much better than the unconfigurable ones since it can save lots of undesirable consideration of the format matching and recognition process.

The new concepts and techniques introduced in this paper include the detailed classification and analysis of multi-style LP formats, the configurable parameters for multi-style LPs, the density-based region growing algorithm for LP location, the skew refinement algorithm, the multi-line LP separation algorithm, the optimized character segmentation algorithm, and the trainable character recognition method. Detailed experimental results and comparisons with other methods are also presented, confirming that the proposed method is capable of handling multi-style LPs accurately and is robust to various LP formats and application conditions.

A known disadvantage of the current method is that it does not focus on improving the detail algorithms of the location, segmentation and recognition processes, which induce that the performance of the current implementing method can be worse than some existing methods that are developed purposely for only one nation LPs or only several styles. However, this can be solved by improving or integrating the existing algorithms of the best performance. Another limitation of the proposed configurable method is that the configurable parameters do not contain external conditions that influence LP images, for example, LP images taken in raining or snowy day, partial LP occlusion, etc. From the feedback of the users of our system, we know that the method is sensible to these factors. Therefore, improvement of the configurable parameters for external conditions and algorithm development to support these parameters will be done in future.

Acknowledgments

The authors would like to thank the anonymous reviewers for their constructive comments. This work is supported by HuaHeng-Tong InfoTech Ltd., PR China, partly supported by National Science Foundation of China with No. 60672147, 2007–2009 and the Bairen Project of Chinese Academy of Sciences.

References

[1] C.N.E. Anagnostopoulos, I.E. Anagnostopoulos, V. Loumos, E. Kayafas, A license plate-recognition algorithm for intelligent transportation system applications, IEEE Trans. Intell. Transp. Syst. 7 (2006) 377–391.
[2] H. Zhang, W. Jia, X. He, Q. Wu, Learning-based license plate detection using global and local features, in: International Conference on Pattern Recognition, 2006, pp. 1102–1105.
[3] F. Martín, M. García, L. Alba, New methods for automatic reading of VLP's (Vehicle License Plates), in: Proceedings of IASTED International Conference on SPPRA, 2002, Available from: ⟨http://www.gpi.tsc.uvigo.es/pub/papers/sppra02.pdf⟩.
[4] B. Hongliang, L. Changping, A hybrid license plate extraction method based on edge statistics and morphology, in: Proceedings of ICPR, 2004, pp. 831–834.
[5] D. Zheng, Y. Zhao, J. Wang, An efficient method of license plate location, Pattern Recognition Lett. 26 (2005) 2431–2438.
[6] S.Z. Wang, H.M. Lee, Detection and recognition of license plate characters with different appearances, in: Proceedings of Conference on Intelligent Transportation Systems, vol. 2, 2003, pp. 979–984.
[7] A. Broumandnia, M. Fathy, Application of pattern recognition for Farsi license plate recognition, in: ICGST International Conference on Graphics, Vision and Image Processing, 2005.
[8] T.D. Duan, T.L. Hong Du, T.V. Phuoc, N.V. Hoang, Building an automatic vehicle license plate recognition system, in: Proceedings of International Conference on Computer Science RIVF, 2005, pp. 59–63.
[9] V. Kamat, S. Ganesan, Efficient implementation of the Hough transform for detecting vehicle license plates using DSP'S, in: First IEEE Real-Time Technology and Applications Symposium, 1995, pp. 58–61.
[10] T.-H. Wang, F.-C. Ni, K.-T. Li, Y.-P. Chen, Robust license plate recognition based on dynamic projection warping, in: Proceedings of IEEE International Conference on Network, Sensing and Control, 2004, pp. 784–788.
[11] V. Shapiro, G. Gluhchev, Multinational license plate recognition system: segmentation and classification, in: ICPR, vol. 4, 2004, pp. 352–355.
[12] A. Mecocci, A.C. Tommaso, Generative models for license plate recognition by using a limited number of training samples, in: Proceedings of 2006 IEEE International Conference on Image Processing, pp. 2769–2772.
[13] X. Shi, W. Zhao, Y. Shen, Automatic license plate recognition system based on color image processing, in: Lecture Notes on Computer Science, vol. 3483, 2005, pp. 1159–1168.
[14] P. Comelli, P. Ferragina, M.N. Granieri, F. Stabile, Optical recognition of motor vehicle license plates, IEEE Trans. Vehicle Technol. 44 (1995) 790–799.
[15] K.I. Kim, K. Jung, J.H. Kim, Color texture-based object detection: an application to license plate localization, in: Lecture Notes on Computer Science, vol. 2388, 2002, pp. 293–309.
[16] F. Kahraman, B. Kurt, M. Gökmen, License plate character segmentation based on the Gabor transform and vector quantization, in Lecture Notes on Computer Science, vol. 2869, 2003, pp. 381–388.
[17] H. Zhang, W. Jia, X. He, Q. Wu, Learning-based license plate detection using global and local features, Int. Conf. Pattern Recognition 2 (2006) 1102–1105.
[18] C.-T. Hsieh, Y.-S. Juan, K.-M. Hung, Multiple license plate detection for complex background, in: Proceedings of International Conference on Advanced Information Networking and Applications, vol. 2, 2005, pp. 389–392.
[19] Q.X. Ye, Q.M. Huang, W. Gao, D.B. Zhao, Fast and robust text detection method in images and video frames, Image Vision Comput. 23 (2005) 565–576.
[20] S.L. Chang, L.S. Chen, Y.C. Chung, S.W. Chen, Automatic license plate recognition, IEEE Trans. Intell. Transp. Syst. 5 (2004) 42–53.
[21] J. Barroso, A. Rafel, E.L. Dagless, J. BulasCruz, Number plate reading using computer vision, IEEE Int. Symp. Ind. Electron. ISIE'97 3 (1997) 761–766.
[22] N. Xin, S. Lansun, Research on license plate recognition technology. Measurement & Control Technology, 1999, pp. 14–17 (in Chinese).
[23] H.A. Hegt, R.J. De la Haye, N.A. Khan, A high performance license plate recognition system, in: Proceedings of IEEE International Conference on System, Man and Cybernetics, vol. 5, 1998, pp. 4357–4362.
[24] D.S. Gao, J. Zhou, Car license plates detection from complex scene, in: Proceedings Fifth International Conference on Signal Processing, vol. 2, 2000, pp. 1409–1414.
[25] J.C.H. Poon, M. Ghadiali, G.M.T. Mao, L.M. Sheung, A robust vision system for vehicle license plate recognition using grey-scale morphology, in: Proceedings of IEEE International Symposium on Industrial Electronics, vol. 1, 1995, pp. 394–399.
[26] S. Nomura, K. Yamanaka, O. Katai, H. Kawakami, T. Shiose, A novel adaptive morphological approach for degraded character image segmentation, Pattern Recognition 38 (2005) 1961–1975.
[27] B.R. Lee, K. Park, H. Kang, H. Kim, C. Kim, Adaptive local binarization method for recognition of vehicle license plates, in: R. Klette, J. Uniæ (Eds.), Lecture Notes on Computer Science, vol. 3322, Springer, New York, 2004, pp. 646–655.
[28] V. Franc, V. Hlaváč, License plate character segmentation using hidden Markov chains, in: Proceedings of the 27th DAGM Symposium, vol. 1, 2005, pp. 385–392.
[29] M.R. Lyu, J. Song, M. Cai, A comprehensive method for multilingual video text detection, localization, and extraction, IEEE Trans. CSVT 15 (2005) 243–255.
[30] S. Norul, H. Sheikh, A.M. Khalid, R. Yusof, License plate recognition using multi-cluster and multilayer neural networks, in: Information and Communication Technologies, ICTTA '06, vol. 1, 2006, pp. 1818–1823.
[31] M.A. Ko, Y.M. Kim, License plate surveillance system using weighted template matching, in: Proceedings of Applied Imagery Pattern Recognition Workshop, 2003, pp. 269–274.
[32] Y.P. Huang, S.Y. Lai, W.P. Chuang, A template-based model for license plate recognition, in: Proceedings of the 2004 IEEE International Conference on Networking, Sensing & Control, vol. 2, 2004, pp. 737–742.
[33] R.C. Gonzalez, E.W. Woods, Digital Image Processing, second ed., Prentice-Hall, Engelwood Cliffs, NJ, 2001.
[34] Q.X. Ye, W. Gao, W. Zeng, Color image segmentation using density-based clustering, Int. Conf. Acoust. Speech Signal Process. 3 (2003) 345–348.
[35] R.O. Duda, P.E. Hart, D.G. Stork, Pattern Classification, second ed., Wiley, New York, 2001.

About the Author—JIANBIN JIAO received the B.S., M.S. and Ph.D. degrees in Mechanical and Electronic Engineering from Harbin Institute of Technology of China (HIT), Harbin, in 1989, 1992 and 1995, respectively. From 2002 to 2005, he had been working as a teacher of HIT. Since 2007 he has been working as a Professor of Graduate University Chinese Academy of Science, Beijing. His research interests include image processing and intelligent surveillance etc.

About the Author—QIXIANG YE received the B.S. and M.S. degrees in Mechanical and Electronic Engineering from Harbin Institute of Technology of China (HIT), Harbin, in 1999 and 2001, respectively, and the Ph.D. degree from the Institute of Computing Technology, Chinese Academy of Science 2006. Since 2006, he has been working as an Assistant Researcher at the Graduate School of the Chinese Academy of Sciences, Beijing. His research interests include image processing, pattern recognition and statistic learning, etc.

About the Author—QINGMING HUANG received the B.S., M.S. and Ph.D. degrees in Computer Engineering from Harbin Institute of Technology of China (HIT), Harbin, in 1988, 1991 and 1994, respectively. Then he became a teacher of HIT. From 1996 to 2003 he was an Associated Researcher of National University of Singapore. Since 2003 he has been working as a Professor of Graduate University Chinese Academy of Science, Beijing. His research interests include image processing and pattern recognition, video data compression, etc.