License Plate Recognition From Still Images and Video Sequences: A Survey

Christos-Nikolaos E. Anagnostopoulos, *Member, IEEE*, Ioannis E. Anagnostopoulos, *Member, IEEE*, Ioannis D. Psoroulas, Vassili Loumos, *Member, IEEE*, and Eleftherios Kayafas, *Member, IEEE*

***Abstract*—License plate recognition (LPR) algorithms in images or videos are generally composed of the following three processing steps: 1) extraction of a license plate region; 2) segmentation of the plate characters; and 3) recognition of each character. This task is quite challenging due to the diversity of plate formats and the nonuniform outdoor illumination conditions during image acquisition. Therefore, most approaches work only under restricted conditions such as fixed illumination, limited vehicle speed, designated routes, and stationary backgrounds. Numerous techniques have been developed for LPR in still images or video sequences, and the purpose of this paper is to categorize and assess them. Issues such as processing time, computational power, and recognition rate are also addressed, when available. Finally, this paper offers to researchers a link to a public image database to define a common reference point for LPR algorithmic assessment.**

***Index Terms*—Image processing, license plate identification, license plate recognition (LPR), license plate segmentation, optical character recognition (OCR).**

I. Introduction

A. License Plate Recognition (LPR)

INTELLIGENT transportation systems (ITSs) are made up of 16 types of technology-based systems divided into intelligent infrastructure systems and intelligent vehicle systems [148]. Computer vision and character recognition algorithms for LPR are used as core modules for intelligent infrastructure systems like electronic payment systems (toll payment and parking fee payment) and freeway and arterial management systems for traffic surveillance.

LPR algorithms are generally composed of the following three processing steps: 1) location of the license plate (LP) region; 2) segmentation of the plate characters; and 3) recognition of each character. The first two steps incorporate image processing techniques on still images or frame sequences (videos), whose evaluation relies on the true recognition rate and the error recognition rate.

In addition, LPR algorithms should operate fast enough to fulfill the needs of ITS. In technical terminology, a "real-time" operation for LPR stands for a fast-enough operation to not miss a single object of interest that moves through the scene. Nevertheless, with the exponential growth of the processing power, the latest developments operate within less than 50 ms [3], [152], [156] for plate detection and recognition (processing more than 20 frames/s for videos).

B. Scope of This Survey

Papers that follow the three-step framework are surveyed and classified according to their major methodology. When available, issues such as performance, execution time, and platform for each method are reported. It should be emphasized that there is a lack of uniformity in the way that methods are evaluated, and therefore, it is inappropriate to explicitly declare which methods actually demonstrate the highest performance. Indeed, one of the scopes of this paper is to highlight the lack of common test sets to achieve a common reference point for algorithmic assessment. As the first step toward this goal, a large image and video data set of Greek LPs has been collected and grouped according to several criteria such as type and color of plates, illumination conditions, various angles of vision, and indoor or outdoor images at http://www.medialab.ntua.gr/research/LPRdatabase.html.

Aiming to present a comprehensive and critical survey of up-to-date LPR methods, this paper is organized as follows: In Section II, we provide a detailed review of techniques to detect LPs in a single image or video sequence. Character segmentation methods and criteria are discussed in Section III, whereas Section IV demonstrates the character classification techniques. Finally, this paper concludes with a discussion of current trends and anticipated research in LPR.

II. LP Detection

As far as the extraction of the plate region is concerned, a categorization of methods that were reported in the literature follows, along with the description of the main processing method. It should be noted that some of these methods can be classified into more than one category and that boundaries between subcategories are not always unambiguous.

A. Binary Image Processing

Techniques based upon combinations of edge statistics and mathematical morphology [1]–[4] featured very good results.

Manuscript received April 26, 2007; revised September 6, 2007, December 18, 2007, and January 16, 2008. The Associate Editor for this paper was S. Nedevschi.

C.-N. E. Anagnostopoulos is with the Department of Cultural Technology and Communication, University of the Aegean, 81100 Mytilene, Greece (e-mail: canag@ct.aegean.gr).

I. E. Anagnostopoulos is with the Department of Information and Communication Systems Engineering, University of the Aegean, 83200 Karlovassi, Greece (e-mail: janag@aegean.gr).

I. D. Psoroulas, V. Loumos, and E. Kayafas are with the School of Electrical and Computer Engineering, National Technical University of Athens, 15780 Athens, Greece (e-mail: psoroulas@telecom.ntua.gr; loumos@cs.ntua.gr; kayafas@cs.ntua.gr).

Digital Object Identifier 10.1109/TITS.2008.922938

In these methods, the gradient magnitude and the local variance of an image are computed, based on the principle that the change of brightness in the LP region is more remarkable and more frequent than elsewhere. A disadvantage is that edge-based methods alone can hardly be applied to complex images, since they are too sensitive to unwanted edges, which may also show a high edge magnitude or variance (e.g., the radiator region in the front view of the vehicle). Despite this, when combined with morphological steps that eliminate unwanted edges in the processed images, the LP extraction rate is relatively high and fast, compared to other methods.

In [1], the conceptual model underneath the proposed algorithm is based on the morphological operation called "**top-hat**," which is able to locate small objects of significantly different brightness [55]. This algorithm, however, with a detection rate of 80%, is highly dependent on the distance between the camera and the vehicle, as the morphological operations relate to the dimensions of the binary objects. A similar application was also described in [77] with an accuracy of 93%.

In [2], a hybrid LP extraction algorithm based on **edge statistics** and **morphology** for monitoring highway ticketing systems is proposed. This approach can be divided into the following four sections: vertical edge detection, edge statistical analysis, hierarchical-based LP location, and morphology-based LP extraction. The average accuracy of locating a vehicle LP is an impressive rate of 99.6% (9786 from 9825 images). The digital images were acquired from a fixed distance and angle, and therefore, candidate regions in a specific position are given priority as already described. This *a priori* knowledge would certainly boost the results to a high level of accuracy. Performance measurements were carried out on a P IV 1700-MHz processor with 256 RAM.

Connected component analysis (CCA) is a vital technique in binary image processing that scans an already binarized image and labels its pixels into components based on pixel connectivity (either 4-connected or, usually, 8-connected). Once all groups of pixels have been determined, each pixel is labeled with a value according to the component to which was assigned. Extracting and labeling of various disjoint and connected components in an image is basic to many automated image analysis applications, as many helpful measurements and features in binary objects may be extracted. **Spatial measurements** such as area, orientation, and aspect ratio (AR) are just few of the features frequently integrated in image processing algorithms for LP detection [78], [99]. Then, using simple filtering techniques, binary objects with measurements that exceed the desired limits can be eliminated in the next algorithmic steps.

In [3], [4], [44], [45], [89], [100], [101], [125], and [131], the vertical edges of a car image using image enhancement and a **Sobel operator** were employed, followed by removing most of the background and noise edges and searching for the plate region using a rectangular window. Fig. 1 demonstrates the effect of applying a vertical edge Sobel operator in an input image. According to Zheng *et al.* [3], if the vertical edges are extracted from the car image and most of the background edges are removed, the plate area can be easily isolated from the whole edge image. The overall success rate was reported to be around 97%. The total time of processing one 384×288 image is 47.9 ms, meeting the requirements of real-time processing.

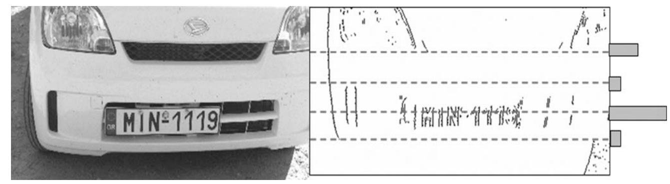

Fig. 1. Scanning in N rows and counting the existent edges.

B. Gray-Level Processing

1) Global Image Processing: Comelli *et al.* [6] presented in 1995 a system (RITA) for the recognition of car LPs. In terms of image processing for plate identification, RITA was composed of an LP area location module and a preprocessing module. LP location steps were hinged on the structure of the Italian LP, which is rectangular and contains black characters over a white background. Thus, the algorithm selects the area that presents the maximum local contrast that (possibly) corresponds to the rectangle that contains the LP.

The approach used in [14] for the plate location was to horizontally scan the image, looking for repeating contrast changes on a scale of 15 pixels or more. Draghici set the assumptions that the contrast between the characters and the background of the plate is sufficiently good and that there are at least three to four characters on a plate whose minimum vertical size is about 15 pixels. It should be noted that the particular value of 15 pixels is determined by the resolution of the camera/frame grabber used, the average distance of the vehicle from the camera, as well as the real size of the characters.

2) Partial Image Analysis: Similar methods appear in [15]–[17], [87], and [137], where the vehicle image is scanned with N-row distance, counting the existent edges (see Fig. 1). If the number of the edges is greater than a threshold value, the presence of a plate can be assumed. Specifically, in [15], if the plate is not found in the first scanning process, then the algorithm is repeated, reducing the threshold for counting edges. The method features very fast execution times as it only scans some rows of the image. Nonetheless, this method is too simple to locate LPs in several scenarios, and moreover, it is not size or distance independent.

3) Statistical Measurements: Block-based gray-level processing was also presented in the literature [5]. Similar to the methods described above, blocks with a high edge magnitude or high edge variance are identified as possible LP regions. Since block processing does not depend on the edge of the LP boundary, it can be applied to an image with an unclear LP boundary and can be implemented simply and fast. However, as not all blocks detected are LP regions, those who satisfy geometrical criteria like area and AR are favored. The accuracy rate reported in [5] in 180 pairs of images, which mainly include motorcycles, was reported to be 92.5%.

4) Hierarchical Representations: In [33], a method based on vector quantization (VQ) to process vehicle images is presented. VQ encoding can give some hints about the contents of image regions, giving additional information that can

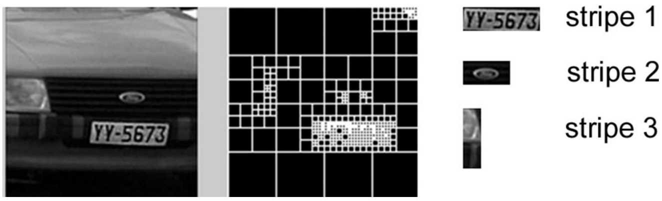

Fig. 2. (Top) Image-compression and LP location schema proposed in [33]. (Bottom) Example of quadtree decomposition and stripe extraction.

be exploited to boost location performance. Stripe extraction takes advantage of the same adaptive mechanism as adopted by VQ encoding, as image regions with higher contrast and more details are mapped by smaller blocks. LPs belong to this class of regions due to the high contrast between the text and the background. In addition, average background information could be of great aid to stripe identification, as LPs have either a bright or a dark background. Thus, the set of interesting image regions can be easily compiled by the following: 1) scanning the image structure (quadtree) for adjacent areas mapped by small-size blocks and 2) scanning the corresponding set of block mean values for high-brightness (or low-brightness) contiguous segments. Fig. 2 illustrates a plate detection example. Each stripe is scored by rewarding its consistency with the expected size. There, as a modified VQ method is initiated [34]–[36], a set of codewords is compiled and used to encode the blocks in the extracted stripes. Each stripe is characterized according to the sum of its values. Thus, the final score is associated with stripe results from the following two additive terms: 1) the partial term from stripe extraction (consistency with expected size) and 2) codeword scores that take into account the actual contents of the coded blocks. The location process eventually selects the highest score stripe from the final sorted list. According to Zunino *et al.*, LPs are identified within the two highest score stripes in 98.0% of the cases (87.6% belong to the highest score stripe).

5) Region Segmentation: An adaptive image segmentation technique sliding concentric windows (SCW) is considered for LP location in [39]. The SCW method was developed to describe the "local" irregularity in the image. The method uses image statistics such as the standard deviation and the mean value as a "heuristic" for possible plate location. In two concentric windows A and B of different sizes ($X_1 \times Y_1$ and $X_2 \times Y_2$, respectively), which scan the image from left to right and from top to bottom, the mean value [39], [40] or the standard deviation [41] is calculated (see the top part of Fig. 3). If the ratio of the statistical measurements in the two windows exceeds a threshold set by the user, then the central pixel of the concentric windows is considered to belong to an LP. The result is a binary image, which eliminates all the redundant regions from the original image (see the bottom part of Fig. 3). Anagnostopoulos *et al.* report a success rate of 96.5% for plate localization with proper parameterization of the method in conjunction with CCA measurements and the Sauvola binarization method [42].

6) Probabilistic Object Tracking in Videos: A novel method for position estimation and tracking of LPs in 3-D from a monocular camera view has been recently proposed by Yalçın and Gökmen in [106]. Given an initial estimate, the position of the plate is tracked in the successive video frames. To estimate the object and filter the measurements, the authors developed a new algorithm composed of the probability density propagation of the condensation algorithm [107] and a fine-scale optimization step according to the differential evolution (DE) algorithm [108]. According to the experiments carried out in [106], the DE–condensation algorithm outperformed the standard condensation algorithm, decreasing the computation time by about 35%, for the same level of tracking accuracy.

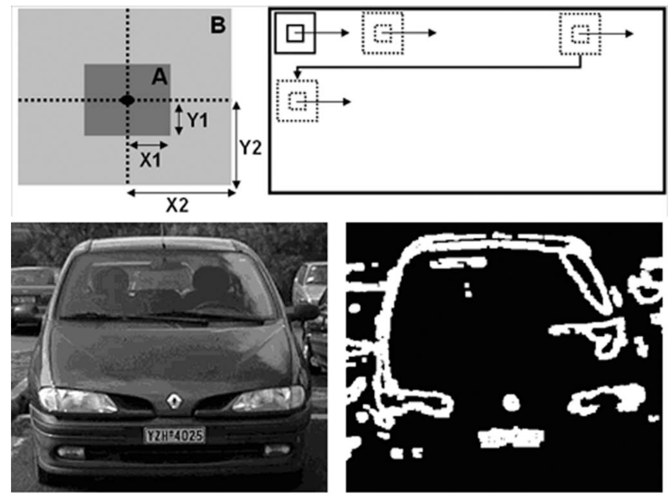

Fig. 3. (Top) SCW method. (Bottom) Resulting image after SCW execution featured in [39].

7) Image Transformations: Image transformations were also widely implemented for LP location. Gabor filters have been one of the major tools for texture analysis. This technique has the advantage of analyzing texture in an unlimited number of directions and scales. The results reported in [21] were encouraging (98% for LP detection) when applied to digital images acquired strictly in a fixed and specific angle. However, this method was tested on small sample images, as the method is computationally expensive and slow for images with large analysis.

In the method that uses Hough transform (HT), edges in the input image are detected first. Then, HT is applied to detect the LP regions. In [25], the authors acknowledge that the execution time of the HT requires too much computation when applied to a binary image with great number of pixels. As a result, the algorithm they used was a combination of the HT and a contour algorithm, which produced higher accuracy and faster speed so that it could be applied to real-time systems. However, since HT is very sensitive to boundary deformation, this approach achieved very good results (98.8% average accuracy) when applied only to close shots of the vehicle. Similar applications are described in [48], [49], [50], and [138]. Kamat and Ganesan considerably reduced the computational overhead of performing the transformation, by implementing a lookup table, limited angle, limited magnitude, and limited area transformation on a window of interest. Kong *et al.* [87] implemented a radon transform stage [105] in their LPR algorithm to draw parallel lines along the horizontal edges of the plate and detect their angle in relation to the horizontal axis. Thus, they achieved

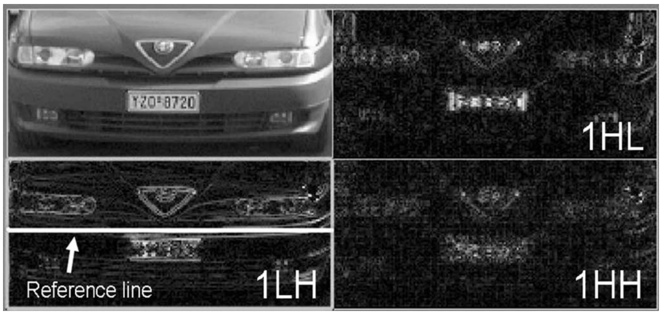

Fig. 4. Reference line for plate location was reported to be found in 1LH according to [26]. The method was executed in an image of the sample described in [39].

skew correction for plate localization with a performance rate of 96.1% in 380 images.

A wavelet transform (WT)-based method is used in [26] for the extraction of important contrast features to be used as guides in searching for LPs. In the WT, there are four subimages (subbands), namely LL, LH, HL, and HH, where L and H stand for low and high frequency, respectively. According to [26], a reference line in the first-level LH subband (1LH) subimage exactly above the plate is noticeable. Using the above reference line, a searching mask is created, speeding up the execution time. The average accuracy of detection was 92.4%. Nevertheless, the method is unreliable when the distance between the vehicle and the acquisition camera is either too far or too close or the angle of viewpoint is wide. Fig. 4 illustrates the results of first-level WT using the Haar scaling factor when executed in a close view of a vehicle image from the sample in [39].

Symmetry is also used as a feature for car LP extraction. The generalized symmetry transform (GST) produces continuous features of symmetry between two points by combining the locality constraint and reflectional symmetry. In [27], the authors propose a scan line decomposition method of calculating GST to achieve considerable reduction of the computational load. The computational speed of the proposed GST schema is approximately 30 times faster than the conventional GST. However, the effective distance is limited by the algorithm, as a closer view of the plate results to increased processing time. Moreover, this approach is insufficient in the case of slightly rotated or distorted plates. The performance was reported to be around 93% in 330 images.

C. Color Processing

Many color-based processing methods are proposed in the literature for LP location. These techniques make use of the expected plate appearance (plate background and text color) in each country. On the other hand, the solutions currently available do not provide a high degree of accuracy in natural scenery, since color is not stable when the lighting conditions change. In addition, as these methods are color based, they are country specific.

1) Color Model Transformation: The basic idea of the extraction of a plate region, according to the work proposed in [7] and [43], is that the color combination of a plate (background) and character (foreground) is unique, and this combination occurs almost only in a plate region. As the Chinese LPs have specific formats, Shi *et al.* propose that all the pixels in the input image should be classified using the HLS color model into the following 13 categories: 1) dark blue; 2) blue; 3) light blue; 4) dark yellow; 5) yellow; 6) light yellow; 7) dark black; 8) black; 9) gray black; 10) gray white; 11) white; 12) light white; and 13) other. After the classification of a region in the above colors, the AR of the expected plate is then verified.

2) Fuzzy Sets Theory: Fuzzy logic has been applied to the problem of locating LPs [18]–[20]. The authors made some intuitive rules to describe the LP and gave some membership functions for the fuzzy sets "bright," "dark," "bright and dark sequence," "texture," and "yellowness" to get the horizontal and vertical plate positions.

Zimic *et al.* [18] defines the following intuitive rules based on human perception for the object "LP": 1) bright rectangle area within which there are some dark areas; 2) located approximately in the middle or lower middle part of the image; 3) the border of the plate is bright; and 4) the approximate dimension of the plate is 530×120 mm. The concepts of "brightness" or "darkness," which are present in rules 1) and 3), are described as a fuzzy set with trapezoidal membership functions on the interval [0, 255], where 0 represents black, and 255 represents the white color in gray scale. The input image (768×576 pixels) is partitioned into subimages (elements) of size 75×25 pixels. After partitioning the image, every partitioning element computes its fitness to the four intuitive rules. The algorithm successfully located LPs in 97 out of 100 images, requiring 5 s on an SG-INDIGO 2 workstation for every image. The time was later reduced to 2 s, taking into account only every fourth line of an image. However, rules 2) and 4) restrict this algorithm to identify LPs in a specific distance.

In [19], the segmentation method incorporates the fuzzy sets of "yellowness" and "texture." The membership function for "yellowness" is determined using a histogram-based method. First, the red–green–blue (RGB) values of pixels taken from a large number of hand-cut LPs were used to construct a frequency table. For each RGB value, the aforementioned table gives the number of times each particular color occurs in the created set. The membership function was directly derived from this table by normalizing it so that the most frequently used color has a membership degree of 1. On the other hand, the membership function for "texture" in a pixel was determined on the basis of the gray-scale values of its 8-pixel neighborhood. Finally, the segmentation is performed using a fuzzy c-means clustering algorithm with two clusters (LP and non-LP). The system was tested in a huge data set of approximately 10 000 images. The system performance reached 75.4% in LP location.

The algorithm in [20] begins with an edge detector sensitive to only three kinds of edges, black–white, red–white, and green–white, as this research focuses on Korean LPs. Thus, the method creates an initial edge image E in which all other color tones beside white, black, red, and green are eliminated. Next, the RGB model of the input color image is transformed into the hue–saturation–intensity (HSI) model, and the respective H, S, and I maps of the initial image are generated. Since every map

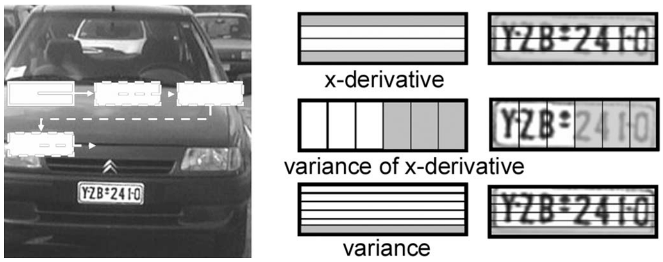

Fig. 5. (Left) Subregions being scanned in the original image. (Right) Example of features selected by AdaBoost in the 45 × 15 regions [28] over the LP region. The sum of values computed over colored regions is subtracted from noncolored regions.

encodes some characteristics about the scene, the entry of any pixel in the map expresses the degree of the pixel possessing the property. This was the basic idea of generating a fuzzy map from a given one (e.g., $\tilde{H}$ from H). Each of the four fuzzy maps, i.e., $\tilde{H}$, $\tilde{S}$, $\tilde{I}$, and $\tilde{E}$, serves as a universal set of the complete fuzzy set and is defined with specific distinctive membership functions. Finally, fuzzy maps $\tilde{H}$, $\tilde{S}$, $\tilde{I}$, and $\tilde{E}$ are integrated into a single map $\overset{\leftrightarrow}{M}$. Based on this map, regions of interest (ROIs; plates) are located in the input image, which have locally maximal values. When tested in 1088 images, the above method had the remarkable success rate of 97.9%. A similar work based on fuzzy logic was implemented in a parallel cluster unit of five personal computers to keep the processing time below 1 s [129], but the results were not reported.

3) Histogram Processing: The mean-shift estimate of the gradient of a density function and the associated iterative procedure of mode seeking is presented in [93] by Cheng. In [94], Comaniciu and Meer propose a practical method that employs mean shift in the joint spatial-range domain of color images for discontinuity-preserving filtering and image segmentation. Based on the above work, Jia *et al.* [95] applied a mean-shift procedure for color segmentation of the vehicle images to directly obtain candidate regions that may include LP regions. According to the statistical analysis performed in [95], compared to all other objects, LPs adhere to a unique feature combination of rectangularity, AR, and edge density. These three features were then calculated to candidate regions to decide whether these regions represent an LP or not. The same problem was further investigated in [96] using a modified histogram intersection (HI) method. Motivated from the work presented in [98], Jia *et al.* proposes a Gaussian weighted HI (GWHI) algorithm to facilitate the color matching using histograms. It was proved that using this algorithm, the within-class variability becomes smaller, whereas the between-class variability is maximized.

D. Classifiers

1) Statistical Classifiers: Adaptive boosting (AdaBoost) [21], [157] was used in conjunction with Haar-like features for training cascade classifiers in [28], [109], [110], and [156]. In [28], a total of 100 Haar-like features are applied to subregions sized 45 × 15 pixels being scanned as expected LP areas in the original image (e.g., Fig. 5). In these features, 37 are based on variance, 40 on the x-derivative, 18 on the y-derivative, and five on the mean pixel intensity. The classifier used was a conditional density function. Despite the fact that this study appears to be promising for the task of LP detection (success 95.6%), the authors denote that since the classifier is applied to subregions of a specific dimension, the system could not detect plates of different sizes or images acquired from different views/distances without retraining. Similar works that implement Haar-like features in conjunction with cascade classifiers are presented in [109], [110], and [156]. Decision trees were selected for the cascade classifier in [109] with a detection rate of 94.5% for the perceptrons in [110] with a success rate of 93.5% and for the statistical functions in [156].

An enhanced color–texture-based method for detecting LPs in images was presented in [8]. Kim *et al.* focus their attention on a support vector machine (SVM)-based approach that extended previous works [9]–[12] for texture classification and on the continuous adaptive mean shift (CAMShift) algorithm [13]. Specifically, the system uses a small window to scan an input image and classifies the pixel located at the center of the window into plate or nonplate (background) by analyzing its color and texture properties using an SVM. Instead of running the SVM in the whole image, it was executed only in search windows Ws within the image at a regular interval (e.g., 5 × 5 sized windows located at a regular interval of [25, 25]). Then, an LP score image is generated, in which each pixel represents the possibility of the corresponding pixel in the input image being a part of a plate region followed by the identification of LP bounding boxes by applying the CAMShift algorithm. This method was originally used to locate faces in a video stream by seeking the modes of flesh probability distribution [13]. Kim *et al.* [8] replaced the flesh probability with the LP score that is obtained by performing color texture analysis on the input image. The combination of CAMShift and SVM produced efficient LP detection by excluding a considerable number of pixels (91.3% of all the pixels in the image) from the color texture analysis, therefore achieving a substantial benefit in terms of computational time.

2) Computational Intelligence: Apart from image processing techniques, various computational intelligence architectures were proposed and implemented for plate identification, such as artificial neural networks (ANNs), genetic programming (GP), and genetic algorithms (GAs). In [29], the authors demonstrate that the most intensive computational steps in LPR could be accomplished by discrete-time cellular neural networks (DTCNNs). The tests revealed that the DTCNNs were capable of correctly identifying 85% out of all the LPs. Inspired by the work in [19], two features, "grayness" ("yellowness" in [19]) and "texture," have been appointed to each pixel in the image. The ranges for "grayness" and "texture" were determined by a histogram-based method. For the "grayness" feature, the gray values of pixels taken from a large number of exemplary LPs are used to construct a frequency table. Pixels that belong to fixed ranges were then identified using several DTCNNs, whose templates were constructed by combining the appropriate morphological operations and traditional filter techniques (dilation, Sobel, and Laplacian operators), according to the research described in [30]. In [31], the pulse-coupled neural

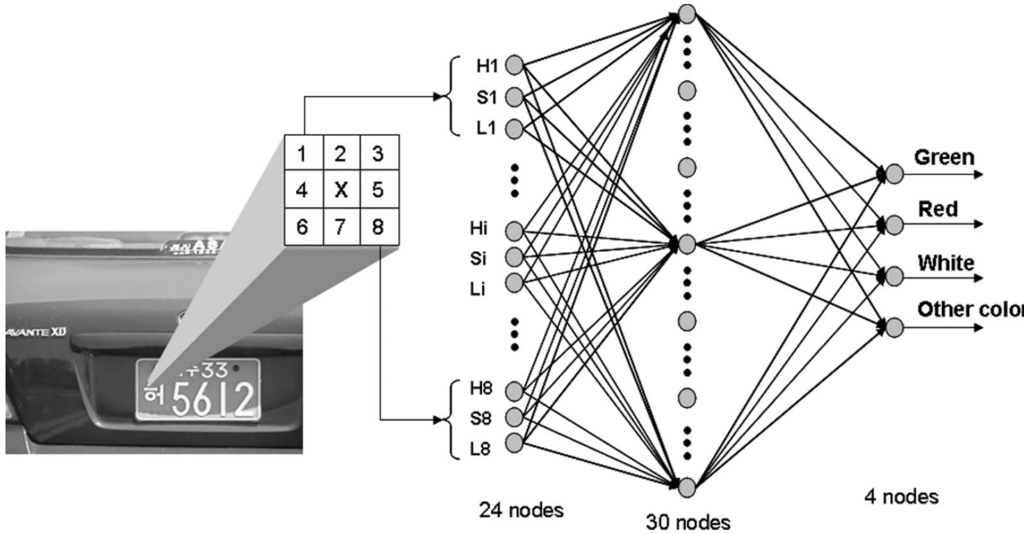

Fig. 6. Neural network proposed in [51] for LP location.

network (PCNN) schema was described to generate candidate regions that may contain an LP. A number of modifications and variations were introduced to the pioneering research in [122] and [123] to tailor its performance to image processing algorithms [124]. The results in [31] indicate that the PCNN could serve as a good preprocessing element for plate detection, but much research still remains to be performed.

Impressively good results were achieved in [32], where the time-delay neural network (TDNN) schema was implemented. TDNN is a multilayer feedforward network, whose hidden neurons and output neurons are replicated across time. TDNNs have the ability to represent relationships between events in time and to use them in trading relations for making optimal decisions [121]. Kim *et al.* presented an LP segmentation module that extracts the LP using two TDNNs as filters for analyzing color and texture properties of the LP by examining small windows of vertical and horizontal cross sections of the image. The results were remarkable in terms of speed (less than 1 s) and accuracy (97.5%). It should also be noted that there was a rough knowledge of possible plate location, which explains the fast execution time versus the complexity of the algorithm. A similar application is presented in [120].

Recently, multilayered feedforward ANNs were introduced as the first stage for LP detection, taking as input a current pixel and its neighborhood. An early research was initiated in [51], where a backpropagation neural network of 24-30-4 topology was trained to classify the color of a pixel into four distinct classes (green, red, white, and other), according to the classes of Korean LPs (see Fig. 6). The 24 input nodes correspond to the H, S, and I values of the eight neighboring pixels in a 3×3 pixel neighborhood, whereas the neurons in the hidden layer were experimentally set to 30. The main drawback was reported to be the long execution time while the performance reached 90%. Similarly, Li *et al.* [88] trained an ANN whose topology was 27-30-6 neurons with an accuracy of 92%. An identical topology is also proposed in [126] and [79]. Oz and Ercal utilized the ANN to identify LPs as text (plate) regions in a gray-level image and classified all other pixels as nontext (nonplate).

Remarkably successful results (98%) were reported in [102], where an LP detector has been proposed based on a convolutional neural network (CNN) verifier that was originally introduced by Garcia and Delakis [103]. The input image was repeatedly subsampled by a factor of 1.2, and the CNN-based verifier operated on each pyramid image to find possible text candidates that appear in the LP regions. Then, the text candidates were fused and labeled to locate the LPs using pyramid-based techniques and geometrical rules. Two geometrical constraints were designed to remove the false alarms. The width/height ratio had to be larger than 1.5, and the size of an LP had to be larger than 60×25 pixels. Lately, the architecture of a fuzzy neural network [139], [140] has been used for LP classification between the six official types in Iran, after their successful extraction using HT [138].

GP [22] and GAs [23], [24], [46], [47] were also implemented for the task of LP location. GP is usually much more computationally intensive than GA, although the two evolutionary paradigms share the same basic algorithm. The higher requirements in terms of computing resources with respect to GAs are essentially due to the much wider search space and the higher complexity of the decoding process, as well as the crossover and mutation operators. The authors indicate that the research carried out in [22]–[24], despite encouraging results, is still very preliminary and requires deeper analysis. Although the authors of [22] and [24] do not clearly report the results of their work, in [23], the identification ratio reached 80.6% on average, with a very fast execution time (0.18 s). In [24], the GA was implemented in video sequences for character extraction. Finally, recent researches incorporating GAs [46], [47] featured promising results in a limited testing set.

E. Discussion

In the LPR algorithms presented in the literature, experimentation setups are typically restricted to well-defined working conditions to obtain predictable scene features (e.g., perspective, distance, background, illumination, and vehicle position).

TABLE I
PLATE DETECTION PERFORMANCE AND MINIMUM PLATE RESOLUTION

Ref.	Size	Success	Ref.	Size	Success
[6]	100x25	84,2%	[39]	61x20	87,8%
[8]	79×38	~90%	**[43]**	**41x13**	**100%**
[14]	100x25	99%	[65]	200x40	88%
[17]	100x25	98,50%	[76]	60x25	98%
[18]	120x35	97%	**[90]**	**45x10**	**96,50%**
[19]	100x25	75,40%	[100]	65x20	96,50%
[20]	80x45	97,60%	**[110]**	**48x16**	**94,50%**
[21]	173×37	98%	**[111]**	**48x16**	**93,50%**
[24]	280x40	-	[131]	100x20	87%
[27]	200x100	93,60%	[134]	280x80	98,20%
[28]	**45x15**	**95,60%**	[139]	180x60	99%
[29]	**91x13**	**85%**	[151]	105x32	-
[32]	82x52	100%	[153]	90x30	96%
[37]	160x80	97%	**[157]**	**40x15**	**96,6%**

Bold indicates that character recognition is not possible

To overcome the problem of varying illumination, IR auxiliary units have been successfully used. This method emerged from the nature of the LP surface (retroreflective material) and has already been tested in the literature [14], [37], [79], [89], [145], [149] and in several commercial systems. Indicatively, the test set in [89] includes 2483 images of Iranian vehicles captured using IR illumination units, achieving an impressive detection rate of 99.3%.

In addition, some authors claim that their plate detection systems are able to cope with highly variable camera-to-car distances [17], [150]. However, the majority of the experiments verified that there is always a tradeoff between detection accuracy and minimum plate resolution. Table I holds the performance rates for LP detection over the minimum necessary plate resolution reported. There are several studies that report successful plate localization when the vertical resolution (height) of the plate ranges from 10 to 16 pixels [28], [29], [43], [89], [109], [110], [156]. However, such poor plate resolution does not allow reliable extraction of the characters, and therefore, these studies address only the task of LP detection.

In most systems with a subsequent recognition module, the vertical resolution of the plate fluctuates from 20 to 40 pixels. Later on, to proceed to character recognition, the segmented plates are spatially transformed and/or are subjected to enhancement techniques. The algorithms in [6], [19], [39], and [130] reveal that plates with a height of at least 20–25 pixels can be successfully processed for character segmentation and recognition, provided, of course, that the plates are in sufficiently good condition.

III. CHARACTER SEGMENTATION

The LP candidates considered in the plate location stage are examined in the phase of character segmentation. A wide variety of techniques to segment each character after plate localization has been developed, as described in this section, which groups them according to their methodology. Again, it should be emphasized that many papers incorporate more than one method (e.g., adaptive thresholding followed by the projection method), and therefore, the classification that follows is not unique.

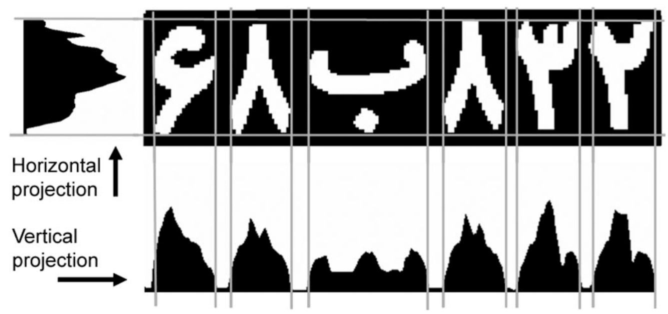

Fig. 7. Character extraction using the horizontal and vertical projection method.

A. Binary Image Processing

1) Projections and Binary Algorithms: Reviewing the literature, it was evident that the method that exploits vertical and horizontal projections of the pixels [7], [16], [25], [32], [49], [51], [52], [77]–[79], [87], [125], [130], [134], [137] is the most common and simplest one. Obtaining a binary image, the idea is to add up image columns or rows and obtain a vector (or projection), whose minimum values allow us to segment characters (see Fig. 7). CCA is also intensely involved in character segmentation, in conjunction with binary object measurements such as height, width, area [1], [19], [88], [89], [90], and orientation [4], [89]. In other cases, CCA is supported by either VQ [21] or mathematical morphology [29], [52], [128]. Usually, the CCA method labels the pixels into components based on 8-neighborhood connectivity, but in [65], the binarized image is decomposed into 4-neighbor connected components.

2) Mathematical Morphology: The work in [52] is completely dedicated to the task of character segmentation, describing a morphology-based adaptive approach for seriously degraded plate images. Following the adaptive binarization method in [53] for color images, noise is identified by applying the thickening [54] and pruning algorithm [55] to the binary image. Nomura *et al.* performs an adaptive segmentation of characters, searching for natural segmentation points in the projection histogram and merging fragments that belong to the same character. For the aforementioned task, prior knowledge of the maximum quantity of segments for each set (letters or digits) was employed to decide whether the merging is necessary. The mathematical operators for the merging process, as well as the respective ones for separating overlapping or connected characters, mainly exploit the information provided by the vertical projection of the set and are given in detail in [52]. From a test set of 1189 degraded plate images, the entire character content was correctly segmented in 1005 of them.

3) Contours: Contour tracking and modeling is also incorporated for character segmentation. In [64], an algorithm inspired by the contour tracking method known as "bug following" [117] is implemented for character segmentation. Capar and Gokmen [111] established a shape-driven active contour model, which utilizes a variational fast marching algorithm, and applied it to the plate character-segmentation problem. First, coarse location of each character is found by an ordinary fast marching technique [112] combined with a gradient- and

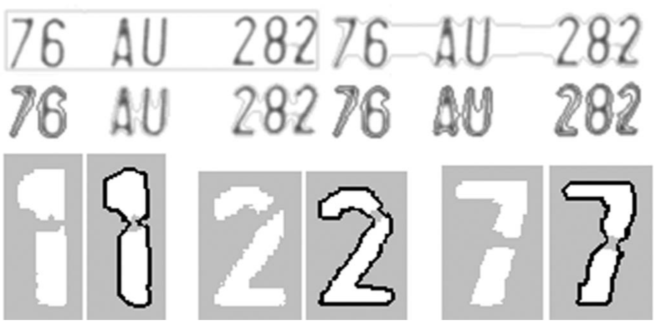

Fig. 8. (Top) Segmentation sequence. (Bottom) Merging broken characters.

curvature-dependent speed function, as presented in [113]. The method proceeds with the segmentation of exact boundaries through the calculation of a special fast marching methodology, which depends on gradient, curvature, and shape similarity information. Shape similarity statistics were again embedded into a fast marching method to stop the evolving front when the front resembles one of the trained shapes (e.g., top of Fig. 8). In addition, Fig. 8 (bottom) demonstrates an example of initially broken characters and the final merged segmentation results using this method.

B. Gray-Level Processing

1) Histogram Processing: Draghici [14] concluded that no unique thresholding technique can produce acceptable results in most situations. Therefore, the system proposed in [14] handled the character segmentation problem on a higher level by combining the results of various histogram-thresholding techniques, like using intensity–gradient scattergrams or finding a valley in the intensity distribution [63], also using feedback from later stages of the system itself. Recently, a similar histogram-based method was implemented in [15], where after the histogram generation of the plate image, the threshold is determined by a formula that employs the entropy values of background and foreground or by cumulative distribution functions.

2) Local/Adaptive Thresholding and Transformations: Since binarization with one global threshold cannot always produce useful results in such cases, adaptive local binarization methods were also implemented. In many local binarization methods, an image is divided into $m \times n$ blocks, and a threshold is chosen for each block. The research carried out in [37] follows the above block division technique that implements the dynamic binarization method of Otsu [38]. Using the configurations of the sensor and the distance between the sensor and vehicles, the expected size of characters in the images was estimated in advance. To efficiently deal with all possible combinations of candidate regions of characters, the authors employed several hypotheses that were represented using tree interpretation, as in [58]. Similarly, in [65] and [127], local thresholding techniques are used for each pixel. In [65], the threshold is computed by subtracting a constant c from the mean gray level in an $m \times n$ window centered in the pixel, whereas in [127], the threshold is given by the Niblack binarization formula $T(i,j) = m(i,j) + k \cdot \sigma(i,j)$, where T is the threshold at pixel (i,j), $k = -0.2$ is a weight factor, and $\sigma(i,j)$, $m(i,j)$ are the standard deviation and the mean of the 15×15 local neighborhood of pixel (i,j), respectively.

Fig. 9. Binarization results from left to right: noisy LP image, global threshold, and adaptive threshold in [59].

Fig. 10. Successful segmentation of characters following the SCW method. (Left column) Original segmented plate, (middle column) SCW result, and (right column) character segmentation after negation and height and orientation measurements.

In [59], the authors propose an improved local binarization method that determines a threshold for each character region. Therefore, the positions and sizes of rectangles that possibly contain characters in an LP were determined first using projections. Based on the assumption that brightness changes mainly occur in a vertical direction, Lee *et al.* recommend a horizontal pixel accumulation histogram for every character region to find out if a character is split or missing. In this case, the region is divided into two, and thresholds are redetermined for the new regions. The authors report an improvement of 5% compared to local binarization methods, and an example is shown in Fig. 9.

The character segmentation phase in [6] removes the noise and enhances the image, and moreover, it detects, centers, and normalizes the LP image. The output of this module is still an image that represents the whole LP and not the discrete characters. Comelli *et al.* describe that iterative methods founded on homomorphic filtering [60], [61] provided satisfactory results for a good amount of their testing images.

An adaptive binarization method based on a previous study [62] is also adopted in [20] to avoid the problem that nonuniform illumination creates. Chang *et al.* combine the research in [62] with HT and CCA. Specifically, they proceed to the calculation of the AR in each binary object, deleting those with an AR outside the prescribed range. Then, an alignment of the remaining objects is derived by applying the HT to their centers of gravity. The objects that disagree with the alignment are removed. In addition, an adaptive binarization method is described in [39] and [40], implementing the SCW technique using the standard deviation measurement in conjunction with CCA measurements (see Fig. 10).

C. Classifiers

The method proposed in [24] is different from many existing single-frame approaches, because it simultaneously utilizes spatial and temporal information. First, it models the extraction of characters as a Markov random field (MRF), where the randomness is used to describe the uncertainty in pixel label assignment. MRF models can be used to incorporate prior contextual information or constraints in a quantitative way. Local spatial/contextual dependences can be utilized to perform binarization [56]. Therefore, Cui and Huang prove that by using the MRF modeling, the extraction of characters can be modeled as the problem of maximizing *a posteriori* probability based on given prior knowledge and observations. After that, a GA with a local greedy mutation operator is employed to optimize the objective function and speed up the convergence based on [57]. The method was developed for LP character segmentation in video sequences.

Hidden Markov chains are used to model a stochastic relation between an input image and the corresponding character segmentation. Franc and Hlavac [91] proposed a character segmentation method for noisy low-resolution LP images, where the segmentation problem was expressed as the maximum *a posteriori* estimation from a set of admissible segmentations. The method was based on [92], exploiting *a priori* knowledge such as the predetermined number of characters in the plate and their equal (but unknown) segmented width. The statistical model was created from a training set of ground-truth segmentation examples of Czech plates provided by a user. The algorithm was executed on the basis of dynamic programming and tested on 1000 examples captured from a real-life LPR system with an error rate of 3.3%.

D. Discussion

Character segmentation is needed to perform character recognition, which fully relies on isolated characters. Incorrectly segmented characters are not likely to be successfully recognized. In fact, most of the recognition errors in the LPR systems are not due to missing recognition power but to segmentation errors. As already discussed at the end of Section II, LP height should be at least 20–25 pixels if character segmentation and processing is required. However, sometimes, this resolution is not enough as dirt, physical damage, and unpredictable shadows degrade the segmentation performance.

IV. Character Recognition

For the recognition of segmented characters, numerous algorithms use statistical classifiers, computational intelligence architectures, and common pattern-matching techniques.

A. Classifiers

1) Statistical/Hybrid Classifiers: When hidden Markov models (HMMs) are employed, the recognition begins with preprocessing and parameterization of the ROIs detected in the previous phase (character segmentation). Based on [66], the recognition result in [65] was reported to be 95.7% after a complex procedure of preprocessing and parameterization for the HMMs: one for every character. The authors also reported that the width of the plate in the image after rescaling lies between 25% and 75% of the original image width (between 200 and 600 pixels). This reveals the necessity for good character analysis when implementing HMMs, which poses a restriction on the effective distance of the plate recognition system. This prerequisite is also featured in [25], where the recognition results reached 92.5%.

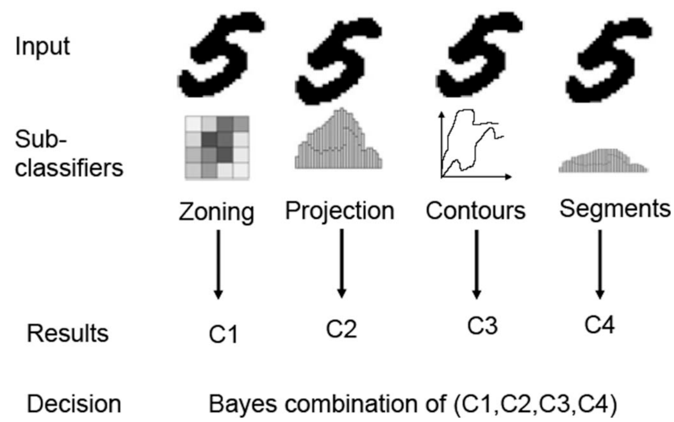

Fig. 11. Statistical classification stage in [86] combining four sub-classifiers using the Bayes method [83].

Furthermore, the authors in [32] designed a system that implements SVMs and reports an impressive average character recognition rate of 97.2% for Korean plates. Four SVM-based character recognizers were applied to recognize upper characters, upper numerals, lower characters, and lower numerals on the plate.

Many researchers integrate two kinds of classification schemes [80], [81], multistage classification schemes [82], or a "parallel" combination of multiple classifiers [83]–[85]. Pan *et al.* [86] proposed a two-stage hybrid recognition system that combines statistical and structural recognition methods to achieve robust and high recognition performance. Initially, skew images of car plates were corrected and normalized. In the first recognition stage, four statistical subclassifiers (SC1, SC2, SC3, and SC4) independently recognize the input character, and the recognition results are combined using the Bayes method [83]. Subclassifier SC1 uses the zoning density [119], SC2 uses the vertical projections, SC3 calculates the contour profile, and SC4 counts line segments in each row and column (see Fig. 11). Finally, if the output of the first (statistical) stage contains characters that belong to prescribed sets of similar characters, the second (structural) stage is initiated as a complement to the first. Structure features are obtained and are then fed into a decision tree classifier. The success ratio reached 95.41% in a huge testing data set of more than 10 000 plates.

Alternatively, coarse-to-fine classification is an efficient way to organize object recognition to accommodate a large number of possible hypotheses and to systematically exploit shared attributes and the hierarchical nature of the visual world. The basic structure is a nested representation of the space of hypotheses and a corresponding hierarchy of (binary) classifiers [114]. A scene is processed by visiting nonoverlapping 5×5 blocks, processing the surrounding image data to extract "spread" edge features based on the research conducted in [115], and classifying this subimage according to the coarse-to-fine search strategy described in [116]. There are 37 classes defined by the prototypes (bit maps), shown at the top of

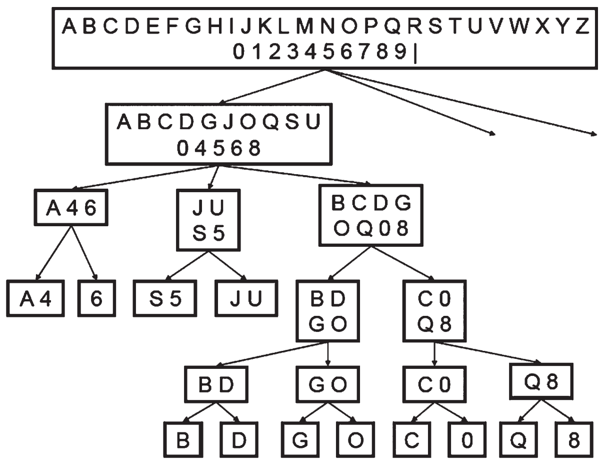

Fig. 12. Root of the tree (up) is the 37 prototypes for characters in Massachusetts LPs. The leaves are pure classes [114]. The figure is not complete as only one branch of the class hierarchy is shown.

TABLE II
MULTILAYERED FEEDFORWARD NEURAL NETWORKS FOR CHARACTER RECOGNITION DETAILS

Ref.	Topology/(performance)		No. of output classes
[15]	16-20-9	(95%)	9 numbers
[19,29]	24-15-36	(98.5%)	26 letters+10 numerals
[80]	209-104-36	(95%)	26 letters +10 numerals
[126]	15-10-8	(96%)	8 (binary ASCII code of 36 characters)
[127]	108-50-35	(95%)	25 letters +10 numerals
[128]	50-60-100-36	(92.5%)	26 letters +10 numerals
[133,136]	Not reported	(81%)	Not reported
[134]	420-26-38	(98.2%)	36 alphanumeric classes + 1 class for border + 1 class for stamps

Fig. 12, which correspond to the 36 alphanumerical characters plus the special character "|." Special emphasis was given to "pairwise" competition between any two similar interpretations of a character (e.g., S/5 and J/U). The algorithm was evaluated on 520 plates. The correct character string was found on all but 17 plates. However, the classification rate per symbol was much higher: more than 99%.

2) ANNs: Multilayered feedforward neural networks are also used for LP character identification in many works [15], [19], [29], [79], [125]–[127], [132], [135], usually following a common methodology. The classical training method for feedforward neural networks is error backpropagation [118]. The network has to be trained for many training cycles to attain good performance. Moreover, the number of hidden layers, as well as the number of respective neurons, has to be defined after a trial-and-error procedure [67].

Table II indicates the various multilayered neural network topologies for plate character recognition along with specific details as reported in the literature. In [19] and [29], a multilayered perceptron (MLP) architecture that contains 24 input, 15 hidden, and 36 output neurons was trained to recognize 36 characters of the Latin alphabet. Contrary to [15], where the classification was done on the basis of binary image input, the input neurons in [19] were fed with 24 features previously generated from a DTCNN. Moreover, a character was confidently recognized only if the corresponding output neuron exceeded 0.85 and all other output neurons had an output level below 0.25, where 1 denotes a 100% confident level for the output. The network was applied to segmented LPs for character recognition, achieving excellent results (98.5%) in a large set of 10 000 images.

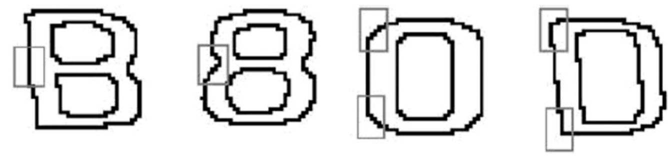

Fig. 13. Distinguishing parts of ambiguous characters [20].

Similar approaches are reported in [79] and [125]. In [133], special attention was given to the training of a three-layered MLP for the correct identification of problematic pairs (e.g., I/l, B/8, and O/0) or for the classification of border parts and legislation stamps that appear in LPs. During the training procedure, the problematic samples were trained more often, and special characters for borders and legislation stamps were added to the training sets. After the special training, the rate of correct classification was reported to be 98.2%.

In addition, a nonstandard feedforward neural network based on the adaptive resonance theory (ART) introduced by Grossberg [136] is studied in [64], as an improvement of classical multilayered feedforward networks. The basic difference between backpropagation and ART is that the latter is an unsupervised learning technique, which dynamically creates the number of nodes in the hidden layer and guarantees the convergence to learning. The results reported in [64] reveal a slight improvement in the recognition performance of the ART model (95%) in a relatively small data set.

In [20], self-organized neural networks based on Kohonen's self-organized feature maps were implemented to handle noisy, deformed, broken, or incomplete characters acquired from LPs that were bent and/or tilted with respect to the camera. The method focuses on accuracy at the cost of increased complexity and execution speed. The success rate for character identification, in a large set of 1061 LPs in various viewpoints (combinations of angle and distance), was a remarkable 95.6%. To overcome misclassification of the similar character pairs (8,B), (0/D), and (O/D), the authors in [20] predefined an ambiguity set that contains the pairs of the misclassified characters. During character recognition, once an unknown character is classified as one of the characters in the ambiguity set, an additional minor comparison between the unknown character and the classified character is performed. The comparison then focuses only on the nonambiguous parts (see Fig. 13).

Probabilistic neural networks (PNNs) were introduced in the neural network literature by Specht [70]. These types of neural networks can be designed and trained faster, as the hidden-layer neurons are defined by the number of the training patterns and are only trained once [71]. PNNs for LPR were first introduced in an early version of an LPR system [69], where two PNNs, i.e., one for alphabet recognition and the other for number recognition, were trained and tested. The recognition rates reported in the literature are very encouraging when PNNs were trained and tested in noisy, tilted, and degraded patterns (over 90%) [39], [72]. Additionally, in [73], the authors reported an impressive recognition rate that reaches 99.5%. Kohonen's algorithm and the learning vector quantizer for training neural networks are mentioned in [68] and [59], respectively, without further details.

B. Pattern/Template Matching

The pattern matching technique is a suitable technique for the recognition of single-font, not-rotated, and fixed-size characters. Although this method is preferably used in binary images, properly built templates also obtained very good results for gray-level images. A template matching application is described in [6], where the recognition process was based on the computation of the normalized cross-correlation values for all the shifts of each character template over the subimage that contains the LP. It was reported that more than 90% of central processing unit time was consumed for the computation of the cross-correlation measures between the various templates and the relative subimage. However, as the subimage is a small image, the problem of computational time can be overcome. Template matching is successfully implemented in [76], where the whole recognition process is based on the computation of the root-mean-square error, for all the shifts of the template g over a subimage f with size $M \times N$. Template matching for character recognition is also incorporated in [77], [78], and [137].

Finally, the Hausdorff distance is a method of comparing two binary images (or two sets of "active" pixels). The method possesses all the mathematical properties of a metric, and its recognition rate is very similar to that obtained with neural network classifiers but slightly slower. On the basis of the research conducted in [74] and [75], Martin *et al.* concluded that the Hausdorff distance may constitute a complementary recognition method if real-time requirements are not very strict.

C. Discussion

Since incorrectly segmented characters from the character segmentation step may be forwarded to the recognizer, the last module in the LPR process should be able to successfully handle any ambiguity that may arise. Very good results have been reported using neural networks and statistical classifiers, but a huge amount of learning samples is needed to train such algorithms. Given also that optical character recognition (OCR) technologies are already mature and that they are continuously enhanced over time, future LPR developers shift their attention to OCR improvement in sets of ambiguous characters (1/I, 0/O, 0/D, 2/Z, 8/B, and 5/S) rather than redesigning or retraining a character recognition system from scratch.

V. Discussion and Conclusion

A. Current Trends

LPR, as a means of vehicle identification, may be further exploited in various ways such as vehicle model identification, under-vehicle surveillance, speed estimation, and intelligent traffic management. For the vehicle model identification task, the position of the LP could play an important role in segmenting a distinctive reference area of the frontal view of the vehicle [141]–[143]. Moreover, for under-vehicle inspection, it is assumed that a template under-vehicle image for each inspected vehicle has been archived into a database in advance. Based on the incoming vehicle LP, the respective template image is retrieved from the database and then compared to the one acquired during real-time under-vehicle inspection [40]. With a binocular arrangement of the image capturing system, the spatial and temporal range of plate image between consecutive views could provide an improved accuracy in speed computation of the vehicle [145].

LPR may also support intelligent vehicle management [146], [147] or vehicle behavior monitoring and warning [144]. Vehicle reidentification in an integrated highway management system is also proposed [149] with numerous practical traffic applications. The derivation of section travel times (time taken by a vehicle to go from one point to another) could be useful to transportation engineers for traffic operations, planning, and control.

B. Outlook for the Future

Although significant progress has been made in the last decade, there is still work to be done, as a robust LP detection system should effectively work for a variety of environmental illumination, plate types/conditions, as well as acquisition parameters.

Increased mobility and internationalization set the challenge of developing an effective LPR system that could handle plates from various countries with different character sets and syntax. So far, this issue has not been significantly addressed in the literature, as the recognition systems were country specific. As already described in this survey, many algorithms utilize fixed plate geometry, color, and character fonts for LP location, segmentation, and character recognition. Little early research [39], [153], and [154] addressed this issue, still with several restrictions.

Moreover, most LPR systems focus on the processing of images with one vehicle. Nevertheless, input images may contain more than one vehicle or motorcycles. As far as the first case is concerned, the algorithms proposed in [5], [8], [17], [26], [32] [39], [87], [89], [109], and [110] were tested in samples containing more than one vehicle in the image, but only in the research conducted in [5] were motorcycle plates successfully handled.

In addition, assuming that LP regions are detectable even in very low resolution, an open topic for future research is the readability improvement of LP text using image processing techniques. Research for improving degraded plates has lately been directed to superresolution methods for video sequences [151] or to blurred plate images [155] with promising results.

C. Summary

This paper has attempted to provide a comprehensive survey of research on LPR and also to offer some structural categories for the methods described in more than 100 papers. The major contribution of this paper has been to provide a brief source of reference for researchers involved in LP detection and recognition. It should be noted that there are several commercial LPR systems, whose evaluation is beyond the scope of this paper. This is due to the fact that their operation is strictly confidential, and their performance rates are often overestimated for promotional reasons.

In addition, it is evident that the number and quality of testing examples have a direct effect on the overall LPR performance. However, this factor is often ignored in performance evaluation or comparison, which is the appropriate criterion for an algorithmic assessment. Therefore, providing the Medialab LPR database (as discussed in the introduction), we anticipate researchers engaged in LPR or in related projects to report their results on this publicly available set or, alternatively, to contribute to the enrichment of this test database with plates of various national types and formats. This would support the systematic performance evaluation in the scientific community worldwide and allow system developers to know which methods are competitive in which application.

References

[1] F. Martin, M. Garcia, and J. L. Alba, "New methods for automatic reading of VLP's (Vehicle License Plates)," in *Proc. IASTED Int. Conf. SPPRA*, 2002.

[2] B. Hongliang and L. Changping, "A hybrid license plate extraction method based on edge statistics and morphology," in *Proc. ICPR*, 2004, pp. 831–834.

[3] D. Zheng, Y. Zhao, and J. Wang, "An efficient method of license plate location," *Pattern Recognit. Lett.*, vol. 26, no. 15, pp. 2431–2438, Nov. 2005.

[4] S. Wang and H. Lee, "Detection and recognition of license plate characters with different appearences," in *Proc. Conf. Intell. Transp. Syst.*, 2003, vol. 2, pp. 979–984.

[5] H.-J. Lee, S.-Y. Chen, and S.-Z. Wang, "Extraction and recognition of license plates of motorcycles and vehicles on highways," in *Proc. ICPR*, 2004, pp. 356–359.

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

[7] X. Shi, W. Zhao, and Y. Shen, *Automatic License Plate Recognition System Based on Color Image Processing*, vol. 3483, O. Gervasi *et al.*, Ed. New York: Springer-Verlag, 2005, pp. 1159–1168.

[8] K. I. Kim, K. Jung, and J. H. Kim, *Color Texture-Based Object Detection: An Application to License Plate Localization*, vol. 2388, S.-W. Lee and A. Verri, Eds. New York: Springer-Verlag, pp. 293–309.

[9] Y. Zhong and A. K. Jain, "Object localization using color, texture and shape," *Pattern Recognit.*, vol. 33, no. 4, pp. 671–684, Apr. 2000.

[10] M. Mirmehdi and M. Petrou, "Segmentation of color textures," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 22, no. 2, pp. 142–159, Feb. 2000.

[11] K. I. Kim, K. Jung, S. H. Park, and H. J. Kim, "Support vector machines for texture classification," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 24, no. 11, pp. 1542–1550, Nov. 2002.

[12] E. Osuna, R. Freund, and F. Girosi, "Training support vector machines: An application to face detection," in *Proc. IEEE Conf. CVPR*, San Juan, Puerto Rico, 1997, pp. 130–136.

[13] G. R. Bradski, "Real time face and object tracking as a component of a perceptual user interface," in *Proc. IEEE Workshop Appl. Comput. Vis.*, Sarasota, FL, 1998, pp. 214–219.

[14] S. Draghici, "A neural network based artificial vision system for license plate recognition," *Int. J. Neural Syst.*, vol. 8, no. 1, pp. 113–126, Feb. 1997.

[15] A. Broumandnia and M. Fathy, "Application of pattern recognition for Farsi license plate recognition," in *Proc. Int. Conf. GVIP*, Cairo, Egypt, 2005.

[16] T.-H. Wang, F.-C. Ni, K.-T. Li, and Y.-P. Chen, "Robust license plate recognition based on dynamic projection warping," in *Proc. IEEE Int. Conf. Netw., Sens. Control*, 2004, pp. 784–788.

[17] J. Cano and J. C. Perez-Cortes,*Vehicle License Plate Segmentation in Natural Images*, vol. 2652, F. J. Perales *et al.*, Eds. New York: Springer-Verlag, 2003, pp. 142–149.

[18] N. Zimic, J. Ficzko, M. Mraz, and J. Virant, "The fuzzy logic approach to the car number plate locating problem," in *Proc. IIS*, Grand Bahama Island, The Bahamas, 1997, pp. 227–230.

[19] J. A. G. Nijhuis, M. H. ter Brugge, K. A. Helmholt, J. P. W. Pluim, L. Spaanenburg, R. S. Venema, and M. A. Westenberg, "Car license plate recognition with neural networks and fuzzy logic," in *Proc. IEEE Int. Conf. Neural Netw.*, 1995, vol. 5, pp. 2232–2236.

[20] 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, Mar. 2004.

[21] F. Kahraman, B. Kurt, and M. Gökmen, *License Plate Character Segmentation Based on the Gabor Transform and Vector Quantization*, vol. 2869, A. Yazici and C. Sener, Eds. New York: Springer-Verlag, 2003, pp. 381–388.

[22] G. Adorni, S. Cagnoni, and M. Mordonini, "Efficient low-level vision program design using sub-machine-code genetic programming," in *Proc. Workshop sulla Percezione e Visione nelle Macchine*, 2002.

[23] S. Yoshimori, Y. Mitsukura, M. Fukumi, and N. Akamatsu, *License Plate Detection Using Hereditary Threshold Determine Method*, vol. 2773, V. Palade, R. J. Howlett, and L. C. Jain, Eds. New York: Springer-Verlag, 2003, pp. 585–593.

[24] Y. Cui and Q. Huang, "Extracting characters of license plates from video sequences," *Mach. Vis. Appl.*, vol. 10, no. 5/6, pp. 308–320, Apr. 1998.

[25] T. D. Duan, T. L. H. Du, T. V. Phuoc, and N. V. Hoang, "Building an automatic vehicle license-plate recognition system," in *Proc. Int. Conf. Comput. Sci. (RIVF)*, 2005, pp. 59–63.

[26] C.-T. Hsieh, Y.-S. Juan, and K.-M. Hung, "Multiple license plate detection for complex background," in *Proc. Int Conf. AINA*, 2005, vol. 2, pp. 389–392.

[27] D.-S. Kim and S.-I. Chien, "Automatic car license plate extraction using modified generalized symmetry transform and image warping," in *Proc. ISIE*, 2001, pp. 2022–2027.

[28] L. Dlagnekov, *License Plate Detection Using AdaBoost*, Mar. 2004, San Diego, CA: Comput. Sci. Eng. Dept., Univ. California, San Diego. [Online]. Available: http://www.cse.ucsd.edu/classes/fa04/cse252c/projects/louka.pdf

[29] M. H. ter Brugge, J. H. Stevens, J. A. G. Nijhuis, and L. Spaanenburg, "License plate recognition using DTCNNs," in *Proc. IEEE Int. Workshop Cellular NNs Appl.*, 1998, pp. 212–217.

[30] M. H. ter Brugge, J. A. G. Nijhuis, and L. Spaanenburg, "Transformational DT-CNN design from morphological specifications," *IEEE Trans. Circuits Syst. I, Fundam. Theory Appl.*, vol. 45, no. 9, pp. 879–888, Sep. 1998.

[31] M. I. Chacon and A. Zimmerman, "License plate location based on a dynamic PCNN scheme," in *Proc. Int. Joint Conf. Neural Netw.*, 2003, vol. 2, pp. 1195–1200.

[32] K. K. Kim, K. I. Kim, J. B. Kim, and H. J. Kim, "Learning-based approach, for license plate recognition," in *Proc. IEEE Signal Process. Soc. Workshop, NNs Signal Process.*, 2000, vol. 2, pp. 614–623.

[33] R. Zunino and S. Rovetta, "Vector quantization for license-plate location and image coding," *IEEE Trans. Ind. Electron.*, vol. 47, no. 1, pp. 159–167, Feb. 2000.

[34] R. M. Gray, "Vector quantization," *IEEE ASSP Mag.*, vol. 1, no. 2, pp. 4–29, Apr. 1984.

[35] N. Nasrabadi and R. King, "Image coding using vector quantization: A review," *IEEE Trans. Commun.*, vol. 36, no. 8, pp. 957–971, Aug. 1988.

[36] S. Ridella, S. Rovetta, and R. Zunino, "Plastic algorithm for adaptive vector quantization," *Neural Comput. Appl.*, vol. 7, no. 1, pp. 37–51, Mar. 1998.

[37] T. Naito, T. Tsukada, K. Yamada, K. Kozuka, and S. Yamamoto, "Robust license-plate recognition method for passing vehicles under outside environment," *IEEE Trans. Veh. Technol.*, vol. 49, no. 6, pp. 2309–2319, Nov. 2000.

[38] N. Otsu, "A threshold selection method from gray-level histograms," *IEEE Trans. Syst., Man, Cybern.*, vol. SMC-9, no. 1, pp. 62–66, Jan. 1979.

[39] C. Anagnostopoulos, I. Anagnostopoulos, E. Kayafas, and V. Loumos, "A license plate recognition system for intelligent transportation system applications," *IEEE Trans. Intell. Transp. Syst.*, vol. 7, no. 3, pp. 377–392, Sep. 2006.

[40] C. Anagnostopoulos, T. Alexandropoulos, S. Boutas, V. Loumos, and E. Kayafas, "A template-guided approach to vehicle surveillance and access control," in *Proc. IEEE Conf. Advanced Video Signal Based Surveillance*, 2005, pp. 534–539.

[41] C. Anagnostopoulos, I. Anagnostopoulos, D. Vergados, E. Kayafas, and V. Loumos, "Sliding windows: A software method for real-time inspection in textile surface," *Textile Res. J.*, vol. 74, no. 7, pp. 646–651, Jul. 2004.

[42] J. Sauvola and M. Pietikäinen, "Adaptive document image binarization," *Pattern Recognit.*, vol. 33, no. 2, pp. 225–236, Feb. 2000.

[43] G. Cao, J. Chen, and J. Jiang, "An adaptive approach to vehicle license plate localization," in *Proc. 29th Annu. Conf. IECON*, 2003, pp. 1786–1791.

[44] F. Yang and Z. Ma, "Vehicle license plate location based on histogramming and mathematical morphology," in *Proc. 4th IEEE Workshop Autom. Identification Advanced Technol.*, 2005, pp. 89–94.

[45] P. V. Suryanarayana, S. K. Mitra, A. Baneree, and A. K. Roy, "A morphology based approach for car license plate extraction," in *Proc. IEEE Indicon*, Chennai, India, Dec. 2005, pp. 24–27.

[46] J. Xiong, S. Du, D. Gao, and Q. Shen, "Locating car license plate under various illumination conditions using genetic algorithm," in *Proc. ICSP*, pp. 2502–2505.

[47] S. K. Kim, D. W. Kim, and H. J. Kim, "A recognition of vehicle license plate using a genetic algorithm based segmentation," in *Proc. Int. Conf. Image Process.*, Sep. 1996, vol. 1, pp. 661–664.

[48] V. Kamat and S. Ganesan, "An efficient implementation of the Hough transform for detecting vehicle license plates using DSP'S," in *Proc. Real-Time Technol. Appl. Symp.*, May 15–17, 1995, pp. 58–59.

[49] Y. Cheng, J. Lu, and T. Yahagi, "Car license plate recognition based on the combination of principal component analysis and radial basis function networks," in *Proc. ICSP*, pp. 1455–1458.

[50] C. R. Jung and R. Schramm, "Rectangle detection based on a windowed Hough transform," in *Proc. 17th SIBGRAPI*, 2004, pp. 113–120.

[51] E. R. Lee, P. K. Kim, and H. J. Kim, "Automatic recognition of a car license plate using color image processing," in *Proc. IEEE ICIP*, Nov. 13–16, 1994, vol. 2, pp. 301–305.

[52] S. Nomura, K. Yamanaka, O. Katai, H. Kawakami, and T. Shiose, "A novel adaptive morphological approach for degraded character image segmentation," *Pattern Recognit.*, vol. 38, no. 11, pp. 1961–1975, Nov. 2005.

[53] S. Nomura, K. Yamanaka, O. Katai, and H. Kawakami, "A new method for degraded color image binarization based on adaptive lightning on grayscale versions," *IEICE Trans. Inf. Syst.*, vol. E87-D, no. 4, pp. 1012–1020, Apr. 2004.

[54] P. Soille, *Morphological Image Analysis: Principles and Applications*. Berlin, Germany: Springer-Verlag, 1999.

[55] R. C. Gonzalez and R. E. Woods, *Digital Image Processing*. Reading, MA: Addison-Wesley, 1993.

[56] K. V. Mardia and T. J. Hainsworth, "A spatial thresholding method for image segmentation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 10, no. 6, pp. 919–927, Nov. 1988.

[57] G. Rudolph, "Convergence analysis of canonical genetic algorithms," *IEEE Trans. Neural Netw.*, vol. 5, no. 1, pp. 96–101, Jan. 1994.

[58] W. E. L. Grimson and T. Lozano-Perez, "Localizing overlapping parts by searching the interpretation tree," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. PAMI-9, no. 4, pp. 469–482, Jul. 1987.

[59] B. R. Lee, K. Park, H. Kang, H. Kim, and C. Kim, *Adaptive Local Binarization Method for Recognition of Vehicle License Plates*, vol. 3322, R. Klette and J. Žunić, Eds. New York: Springer-Verlag, 2004, pp. 646–655.

[60] A. Perez and R. C. Gonzalez, "An iterative thresholding algorithm for image segmentation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. PAMI-9, no. 6, pp. 742–751, Nov. 1987.

[61] J. R. Parker, "Gray level thresholding in badly illuminated images," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 13, no. 8, pp. 813–819, Aug. 1991.

[62] Y. Nakagawa and A. Rosenfeld, "Some experiments on variable thresholding," *Pattern Recognit.*, vol. 11, no. 3, pp. 191–204, 1979.

[63] J. Kittler, J. Illingworth, and J. Foglein, "Threshold selection based on a simple image statistic," *Comput. Vis. Graph. Image Process.*, vol. 30, no. 2, pp. 125–147, May 1985.

[64] K. B. Kim, S. W. Jang, and C. K. Kim, "Recognition of car license plate by using dynamical thresholding method and enhanced neural networks," in *Computer Analysis of Images and Patterns*, vol. 2756, N. Petkov and M. A. Westenberg, Eds. New York: Springer-Verlag, 2003, pp. 309–319.

[65] D. Llorens, A. Marzal, V. Palazon, and J. M. Vilar, *Car License Plates Extraction and Recognition Based on Connected Components Analysis and HMM Decoding*, vol. 3522, J. S. Marques *et al.*, Eds. New York: Springer-Verlag, 2005, pp. 571–578.

[66] L. R. Rabiner, "A tutorial on hidden Markov models and selected applications in speech recognition," *Proc. IEEE*, vol. 77, no. 2, pp. 257–286, Feb. 1989.

[67] S. Haykin, *Neural Networks: A Comprehensive Foundation*, 2nd ed. Englewood Cliffs, NJ: Prentice-Hall, 1999.

[68] E. K. Lim, H. K. Yang, and K. B. Kim, "Recognition of car license plate using Kohonen algorithm," in *Proc. ITC-CSCC*, 2000, vol. 2, pp. 785–788.

[69] C. Anagnostopoulos, E. Kayafas, and V. Loumos, "Digital image processing and neural networks for vehicle license plate identification," *J. Electr. Eng.*, vol. 1, no. 2, pp. 2–7, 2000. [Online]. Available: http://www.medialab.ntua.gr/people/canag/journals.php

[70] D. F. Specht, "Probabilistic neural networks," *Neural Netw.*, vol. 3, no. 1, pp. 109–118, 1990.

[71] C. M. Bishop, *Neural Networks for Pattern Recognition*. New York: Oxford Univ. Press, 1997.

[72] C. Anagnostopoulos, I. Anagnostopoulos, G. Tsekouras, G. Kouzas, V. Loumos, and E. Kayafas, "Using sliding concentric windows for license plate segmentation and processing," in *Proc. IEEE Workshop Signal Process. Syst. Des. Implementation*, Athens, Greece, 2005, pp. 337–342.

[73] Y. Hu, F. Zhu, and X. Zhang, *A Novel Approach for License Plate Recognition Using Subspace Projection and Probabilistic Neural Network*, vol. 3497, J. Wang, X. Liao, and Z. Yi, Eds. New York: Springer-Verlag, 2005, pp. 216–221.

[74] D. P. Huttenlokcer, G. A. Klanderman, and W. J. Rucklidge, "Comparing images using the Hausdorff distance," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 15, no. 9, pp. 850–863, Sep. 1993.

[75] J. Rucklidge, "Efficiently locating objects using the Hausdorff distance," *Int. J. Comput. Vis.*, vol. 24, no. 3, pp. 251–270, Sep./Oct. 1997.

[76] Y.-P. Huang, S.-Y. Lai, and W.-P. Chuang, "A template-based model for license plate recognition," in *Proc. IEEE Int. Conf. Netw., Sens. Control*, 2004, pp. 737–742.

[77] C. Wu, L. C. On, C. H. Weng, T. S. Kuan, and K. Ng, "A Macao license plate recognition system," in *Proc. 4th Int. Conf. Mach. Learn. Cybern.*, Guangzhou, China, Aug. 18–21, 2005, pp. 4506–4510.

[78] N. Bellas, S. M. Chai, M. Dwyer, and D. Linzmeier, "FPGA implementation of a license plate recognition SoC using automatically generated streaming accelerators," in *Proc. 20th IPDPS*, Nice, France, Apr. 2006, pp. 8–15.

[79] C. Oz and F. Ercal, "A practical license plate recognition system for real-time environments," in *Proc. IWANN*, vol. 3512. Berlin, Germany: Springer-Verlag, 2005, pp. 881–888.

[80] P. Zhang and L. H. Chen, "A novel feature extraction method and hybrid tree classification for handwritten numeral recognition," *Pattern Recognit. Lett.*, vol. 23, no. 1–3, pp. 45–56, Jan. 2002.

[81] J. Kittler, H. Mohamad, and P. W. Robert, "On combining classifiers," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 3, no. 20, pp. 226–239, Mar. 1998.

[82] C. J. Ahmad and M. Shridhar, "Recognition of handwritten numerals with multiple feature and multistage classifier," *Pattern Recognit.*, vol. 2, no. 28, pp. 153–160, Feb. 1995.

[83] L. Xu, A. Krzyzak, and C. Y. Suen, "Methods of combining multiple classifiers and their application to handwriting recognition," *IEEE Trans. Syst., Man, Cybern.*, vol. 22, no. 3, pp. 418–435, May/Jun. 1992.

[84] Y. S. Huang and C. Y. Suen, "A method of combining multiple experts for the recognition of unconstrained handwritten numerals," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 17, no. 1, pp. 90–93, Jan. 1995.

[85] H. J. Kang and J. Kim, "Probabilistic framework for combining multiple classifier at abstract level," in *Proc. 4th Int. Conf. Document Anal. Recog.*, Ulm, Germany, Aug. 1997, vol. 1, pp. 870–874.

[86] X. Pan, X. Ye, and S. Zhang, "A hybrid method for robust car plate character recognition," *Eng. Appl. Artif. Intell.*, vol. 18, no. 8, pp. 963–972, Dec. 2005.

[87] J. Kong, X. Liu, Y. Lu, and X. Zhou, "A novel license plate localization method based on textural feature analysis," in *Proc. IEEE Int. Symp. Signal Process. Inf. Technol.*, Athens, Greece, 2005, pp. 275–279.

[88] G. Li, R. Zeng, and L. Lin, "Research on vehicle license plate location based on neural networks," in *Proc. 1st ICICIC*, Beijing, China, 2006, pp. 174–177.

[89] H. Mahini, S. Kasaei, F. Dorri, and F. Dorri, "An efficient features–based license plate localization method," in *Proc. 18th ICPR*, Hong Kong, 2006, vol. 2, pp. 841–844.

[90] V. Shapiro and G. Gluhchev, "Multinational license plate recognition system: Segmentation and classification," in *Proc. 17th ICPR*, Cambridge, U.K., 2004, vol. 4, pp. 352–355.

[91] V. Franc and V. Hlavac, *License Plate Character Segmentation Using Hidden Markov Chains*, vol. 3663. Berlin, Germany: Springer-Verlag, 2005, pp. 385–392.
[92] M. I. Schlesinger and V. Hlavac, *Ten Lectures on Statistical and Structural Pattern Recognition*. Norwell, MA: Kluwer, 2002.
[93] Y. Cheng, "Mean shift, mode seeking, and clustering," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 17, no. 8, pp. 790–799, Aug. 1995.
[94] D. Comaniciu and P. Meer, "Mean shift: A robust approach toward feature space analysis," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 24, no. 5, pp. 603–619, May 2002.
[95] W. Jia, H. Zhang, X. He, and M. Piccardi, "Mean shift for accurate license plate localization," in *Proc. 8th Int. IEEE Conf. Intell. Transp. Syst.*, Vienna, Austria, Sep. 13–16, 2005, pp. 566–571.
[96] W. Jia, H. Zhang, X. He, and Q. Wu, "Gaussian weighted histogram intersection for license plate classification," in *Proc. 18th ICPR*, Hong Kong, 2006, vol. 3, pp. 574–577.
[97] M. J. Swain and D. H. Ballard, "Color indexing," *Int. J. Comput. Vis.*, vol. 1, no. 7, pp. 11–32, Nov. 1991.
[98] K.-M. Wong, C.-H. Cheung, and L.-M. Po, "Merged color histogram for color image retrieval," in *Proc. Int. Conf. Image Process.*, 2002, vol. 3, pp. 949–952.
[99] P. Wu, H.-H. Chen, R.-J. Wu, and D.-F. Shen, "License plate extraction in low resolution video," in *Proc. 18th ICPR*, Hong Kong, 2006, vol. 1, pp. 824–827.
[100] S. Kim, D. Kim, Y. Ryu, and G. Kim, "A robust license-plate extraction method under complex image conditions," in *Proc. 16th Int. Conf. Pattern Recog.*, Quebec, QC, Canada, 2002, vol. 3, pp. 216–219.
[101] Z. Ma and J. Yang, "A license plate locating algorithm based on multiple Gauss filters and morphology mathematics," in *Proc. 24th IASTED Int. Multiconference, Signal Process., Pattern Recog. Appl.*, Insbruck, Austria, Feb. 15–17, 2006, pp. 90–94.
[102] Y.-N. Chen, C.-C. Han, C.-T. Wang, B.-S. Jeng, and K.-C. Fan, "The application of a convolution neural network on face and license plate detection," in *Proc. 18th ICPR*, Hong Kong, 2006, vol. 3, pp. 552–555.
[103] C. Garcia and M. Delakis, "Convolutional face finder: A neural architecture for fast and robust face detector," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 26, no. 11, pp. 1408–1423, Nov. 2004.
[104] H. A. Rowley, S. Baluja, and T. Kanade, "Neural network based face detection," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 20, no. 1, pp. 23–38, Jan. 1998.
[105] S. R. Deans, "Hough transform from the radon transform," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. PAMI-3, no. 2, pp. 185–188, 1981.
[106] I. K. Yalçın and M. Gökmen, "Integrating differential evolution and condensation algorithms for license plate tracking," in *Proc. 18th ICPR*, Hong Kong, 2006, vol. 1, pp. 658–661.
[107] M. Isard and A. Blake, "Condensation—Conditional density propagation for visual tracking," *Int. J. Comput. Vis.*, vol. 29, no. 1, pp. 5–28, Aug. 1998.
[108] R. Storn and K. Price, "Differential evolution—A simple and efficient adaptive scheme for global optimization over continuous spaces," Int. Comput. Sci. Inst., Berkeley, CA, Tech. Rep. TR-95-012, 1995.
[109] Q. Wu, H. Zhang, W. Jia, X. He, J. Yang, and T. Hintz, "Car plate detection using cascaded tree-style learner based on hybrid object features," in *Proc. IEEE Int. Conf. AVSS*, Sydney, Australia, 2006, pp. 15–20.
[110] H. Zhang, W. Jia, X. He, and Q. Wu, "Learning-based license plate detection using global and local features," in *Proc. 18th ICPR*, Hong Kong, 2006, vol. 2, pp. 1102–1105.
[111] A. Capar and M. Gokmen, "Concurrent segmentation and recognition with shape-driven fast marching methods," in *Proc. 18th ICPR*, Hong Kong, 2006, vol. 1, pp. 155–158.
[112] J. A. Sethian, "A fast marching level set method for monotonically advancing fronts," in *Proc. Nat. Acad. Sci.*, Feb. 1996, vol. 93, pp. 1591–1595.
[113] P. Stec and M. Domanski, "Efficient unassisted video segmentation using enhanced fast marching," in *Proc. ICIP*, vol. 3, pp. 427–430.
[114] Y. Amit, D. Geman, and X. Fan, "A coarse-to-fine strategy for multiclass shape detection," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 26, no. 12, pp. 1606–1621, Dec. 2004.
[115] Y. Amit, "A neural network architecture for visual selection," *Neural Comput.*, vol. 12, no. 5, pp. 1059–1082, May 2000.
[116] Y. Amit and D. Geman, "A computational model for visual selection," *Neural Comput.*, vol. 11, no. 7, pp. 1691–1715, Oct. 1999.
[117] W. K. Pratt, *Digital Image Processing: PIKS Inside*, 3rd ed. Hoboken, NJ: Wiley, 2001.
[118] D. E. Rumelhart and J. L. McClelland, "Parallel distributed processing," in *Explorations in the Microstructure of Cognition*, vol. 1/2. Cambridge, MA: MIT Press, 1986.
[119] O. D. Trier, A. K. Jain, and T. Taxt, "Feature extraction methods for character recognition—A survey," *Pattern Recognit.*, vol. 29, no. 4, pp. 641–661, Apr. 1996.
[120] S. H. Park, K. I. Kim, K. Jung, and H. J. Kim, "Locating car license plates using neural networks," *Electron. Lett.*, vol. 35, no. 17, pp. 1475–1477, Aug. 1999.
[121] A. Waibel, T. Hanazawa, G. Hinton, K. Shikano, and K. J. Lang, "Phoneme recognition using time-delay neural networks," *IEEE Trans. Acoust., Speech Signal Process.*, vol. 37, no. 3, pp. 328–339, Mar. 1989.
[122] A. S. French and R. B. Stein, "A flexible neural analog using integrated circuits," *IEEE Trans. Biomed. Eng.*, vol. BME-17, no. 3, pp. 248–337, 1970.
[123] H. J. Reitboeck, R. Eckhorn, M. Arndt, and P. Dicke, "A model of feature linking via correlated neural activity," in *Synergetics of Cognition*, H. Haken, Ed. New York: Springer-Verlag, 1989, pp. 112–125.
[124] J. L. Johnson and M. L. Padgett, "PCNN models and applications," *IEEE Trans. Neural Netw.*, vol. 10, no. 3, pp. 480–498, May 1999.
[125] A. Taleb-Ahmed, D. Hamad, and G. Tilmant, "Vehicle license plate recognition in marketing application," in *Proc. Intell. Vehicles Symp.*, 2003, pp. 90–94.
[126] W. Wei, Y. Li, M. Wang, and Z. Huang, "Research on number-plate recognition based on neural networks," in *Proc. IEEE Signal Process. Soc. Workshop Neural Netw. Signal Process. XI*, pp. 529–538.
[127] C. Coetzee, C. Botha, and D. Weber, "PC based number plate recognition system," in *Proc. IEEE Int. Symp. Ind. Electron.*, 1998, pp. 605–610.
[128] B. Lim, W. Yeo, K. Tan, and C. Teo, "A novel DSP based real time character classification and recognition algorithm for car plate detection and recognition," in *Proc. Int. Conf. Signal Process.*, 1998, pp. 1269–1272.
[129] C. Opas, L. Li, and X. Qian, "Automatic license number extraction and its parallel implementation," in *Proc. 25th Annu. Conf. Ind. Electron. Soc.*, 1999, pp. 704–709.
[130] H. Hegt, R. De la Haye, and N. Khan, "A high performance license plate recognition system," in *Proc. IEEE Int. Conf. Syst., Man, Cybern.*, pp. 4357–4362.
[131] D. Yan, M. Hongqing, L. Jilin, and L. Langang, "A high performance license plate recognition system based on the web technique," in *Proc. Conf. Intell. Transp. Syst.*, 2001, pp. 325–329.
[132] T. Sirithinaphong and K. Chamnongthai, "The recognition of car license plate for automatic parking system," in *Proc. 5th Int. Symp. Signal Process. Appl.*, 1999, pp. 455–457.
[133] M. Raus and L. Kreft, "Reading car license plates by the use of artificial neural networks," in *Proc. 38th Midwest Symp. Circuits Syst.*, 1995, pp. 538–541.
[134] A. Rahman, W. Badawy, and A. Radmanesh, "A real time vehicle's license plate recognition system," in *Proc. IEEE Conf. Advanced Video Signal Based Surveillance*, pp. 163–166.
[135] T. Sirithinaphong and K. Chamnongthai, "Extracting of car license plate using motor vehicle regulation and character pattern recognition," in *Proc. IEEE Asia-Pacific Conf. Circuits Syst.*, pp. 559–562.
[136] S. Grossberg, "Competitive learning: From interactive activation to adaptive resonance," *Cogn. Sci.*, vol. 11, no. 1, pp. 23–63, Jan./Mar. 1987.
[137] J. Barroso, E. Dagless, A. Rafael, and J. Bulas-Cruz, "Number plate reading using computer vision," in *Proc. IEEE Int. Symp. Ind. Electron.*, pp. 761–766.
[138] M. Rouhani, "A fuzzy feature extractor neural network and its application in license plate recognition," in *Computational Intelligence, Theory and Application*. Berlin, Germany: Springer-Verlag, Sep. 9, 2006, pp. 223–228.
[139] J. Buckley and Y. Hayashi, "Neural nets for fuzzy systems," *Fuzzy Sets Syst.*, vol. 71, no. 3, pp. 265–276, May 1995.
[140] H. Kwan and Y. Cai, "A fuzzy neural network and its application to pattern recognition," *IEEE Trans. Fuzzy Syst.*, vol. 2, no. 3, pp. 185–193, Aug. 1994.
[141] V. S. Petrovic and T. F. Cootes, "Analysis of features for rigid structure vehicle type recognition," in *Proc. Brit. Mach. Vis. Conf.*, 2004, vol. 2, pp. 587–596.
[142] V. S. Petrovic and T. F. Cootes, "Vehicle type recognition with match refinement," in *Proc. 17th Int. Conf. Pattern Recog.*, Aug. 23–26, 2004, vol. 3, pp. 95–98.
[143] H. J. Lee, *Neural Network Approach to Identify Model of Vehicles*, vol. 3973. New York: Springer-Verlag, 2006, pp. 66–72.
[144] T. H. Chang, C. H. Lin, C. S. Hsu, and Y. J. Wu, "A vision-based vehicle behavior monitoring and warning system," in *Proc. Intell. Transp. Syst.*, 2003, vol. 1, pp. 448–453.

[145] G. Garibotto, *A Binocular License Plate Reader for High Precision Speed Measurement*, vol. 2059, C. Arcelli *et al.*, Eds. Berlin, Germany: Springer-Verlag, 2001, pp. 749–758.

[146] H. Lee, D. Kim, D. Kim, and S. Y. Bang, “Real-time automatic vehicle management system using vehicle tracking and car plate number identification,” in *Proc. ICME*, vol. 2, pp. II/353–II/356.

[147] J. S. Kang, M. H. Kang, C. H. Park, J. H. Kim, and Y. S. Choi, “Implementation of embedded system for vehicle tracking and license plates recognition using spatial relative distance,” in *Proc. 26th Int. Inf. Technol. Interfaces*, 2004, vol. 1, pp. 167–172.

[148] Date of access, April 26, 2007. [Online]. Available: http://itsdeployment2.ed.ornl.gov/technology_overview/

[149] C. Anagnostopoulos, T. Alexandropoulos, V. Loumos, and E. Kayafas, “Intelligent traffic management through MPEG-7 vehicle flow surveillance,” in *Proc. IEEE Int. Symp. Modern Comput. JVA*, Oct. 2006, pp. 202–207.

[150] F. Porikli and T. Kocak, “Robust license plate detection using covariance descriptor in a neural network framework,” in *Proc. IEEE Int. Conf. AVSS*, pp. 107–112.

[151] K. V. Suresh, G. Mahesh Kumar, and A. N. Rajagopalan, “Superresolution of license plates in real traffic videos,” *IEEE Trans. Intell. Transp. Syst.*, vol. 8, no. 2, pp. 321–331, Jun. 2007.

[152] C. Arth, F. Limberger, and H. Bischof, “Real-time license plate recognition on an embedded DSP-platform,” in *Proc. IEEE Conf. CVPR*, Jun. 17–22, 2007, pp. 1–8.

[153] V. Shapiro, G. Gluhchev, and D. Dimov, “Towards a multinational car license plate recognition system,” *Mach. Vis. Appl.*, vol. 17, no. 3, pp. 173–183, Aug. 2006.

[154] A. Mecocci and C. Tommaso, “Generative models for license plate recognition by using a limited number of training samples,” in *Proc. IEEE Int. Conf. Image Process.*, Oct. 8–11, 2006, pp. 2769–2772.

[155] S. Rajaram, M. D. Gupta, N. Petrovic, and T. S. Huang, “Learning-based nonparametric image super-resolution,” in *EURASIP J. Appl. Signal Process.*, vol. 2006, Jan. 2006, p. 229.

[156] S. Z. Wang and H. J. Lee, “A cascade framework for a real-time statistical plate recognition system,” *IEEE Trans. Inf. Forensics Security*, vol. 2, no. 2, pp. 267–282, Jun. 2007.

[157] P. Viola and M. Jones, “Robust real-time face detection,” *Int. J. Comput. Vis.*, vol. 57, no. 2, pp. 137–154, May 2004.

Christos-Nikolaos E. Anagnostopoulos (M’04) was born in Athens, Greece, in 1975. He received the diploma in mechanical engineering and the Ph.D. degree from the National Technical University of Athens in 1998 and 2002, respectively.

In 2003, he joined the University of the Aegean, Mytilene, Greece, as a Lecturer with the Department of Cultural Technology and Communication. He is the author of more than 60 papers in journals and conference proceedings. His research interests include image processing, computer vision, neural networks, and artificial intelligence.

Dr. Anagnostopoulos is a member of the Greek Chamber of Engineers.

Ioannis E. Anagnostopoulos (M’04) was born in Athens, Greece, in 1975. He received the diploma in electrical and computer engineering from the University of Patras, Patras, Greece, in 1998 and the Ph.D. degree from the National Technical University of Athens in 2004.

In 2004, he joined the University of the Aegean, Karlovassi, Greece, as a Visiting Lecturer with the Department of Information and Communication Systems Engineering. He is the author of more than 60 papers in journals and conference proceedings. His research interests are in the fields of web information management, e-commerce, communication networks, image processing, medical information systems, and multimedia systems and software.

Dr. Anagnostopoulos is a member of the Greek Chamber of Engineers.

Ioannis D. Psoroulas was born in Athens, Greece, in 1972. He received the diploma in chemical engineering and the Ph.D. degree from the National University of Athens in 1998 and 2007, respectively.

Since 2001, he has been working as a Software Developer/Analyst with various IT companies and with national and international research projects. He is currently with the School of Electrical and Computer Engineering, National Technical University of Athens. His research interests are in the fields of simulation systems, internet technologies, multimedia systems, and innovative programming techniques.

Vassili Loumos (M’97) received the diploma in electrical engineering and the Ph.D. degree in computer science from the National Technical University of Athens (NTUA), Athens, Greece.

He was a Research Engineer with the Centre National de Recherhe Scientifique, Paris, France, for four years, and was a Field Engineer with CGEE ALSHTOM for three years, before joining academia. He became a Faculty Member with the School of Electrical and Computer Engineering, NTUA, in 1990, where he is currently a Professor, teaching multimedia technologies and computer graphics. In parallel, he is running the Multimedia Technology Laboratory, NTUA, which he founded in 1995. His research and development activities focus on the fields of image processing and computer vision, Internet technologies and applications, and computer networks.

Dr. Loumos is a member of the Greek Chamber of Engineers.

Eleftherios Kayafas (M’97) received the B.Sc. degree from Athens University, Athens, Greece, in 1970 and the M.Sc. and Ph.D. degrees in electrical engineering from the University of Salford, Salford, U.K., in 1975 and 1978, respectively.

He was with the Civil Aviation Service from 1973 to 1974. After his postgraduate studies, he was with the Hellenic Aerospace Industry from 1978 to 1979. In 1979, he joined the National Technical University of Athens as a Lecturer with the School of Electrical and Computer Engineering, where he has since then become an Assistant Professor (1987), an Associate Professor (1992), and, finally, a Professor of applied electronics (1996). He is the author of more than 140 papers in journals and conference proceedings. His research interests include applied electronics, multimedia applications, and multimedia communication systems.

Dr. Kayafas is a member of the Greek Chamber of Engineers and the International Measurement Confederation (IMEKO), where he is a member of the TC-4 committee.