Car License Plate Recognition with Neural Networks and Fuzzy Logic.

J.A.G. Nijhuis, M.H. ter Brugge, K.A. Helmholt,
J.P.W. Pluim, L. Spaanenburg, R.S. Venema, M.A. Westenberg.
University of Groningen, Department of Computing Science
P.O. Box 800, NL–9700 AV Groningen, The Netherlands
phone x–31–50–637125, email: jos@cs.rug.nl

ABSTRACT

A car license plate recognition system (CLPR–system) has been developed to identify vehicles by the contents of their license plate for speed–limit enforcement. This type of application puts high demands on the reliability of the CLPR–system. A combination of neural and fuzzy techniques is used to guarantee a very low error rate at an acceptable recognition rate. First experiments along highways in the Netherlands show that the system has an error rate of 0.02% at a recognition rate of 98.51%. These results are also compared with other published CLPR–systems.

1. Introduction

The automatic identification of vehicles by the contents of their license plate is important in a number of applications such as: traffic data collection, managing parking lot traffic, collection of motorway charges and weight– and speed–limit enforcement. Especially the use of a CLPR–system for enforcement purposes puts not only high demands on the reliability of the system [1],[2], but also requires that the complete car should be visible in the image. This last requirement results in a very low resolution for the characters on the license plate as the plate occupies only a smaller part of the complete scene. Typical character resolutions are 15 x 10 pixels. In this respect the CLPR–system described in this paper differs from the approaches presented in [3], [4], [5], [6] where the license plate occupies up to 25% of the complete image resulting in characters up to 38 x 47 pixels.
The limited resolution of the recorded characters together with dirt (dust, flies), screws and bolts (used to attach the plate to the car), overhanging car parts (tow bar mounted too high), etc. make the development of a reliable optical character recognizer (OCR) very complicated. Many of the approaches common in conventional OCR systems [7] turn out to be useless. It is therefore necessary to build a system that is capable of exploiting both rules defined by the license plate registration regulation (explicit knowledge) [8] and features present in the measurement of the OCR task on hand (implicit knowledge) [9]. Fuzzy logic and neural networks have demonstrated good performance in capturing both types of knowledge.
The design of the CLPR–system is based on the Dutch license plate registration regulation. This regulation prescribes a set of strict rules for the position of the license plate on the car, the shape, size, color of the plate and the character font (see figure 1), etc.

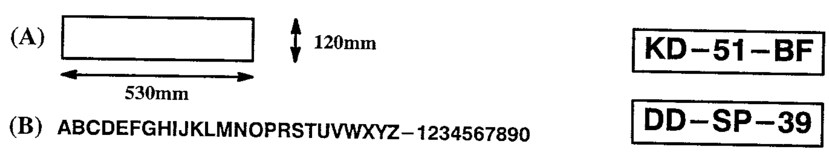

Figure 1: License plates with black characters on a yellow retroflexive background according to model 18.1 (A) with characters according to model B.1 (B) [RVW 6711]. The height of the characters varies from 75mm to 77mm (10mm for the horizontal bar). The width (margins included) varies from 32mm to 87mm. The characters are located symmetrically on the license plate.

Section 2 gives an outline of the complete CLPR–system. The licence plate segmentation is discussed in section 3. The reading of the contents of the license plate is described in section 4. Section 5 presents the syntactical analyzer. The last section gives a comparison with other CLPR–systems and draws some conclusions.

2. System Outline

The CLPR–system consists of four parts: a preprocessor, a segmentation unit, a recognizer and a syntactical analyzer as pictured in figure 2. The preprocessor applies some standard image processing techniques (contrast stretching, noise filtering, etc.) to enhance the quality of an image. This part of the system will not be elucidated any further in this paper as the desired preprocessing depends strongly on characteristics of the camera unit. The second part of the CLPR–system is the segmentation unit that determines the location of the license plate in the image. The segmentation is based on a fuzzy clustering algorithm that uses characteristics like yellowness and texture.

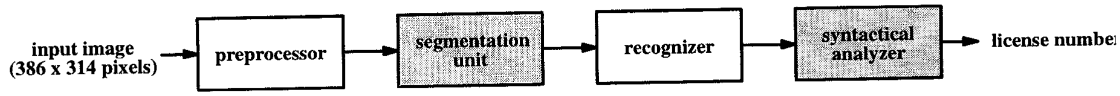

Figure 2: Outline of the recognition system. The recognizer module uses neural techniques whereas the gray shaded units are realized using fuzzy logic techniques.

The reading of the characters on the license plate is performed by the recognizer. This part of the CLPR–system consists of a combination of discrete–time cellular neural networks (DTCNN's) for feature extraction and an ordinary multi–layer perceptron network (MLP) to perform the classification. The last part of the system, the syntactical analyzer, checks whether, the candidate characters returned by the recognizer satisfy a number of (fuzzy) syntactic rules as there exist for Dutch license plates.

Figure 3: Some input examples for the CLPR–system. The images are recorded by a super–VHS camera. Each image contains 386 x 314 pixels with 8 bit color information per pixel.

3. License Plate Segmentation

During the license plate segmentation the input image, in which a vehicle is present, is searched for the region that contains the license plate. Several authors have proposed solutions for this problem [10], [11]. All of these methods incorporate a number of fuzzy features of the license plate like: "a license plate is a *yellow rectangular* area on the car, which contains a number of *black characters*" and "license plates are *mostly* mounted in the *center* of the *lower parts* of the front and rear of a car". Inspired by these rules two fuzzy properties, yellowness and texture, have been appointed to each pixel in the image. The membershipfunction for yellowness is determined by using a histogram based method. First, the RGB–values of pixels taken from a large number of hand–cut license plates are used to construct a frequency table. For each RGB–value this table gives the number of occurrences of this particular color in the set of hand–cut license plates. The membership function is derived directly from this table by normalizing it so that the most frequently used color has a membership degree of 1. Contrary to yellowness, texture is a "global" characteristic. The texture of a pixel is determined on the basis of the gray scale value of its 8–neighbors.

The segmentation is performed using a fuzzy c–means clustering algorithm, with the desired number of clusters set to two (license plate and non–license plate). The algorithm favors clusters that have a rectangular shape with the correct aspect ratio. Figure 4 shows some results of the fuzzy segmentation.

The second step in the segmentation involves the isolation of the individual characters from the license plate. First, a global thresholding is applied based on the average gray scale value of the 100 pixels with the largest gradient value. On the resulting binary image a connected component search is performed. Based on rules concerning the minimal area, width, height of characters, a connected component is marked as a potential character. The two bars between the character pairs are removed by these rules as they do not obey these rules.

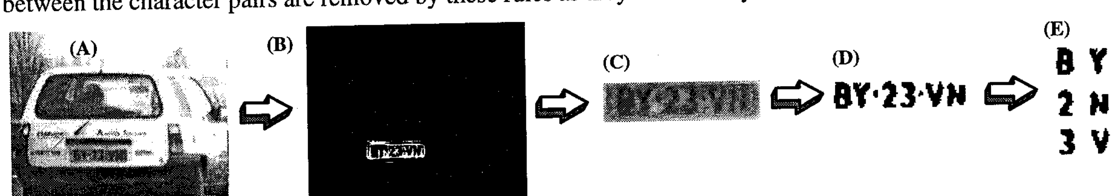

Figure 4: The various stages in the segmentation process. (A) the original image. (B) the result of the fuzzy clustering algorithm. The region that contains the license plate is clearly visible and can be used to extract the license plate (C) from the original image. (D) the license plate after global thresholding. (E) the accepted connected components as potential characters.

The selected components are only passed on to the recognizer module if they make up a valid license plate. A valid license plate contains six selected components that are all of the same height and start at the same vertical position. In all other cases the licence plate is marked by the system as unrecognizable and therefore rejected. The current systems rejects about 24.6% of all images during the segmentation stage.

4. Character Recognition

The next module in the CLPR–system is the recognizer. The binary connected components are used as inputs for this part of the system. Two different types of neural networks are used for the recognizer. Unlike the approaches

presented in [4], [5], [12] the classification is done on the basis of features instead of a bit wise image input. Extensive experiments showed that it is rather impossible for bit wise image input to obtain an error rate below the 1% for the low resolution characters in our system.
By means of DTCNN's four different of features are generated: horizontal projection, vertical projection, horizontal connected component count and vertical connected component count. For each feature a specific DTCNN has been constructed that is fed with a normalized version (the height of each character is scaled to 15 pixels, the width is scaled proportional to the height) of the binary connected components.

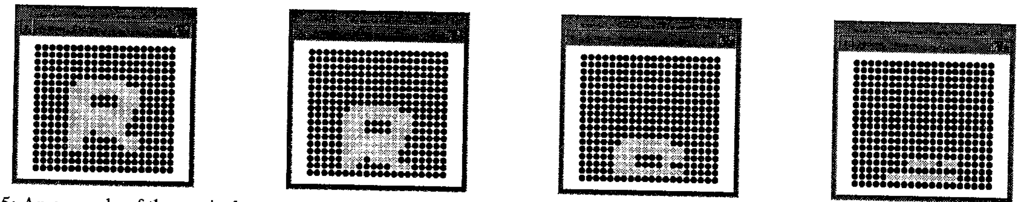

Figure 5: An example of the vertical connected component extraction by a DTCNN. Each dot in the screen captures corresponds to a neuron in the DTCNN. The vertical connected component count can be read after the DTCNN is converged as shown in the last screen shot.

Each of these features is transformed into five inputs for a MLP–neural network, resulting in a total of 24 inputs. A character is said to be recognized only if the output of the corresponding output neuron exceeds 0.85 and all other output neurons have an output level below 0.25.

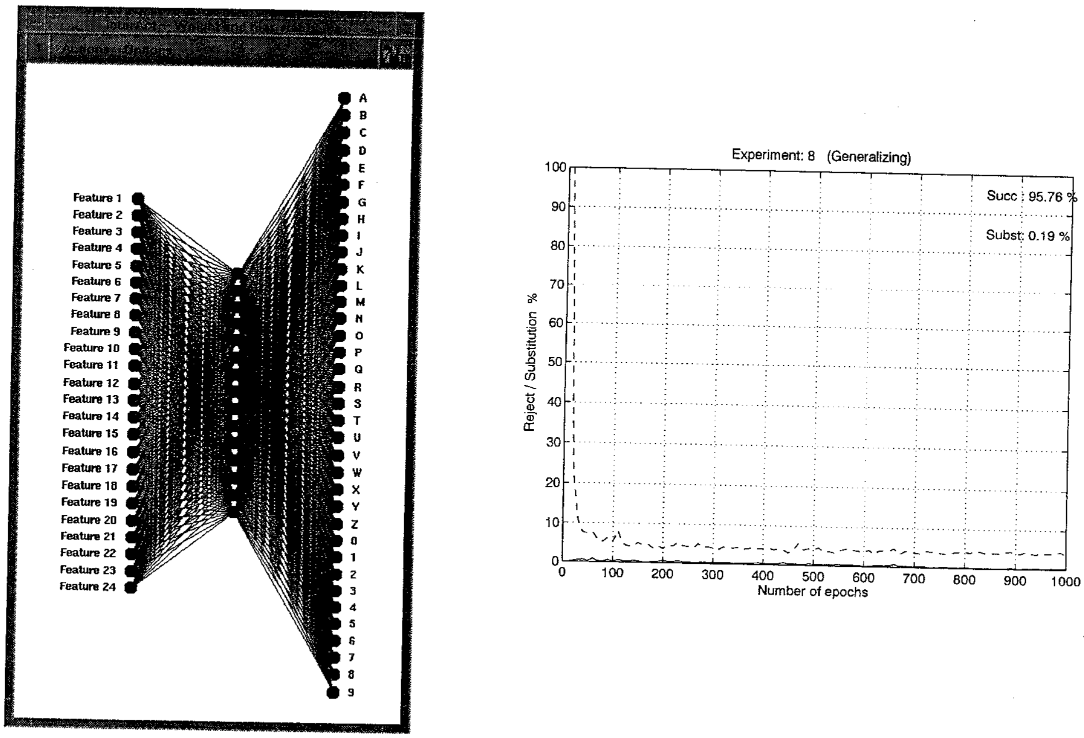

Figure 6: The MLP–neural network used for the classification of the characters. The neural network contains 24 input–, 15 hidden– and 36 output neurons (one output for each possible character on the license plate). Also shown is a generalization curve as function of the number of epoches. After only a few training cycles the network reaches a good performance. This is an indication that the input data contains very powerful features that can be easily learned by the neural network.

5. Syntactical Analysis

In the last step of the CLPR–system the results of the character recognition module are compared with some syntactical rules. These rules can be divided into two types. The first set of rules takes into account the spatial requirements of the characters as defined by the official guidelines. For Dutch license plates the spacing between characters should be zero. The second set of rules detects illegal combinations of numerical and alpha–numerical characters. Legal Dutch license plates contain only pairs of digits and letters, e.g. GD–85–DF, 56–FG–PR or KL–54–68. Furthermore there are some illegal letter pairs like the SS and SD. A candidate license number should pass both set of rules in order to be marked as recognized [12]. In case of any conflicts the image will be rejected. Be adding more rules to the syntactical analysis module it is possible to decrease the error rate at the cost of a higher rejection rate. In [8] the spatial requirements, and in [2] the syntactical requirements, are used to fill in or replace characters by more likely ones. Experiments showed that this should be done only as a last resort, as the recognition rate increases only marginally at the expense of a slightly higher error rate.

6. Results and Conclusions

In most articles the performance of the license plate recognition system is characterized by the recognition rate and error rate as defined by formulae (1) and (2).

$$\textit{recognition rate} = (\textit{number of correctly read characters}) / (\textit{number of found characters}) \quad (1)$$

$$\textit{error rate} = (\textit{number of badly read characters}) / (\textit{number of found characters}) \quad (2)$$

For an evaluation of the CLPR–systems it is necessary to state also the rejection rate as defined by formula (3).

$$\textit{rejection rate} = (\textit{number of rejected characters}) / (\textit{number of found characters}) \quad (3)$$

The number of correctly read, badly read and rejected characters should add up to the number of found characters. The rejection rate is although seldom mentioned. Table 1 contains a comparison between the car license plate recognition systems described be various authors. For some of the articles the recognition rate, error rate and rejection rate are defined in terms of license plates rather than characters. This will be indicated in the table.

reference	recognition rate / error rate	remarks
[1]	61.7% / 11.2% (low speed) 33.6% / 6.6% (high speed)	Evaluation of the 3M RIA–300 CLPR–system. Rates are based on license plates. Rejection rate of 27.1% at low car speeds and 59.8% at high car speeds.
[2]	63% / 37%	Results produced with syntax forcing. Rates are based on license plates.
[3]	76% / ?	Rate base on license plates (7 characters). Input: 512x512 gray–level image. Classification based on features.
[4]	95.5% / 0.9%	Input: 20x30 binary pixels for each character, MLP–network. Rejection rate 3.6%
[5]	98.2% / 1.8%	Segmentation and recognition with MLP–networks with pixel–input.
[6]	95% / 5%	Characters between 16x23 and 38x47 pixels. MLP–network with features input.
[8]	98.2 % / ? (digits) 96.1% / ? (Chinese–chars.)	Input: 512x480 gray level images. Use of a character alignment model to fill in obscured digits.
[9]	96.8% / 3.2%	Classification based on neural network with 12 input features. No details.
[13]	90% / ? (daytime) 65% / ? (nights)	Rates are based on license plates. Based on template matching.
[14]	97% / 1%	Input: 768x493 gray level images. Segmentation with fuzzy logic. Classification with MLP–network with pixel input (14x14) with additional top and bottom pixel inputs (14x9).
[15]	94% / ?	Classification with 12x16 pixel–input MLP–neural network.
this paper	98.51% / 0.02%	Segmentation with fuzzy logic. Classification with neural networks with feature inputs. Rejection rate 1.47%.

Table 1: A comparison between car license plate recognition systems.

The described CLPR–system has been tested on 10,000 different images. All images where processed by the system without checking for any unreadable or illegal licence plates. For the 75.4% of all images for which the CLPR–system decides that a license plate is present, a recognition rate of 98.51%, an error rate of 0.02% and a rejection rate of 1.47% is obtained per character. Summarizing, the complete system rejects 31.25% of all images as been unrecognizable ((100–75.4)% after segmentation + 75.4*6*1.47% after recognition and syntactic analysis). Note that the 31.25% also contains situations in which the license plate is illegal (handwritten, placed behind the car window, etc.).

The first tests with the presented system pointed out two error sources. The main reason for errors is caused by a wrong segmentation of the licence plate due to the presence of screws and bolts. A second error source is caused by the global thresholding step. Many of the errors made during the classification would also be made by a human if using the same binary input data. Experiments indicate that thresholding should be performed on a per–character basis.

References

[1] P. Davies, N. Emmott and N. Ayland, "License Plate Recognition Technology for Toll Violation Enforcement", in: *Proceedings of IEE Colloquium on Image analysis for Transport Applications*, Vol. 035, pp. 7/1–7/5, February 16, 1990.

[2] P. Williams, H. Kirby, F. Montgomery and R. Boyle, "Evaluation of Video–Recognition Equipment for Number–Plate Matching", in: *Proceedings of the 2nd International Conference on Road Traffic Monitoring*, London, Great Britain, pp. 89–93, February 7–9, 1989.

[3] R.A. Lotufo, A.D. Morgan and A.S. Johnson, "Automatic Number–Plate Recognition", in: Proceedings of the IEE Colloquium on Image analysis for Transport Applications, Vol. 035, pp. 6/1–6/6, February 16, 1990.

[4] F. Lisa, J. Carrabina, C. Pérez–Vincente, N. Avellana and E. Valderrama, "Two–bit Weights Are Enough To Solve Vehicle License Recognition Problem", in: *Proceedings of the International Conference on Neural Networks*, San Francisco, CA, Vol. 3, pp. 1242–1246, April 1993.

[5] R. Föhr and M. Raus, "Automatisches Lesen amtlicher Kfz–Kennzeichen", *Elektronik*, (in German), No. 1, pp. 60–64, January 1994.

[6] J. Yoo, B. CHun and D. Shin, "A Neural Network for Recognizing Characters Extracted from Moving Vehicles", in: *Proceedings of World Congress on Neural Networks*, San Diego, CA, Vol. 3, pp. 162–166, June 5–9, 1994.

[7] S. Mori, C.Y. Suen and K. Yamamoto, "Historical Review of OCR Research and Development", *Proceedings of the IEEE*, Vol. 80, No. 7, pp. 1029–1057, July 1992.

[8] K. Tanabe, E. Marubayashi, H. Kawashima, T. Nakanishi and A. Shio, "PC–Based Car License Plate Reading", *Image and Video Processing II*, SPIE Vol. 2182, pp. 220–231, 1994.

[9] S. Abe and M. Lan, "A Classifier Using Fuzzy Rules Extracted Directly from Numerical Data", in: *Proceedings of the Second IEEE International Conference om Fuzzy Systems*, San Francisco, CA, Vol. 2, pp. 1191–1198, March 28 – April 1, 1993.

[10] T. Agui, H.J. Choi and M. Nakajima, "Method of Extracting Car Number Plates by Image Processing", *Systems and computers in Japan*, Vol. 19, No. 3, pp. 46–52, March 1988.

[11] A. Postolache, J. Trecat, "License Plate Segmentation Based on a coarse Texture Approach for Traffic Monitoring System", in: *Proceedings of the ProRISC & IEEE–Benelux workshop on Circuits, Systems and Signal Processing*, Mierlo, The Netherlands, pp.243–250, March 23–24, 1995.

[12] J.A.G. Nijhuis, M.H. ter Brugge, H. Hettema and L. Spaanenburg, "License Plate Recognition by Fuzzy Logic", in: *Proceedings of the 5th Aachener Fuzzy Symposium*, Aachen, Germany, pp.147–153, April 25, 1995.

[13] K. Kanayama, Y. Fujikawa, K. Fujimoto and M. Horino, "Development of Vehicle–License Number Recognition System using Real–Time Image Processing and its Application to Travel–Time Measurement", in: *Proceedings of the 41st IEEE Vehicular Technology Conference*, St. Louis, MO, pp. 798–804, May 19–22, 1991.

[14] C. Hwang, S. Shu, W. Chen, Y. Chen and K. Wen, "A PC–Based Licence Plate Reader", in: *Proceedings Machine Vision Applications, Architectures and Systems Integration*, Boston, MA, Vol. SPIE 1823, pp. 272–283, November 17–18, 1992.

[15] A. Kertész, V. Kertész and T. Müller, "An On–line Image Processing System for Registration Number Identification", in: *Proceedings of IEEE International Conference on Neural Networks*, Orlando, FL, Vol. 6, pp. 4145–4147, June 28 – July 2, 1994.