A High Performance License Plate Recognition System

Hans A. Hegt, Ron J. De la Haye, Nadeem A. Khan
Department of Electrical Engineering
Eindhoven University of Technology
P.O.box 513, 5600 MB, Eindhoven, The Netherlands
E-mail: j.a.hegt@ele.tue.nl

ABSTRACT

This paper presents a powerful automated license plate recognition system, which is able to read license numbers of cars, even under circumstances, which are far from ideal. In a real-life test, the percentage of rejected plates was 13%, whereas 0.4% of the plates were misclassified. Suggestions for further improvements are given.

1. INTRODUCTION

The many useful applications for a license plate recognition system, such as operating a no-stop tolling system or a ticket-free parking lot, speed-limit enforcement, studying traffic habits, etc., have induced much research effort on the automation of these systems, such as described in [1-8].

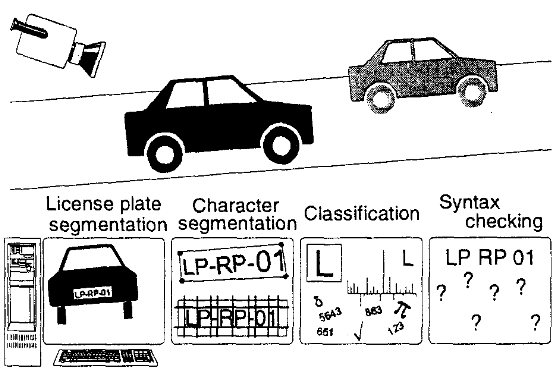

Fig.1: System overview

An overview of our system is visualised in fig. 1. In this system [9], a camera and a frame grabber deliver images of cars passing by, which are processed by a license plate segmenter, which will shortly be described in section 2. Due to perspective distortion, the image of the plate may not be rectangular. As outlined in section 2, rectangular license plates with a fixed size of 180×40 pixels are reconstructed by means of resampling. This image is further improved by a number of image enhancement steps. Next, as treated in section 3, the characters on the plate are approximately segmented and normalised with respect to contrast, intensity and size. The projection of each character image on a low dimensional space that contains all relevant information to distinguish it from other characters is done by the Hotelling transform, as described in section 4. The result of this transform depends on a good segmentation. We used this property, which is normally considered as a drawback, to improve the character segmentation step. Section 5 outlines the actual classification and syntax checking. Results from real-life tests are presented in section 6. In section 7 some conclusions and suggestions for further research are given. Our system was optimised for Dutch license plates, but can be adapted for other types of license plates as well.

2. LICENSE PLATE SEGMENTER

A high-speed shutter camera, combined with a frame grabber delivers images of 439×510 pixels of the backsides of cars passing by. Taking pictures of the backsides has a number of advantages:

- the lights at the back of a vehicle do not influence the images as much as the front lights;
- many trucks have signs at the front, which can be mistaken for license plates;
- drivers who want to avoid being seen by a camera can drive very close to the vehicle in front of them, so the license plate will then be obscured;
- the plate at the back is usually cleaner than the front plate.

The system tries to find a license plate in this image, by searching for a number of features [10]. Among other things, template matching is used to find the corner points of the plate. If four possible corner points are found, the content of the quadrangle is checked on its spatial frequencies. Certain spatial frequencies are expected due to the characters in a plate. Only in case this frequency content confirms its presence, the four corner points are accepted as being the corner points of a license plate. In this way, a powerful license plate segmenter was obtained, which is able to indicate the exact positions of the corner points with a maximum error of only a few pixels. Under normal circumstances, the average size of the license plate is about 100×20 pixels.

0-7803-4778-1/98 $10.00 © 1998 IEEE

Due to the perspective view, the found corner points may not correspond to a rectangle. The area described by the four corner points are transformed into a rectangular area by means of a bilinear transformation and converted to fixed dimensions of 180×40 pixels by means of bilinear interpolation. This is illustrated in fig.2. This figure shows that the bilinear transformation can correct for most of the perspective distortion, but it cannot correct for the distortion that is caused by bent plates.

Fig.2: Image with perspective distortion, corner points of license plate indicated (top) and the plate after the bilinear transformation and resampling (bottom).

3. CHARACTER SEGMENTATION

The horizontal segmentation of the characters of the license plate is based on finding the spaces between them. These spaces are found by examining the maxima of the column sums of the grey values of the license plate. Before these column sums are calculated, a special form of histogram stretching is applied, resulting in a more reliable column sum graph. An ideal image should at least contain a certain percentage of light grey pixels from the background of the plate. There should also be at least a certain percentage of dark pixels from the foreground in the image. By examining some images it was concluded that an image should at least contain about 60% light pixels and about 10% dark pixels.

Experiments showed that an image is enhanced for the character segmentation process by projecting 60% of the lightest pixels to absolute white. Projecting 10% of the darkest pixels to absolute black proved to distort the character segmentation. If this is done then dark areas in between characters are sometimes projected to absolute black, thus degrading the column sum graph. Projecting just 3% of the darkest pixels to absolute black proved to enhance the column sum graph. Linear histogram stretching is applied to the remaining 37% of the pixels. The extremes of the resulting vertical column sums are clearly more distinct, which simplifies the horizontal segmentation process, as shown at the bottom of fig.3.

In cases where the space between two characters cannot clearly be found, the positions of the left and right neighbouring spaces are used to estimate its location. The help of a modelled estimation of the probability of the segmentation position between a character pair proved to enhance the character segmentation a lot.

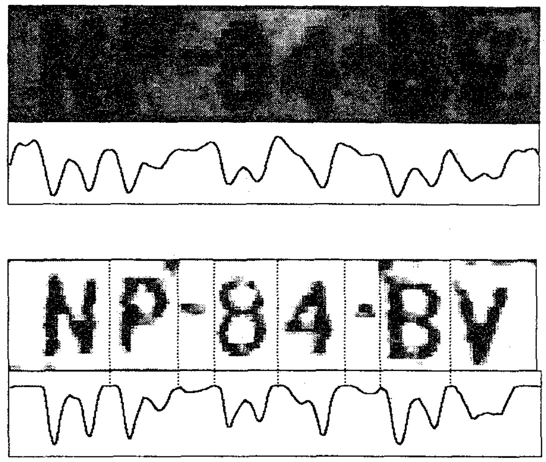

Fig.3: License plate before (top) and after (bottom) histogram stretching with their vertical column sum graphs.

The vertical positions of the characters are found with the help of an edge map and the corresponding grey value row sum (see fig.4). A high increase of this row sum indicates the top or bottom of the characters. Based on a-priori knowledge of the structure of the license plate, only a limited range of rows is searched for the top and bottom of the characters.

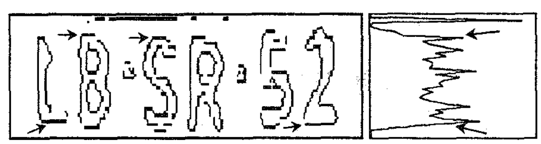

Fig.4: Edge map and its row sum

This segmentation procedure is able to segment characters with an accuracy of about two pixels horizontally and one pixel vertically in almost all cases. In the exceptional situations that a large segmentation error occurs, this can be detected by the method, described in the next section.

4. HOTELLING TRANSFORM

Many classification procedures for segmented characters can be found in literature. A straightforward method is based on template matching. Template matchers are basically comparing characters with prototypes. The prototype that matches best to the character "wins". Template matchers usually treat all pixels as equals. Changes in pixel values, if compared with the prototypes, will all contribute the same to the matching cost, independent of the position and the significance of the pixel. Obviously all pixels forming a character are not equally important to recognise the character. Template matchers are not aware of the specific features that distinguish one character from the other. They just look at the differences between a character and the prototypes. This is one of the major reasons why we have chosen another approach for our license plate recognition system.

In our system, we make use of the Hotelling transformation [11]. The Hotelling transformation can be considered as an orthonormal decomposition by projection onto the relevant eigenvectors of the correlation matrix. Thus the eigenvectors point in the directions of the maximum energy/variation of the prototypes. This way, only the information that distinguishes one prototype from another is kept. Non-relevant data for the recognition process is suppressed. This data reduction while inherently focussing on the most important information for the classification process was the most important reason why this method was adopted in our system. The number of relevant elements in the transformed vectors was estimated as 25. A more exact calculation of the optimal number of elements can for instance be obtained by the application of the generalised Fisher's criterion [12].

The character segmentation offers a segmented plate to the classification stage. This plate is first enhanced with the histogram stretching described in section IV. Again 3% of the darkest pixels are projected to black, however this time only 30% of the lightest pixels are projected to white. These values leave the grey areas surrounding the characters in tact. These areas also contain information about the characters. Then the character size, brightness and contrast are normalised. The character size is normalised based on the character height because this is most accurately found by the character segmentation stage. The character is resampled to a fixed height. The width is scaled with the same amount as to keep the same aspect ratio. Then the brightness and contrast of the resulting character is normalised. Of course the prototypes, that are needed for the classification, are treated in the same way. Prototypes were calculated based on the averages of a large number of normalised example characters per class.

The Hotelling transformation appeared to be very sensitive for character segmentation. This was studied by comparing sample characters in different positions with their reconstruction, obtained by first using the Hotelling transformation and then reconstructing the character by applying inverse Hotelling transformation. This is depicted in fig.5.

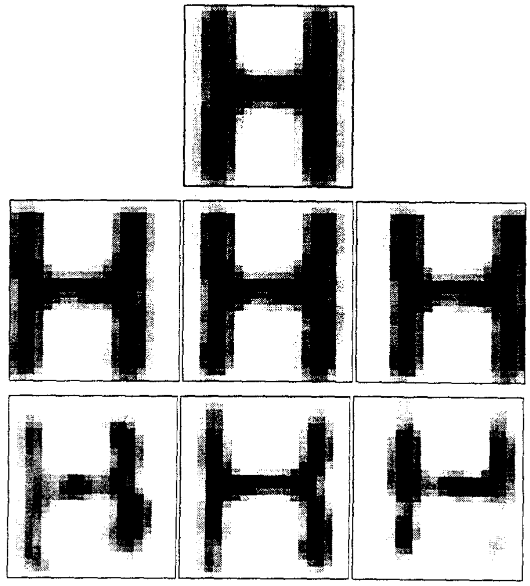

Fig.5: Prototype character 'H' (top), sample character 'H' at different positions (middle), with corresponding reconstructions (bottom).

This figure shows a significant deviation of the reconstructed characters with respect to the original samples, already in case of a small segmentation error. This seems to be an important disadvantage of the Hotelling transformation, but on the other hand, this phenomenon can also be exploited to indicate segmentation errors. Also other kinds of "distortions" can cause differences between the original and the reconstructed sample. We use the Euclidean distance between a sample and its reconstruction as an indicator of possible relevant distortion. This proved to be very useful in practice. Only in case this Euclidean distance is small enough, the Hotelling transformation result is accepted for the classification stage.

5. CLASSIFICATION AND SYNTAX CHECKING

The classification of the characters is based on a distance measure between their Hotelling transformed counterparts and the Hotelling transformed prototypes. The prototype, which is closest to the sample, is accepted. The distance to the second closest prototype gives information about how sure the classification is. By choosing the threshold, defined as one minus the distance ratio, between 0 and 1, the trade-off between misclassification and rejection can be controlled. This is illustrated in fig.6.

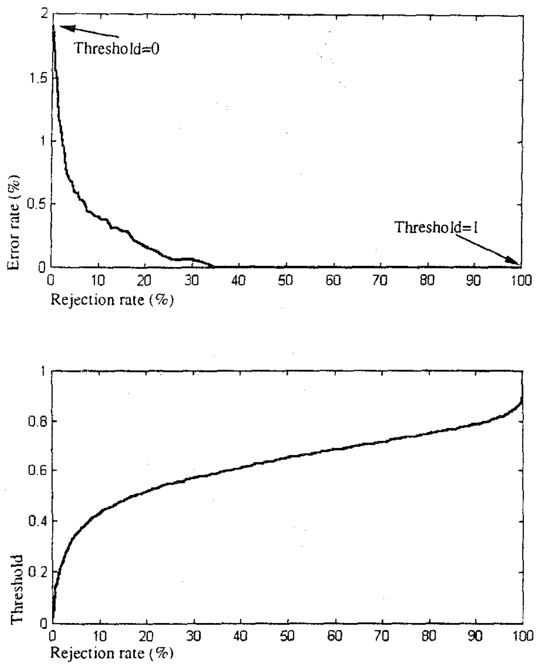

Fig.6: Error rate (top) and threshold (bottom) as a function of rejection rate.

The most important confusion classes found in practice (at a threshold of 0.35) were '8/B', '1/J' and '0/D'. As these all represent confusion between a digit and a letter, a-priori knowledge about the syntax of the license plates can often help solving those problems. According to the syntax of Dutch license plates for instance, characters are organised in pairs, each pair consisting of two digits or two letters. As long as confusion does not occur in both characters of such a pair, this kind of confusion can be detected, resulting for instance in a rejection of the license plate.

6. SYSTEM EVALUATION

A schematic overview of the system is given in fig.7. This system was tried on more than 1000 images, which were first recorded on a VHS tape in PAL format and were digitised afterwards. This extra recording step reduces the optical resolution of the images and introduces some noise. The performance of the system will probably increase if this step is omitted. Of the test set, 13% of the plates were rejected. Some of those plates were even completely or almost completely

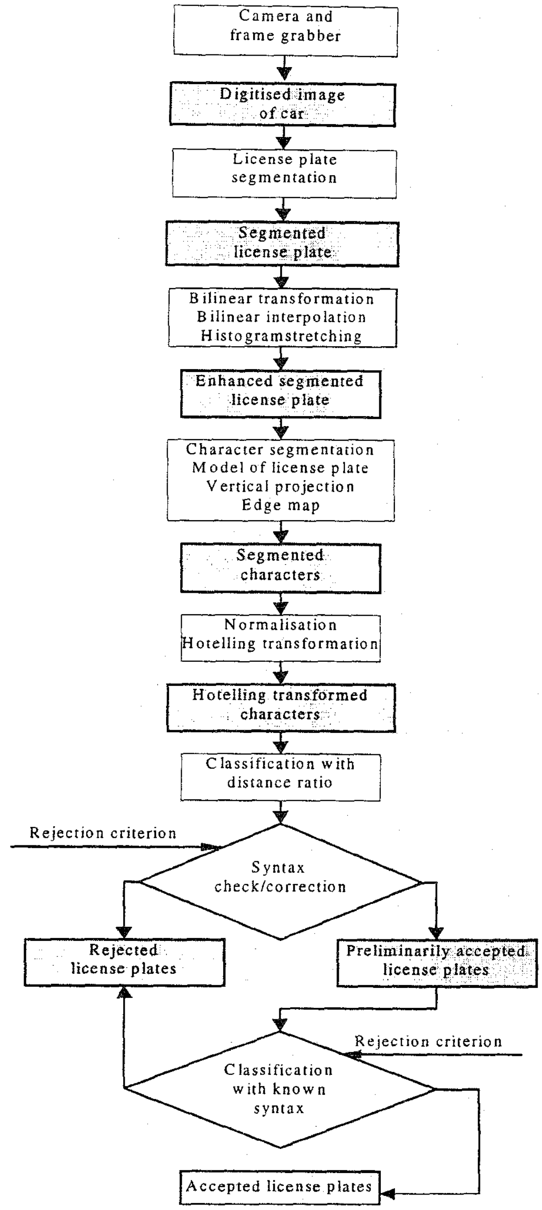

Fig7: Schematic overview of the license plate recognition system

unreadable by a human spectator. Of the remaining plates, 0.4% were misclassified. As was already mentioned in section 6, the threshold can be used to control the trade-off between the rejection and error rate.

7. CONCLUSIONS AND FURTHER RESEARCH

A license plate recognition system was presented, which applies the Hotelling transform to reduce the amount of data while inherently focussing on the most important information to discriminate between the character classes. Furthermore, by comparing the (non-transformed) characters with their forward-and-backward transformed counterparts, a useful indication about possible segmentation errors or other distortions can be obtained.

The character segmentation scheme based on the vertical projection performs well enough for the classification. However, segmentation should not be done independently of the classification, especially if the classification stage is able to correct for small segmentation errors. A co-operation of the classification and segmentation stage is most likely to perform very well. One can for example imagine a scheme where a segmented character is classified. The classifier then enhances the segmentation or even rejects it. This information is fed back to the segmentation stage, which responds accordingly.

The a-priori probability of a certain character for each character position in the license is not equal. This is caused by the rules that are applied for license plate numbers. Even certain character combinations occur more often than other combinations. At this time no use of these properties is made. Using them could improve the system performance.

The syntax checker does not yet use many rules that could help to detect misclassification. The syntax checker should be extended with these rules.

Classification is now based on a simple Euclidean distance measure in the Hotelling transformed space, taking this distance as an indication for the *amount* of distortion with respect to a certain prototype character, without considering the direction of the displacement of the sample with respect to the prototype character. This approach totally ignores the *kind* of distortion and the probability that a special kind of distortion can occur. For this reason the application of a Bayesian classifier [13] was considered, which should be able to deal with this in a more sophisticated way. With a Bayesian classifier it are not only the prototype characters, but the distribution of all the characters of the training set in the Hotelling transformed space that determine how a sample character is classified. However, in order to obtain a proper estimation of the underlying probability distribution one needs a huge and representative training set, which first has to be collected.

Our system relies on only one recognition method. It is expected that a major improvement of the performance could be obtained, when a combination of several (complementary) recognition methods would be applied. This combination could be realised by the application of several character recognisers, working in *parallel*, using a statistically optimised voting mechanism to produce the overall response. Another way to exploit the combination of multiple approaches could be to construct a multi-stage system in which the recognisers work in a *serial* fashion.

Although the obtained results are encouraging, the mentioned suggestions for further improvement can still be studied in the future.

ACKNOWLEDGEMENT

This work was sponsored by the Industrial Vision Department of Philips Centre for Manufacturing Technology.

REFERENCES

[1] D.W. Tindall, "Deployment of automatic license plate recognition systems in multinational environments", European Conference on Security and Detection 1997, p. 28-30.

[2] S. Draghici, "A neural network based artificial vision system for license plate recognition", Int. J. on Neural Systems, Feb. 1997, Vol.8, Iss.1, p.113-126.

[3] T. Sato, H. Sato, S. Ozawa. M. Komatsu, A. Kobayashi and M. Sato, "Motion blur correction of license plate images on running vehicle", Trans. IEE Japan, June 1997, Vol.117-C, Iss.6, p.777-784.

[4] Y.T. Cui and Q. Huang, "Character extraction of license plates from video", Proc. IEEE Computer Society Conf. on Computer Vision and Pattern Recognition 17-19 June 1997, p.502-507.

[5] H. Fujiyoshi, T. Umezaki, T. Imamura and T. Kanade, "Area extraction of the license plate using artificial neural networks", Trans. Of the Institute of Electronics, Information and Communication Engineers D-II, June 1997, Vol. J80D-II, Iss.6. p.1627-1634.

[6] T. Hori, "Traffic camera system development", Proc. SPIE Feb.1997, Vol.3028, p.81-90.

[7] T. Agui, H.J. Choi and M. Nakajima, "Method of extracting car number plates by image processing", Systems and Computers in Japan, 1988, Vol.19, No.3, p.46-52.

[8] M.H. ter Brugge, J.H. Stevens, J.A.G. Nijhuis, L. Spaanenburg, "License plate recognition using DTCNNs", Proc. CNNA 1998, p.212-217.

[9] R.J. De la Haye, "A license plate recognition system", Eindhoven Univ. of Technology, 1998, Master Thesis.

[10] Philips Centre for Manufacturing Technology, "Kenteken verwerking (KEVER)" (Processing of license plates, in Dutch), design documentation, 1993.

[11] R.C. Gonzalez, and R.E. Woods, "Digital image processing", first edition, Addisson-Wesley Publishing Company, 1993, ISBN 0-201-50803-6.

[12] B. Fukunaga, "Introduction to statistical pattern recognition", Academic Press, London, 1990.

[13] P.D. Wasserman, "Advanced methods in neural computing", Van Nostrand Reinhold, 1993, ISBN 0-442-00461-3.