OCR-based Chassis-Number Recognition using Artificial Neural Networks

Parul Shah, Sunil Karamchandani, Taskeen Nadkar, Nikita Gulechha, Kaushik Koli, Ketan Lad
Department of Electrical Engineering, IIT Bombay, Mumbai, India - 400076
Email: {parul,sunilk,taskeenn}@ee.iitb.ac.in

***Abstract*— The automatic detection and recognition of car number plates has become an important application of artificial vision systems. Since the license plates can be replaced, stolen or simply tampered with, they are not the ultimate answer for vehicle identification. The objective is to develop a system whereby vehicle identification number (VIN) or vehicle chassis number is digitally photographed, and then identified electronically by segmenting the characters from the embossed VIN. In this paper we present a novel algorithm for vehicle chassis number identification based on optical character recognition (OCR) using artificial neural network. The algorithm is tested on over thousand vehicle images of different ambient illumination. While capturing these images, the VIN was kept in-focus, while the angle of view and the distance from the vehicle varied according to the experimental setup. These images were subjected to pre-processing which comprises of some standard image processing algorithms. The resultant images were then fed to the proposed OCR system. The OCR system is a three-layer artificial neural network (ANN) with topology 504-600-10. The major achievement of this work is the rate of correct identification, which is 95.49% with zero false identification.**

Index Terms: Vehicle Identification Number (VIN); Optical Character Recognition (OCR); Artificial Neural Network (ANN).

I. Introduction

The Vehicle Identification Number (VIN) or chassis number, as it is sometimes known, is a unique identification number for every car manufactured. A registration number can be changed, but the VIN is constant throughout the lifetime of the car. Moreover, the registration number is not unique to a car since it can be carried forward to another car. The chassis number can be found most likely close to the engine, on the body frame, depending on the the make or brand of the car. The wall behind the motor (firewall) separates the engine bay and the vehicles dashboard in the cabin. The chassis number, in almost all cases, is in the middle, at the top of the firewall and either stamped in the same color as the paintwork or on a silver "credit card" sized plate. The license plates can be deliberately altered in fraud situations or replaced (e.g., with a stolen plate), which is not possible with the VIN.

Optical Character Recognition (OCR) is used for a wide range of character recognition applications which includes anything from transformation of anything humanly readable to machine manipulatable representation. Optical character recognition (OCR) is the translation of optically scanned bitmaps of printed or written text characters into character codes such as ASCII. OCR aims at enabling computers to recognize optical symbols without human intervention [1]-[3]. This is accomplished by searching a match between the features extracted from symbol's image and library of image models. Computer systems equipped with such an OCR system improve the speed of input operation, decrease some possible human errors and enable compact storage, fast retrieval and other file manipulations [4]. A typical OCR system contains three logical components: a camera, OCR software, and an output interface. The final output must be in the form of a string of characters. The whole reason for the existence of OCR is to save time and money in converting characters punched on metal castings to electronic format. The camera optically captures text images to be recognized. Text images are processed with OCR algorithm which performs the following operations: filtering, segmentation of characters, thinning of the segmented characters and finally character recognition using Artificial Intelligence (AI). The output interface is responsible for communication of OCR system results to the outside world.

II. Related Work

The technique of identifying VIN is new and more challenging than license plate recognition. This is because license plates have a fixed format (at least) region-wise, while VIN number differs according to the make and the model of the vehicle. Further, the license plates have a standard and a visible look with foreground and background with stark contrast, making character segmentation comparatively easy. However, the VIN is embossed on the car body which makes it difficult to decipher.

A number of techniques to segment each character after localizing the license plate in the image have been developed, such as feature vector extraction and mathematical morphology [5], and Markov random fields (MRFs) [6]. The work in [5] indicates that the method could be used for character segmentation in plates with indistinguishable characters during off-line operation, but since the algorithm is computationally complex it cannot be proposed for real-time license plate recognition. The method in [5] was developed for license plate segmentation in video sequences. However, the segmentation results were far from suitable for automatic character recognition.

For the recognition of segmented characters, numerous algorithms exploited mainly in optical character recognition applications utilized hidden Markov models (HMMs) [7][8], Hausdorff distance [9], SVM-based character recognizer [10], and template matching [11][12]. The method in [8] reveals the necessity for good character analysis when implementing HMM, which poses a restriction in the effective distance of the recognition system. The main problem of Hausdorff distance is that although its recognition rate is very similar to that obtained with neural-network classifiers, it is slower. Therefore, it is good as a complementary method if real-time requirements are not very strict. It can be used if there is extra time to improve results from another classifier [9]. The authors of [10] have designed a system implementing four Support Vector Machines (SVMs) and report an impressive average character recognition rate of 97.2%. The architecture, however, is strictly designed for Korean plates, rendering it inapplicable to license plates of other countries.

In addition, various neural-network architectures [13][14] have been developed for license plate identification but none of them have been used or tested on the VIN.

III. Proposed Technique

Image Acquisition is the first process of Optical Character Recognition (OCR). The numeric code punched on the machine casting is captured in an image file using a CCD (Charged Coupled Device) camera. Initially a histogram of about 100 images from the set of about thousand chassis numbers were plotted. The histograms showed that data concentration is much more in upper few gray levels of the histogram. As the images differ from each other in terms of size of VIN and the degree of chalking of the punched codes, it is not possible to decide on a static global threshold because there could be a certain amount of loss of information in the same. Thus, preprocessing is required as a prerequisite to the OCR algorithm.

A. Preprocessing

1) Edge Detection: The first module of the preprocessing algorithm consists of horizontal and vertical edge detection. The edge detection is basically required to identify the characters on the chassis. This is a very important step as the VIN differs from one make of the vehicle to the other. A Sobel mask is used because Sobel kernels are designed to respond maximally to edges running vertically and horizontally relative to the pixel grid, one kernel for each of the two perpendicular orientations. The mask is applied separately to the input image, to produce separate measurements of the gradient component in each orientation (G_x and G_y) as shown in Figures 1(b), 1(c), 2(b) and 2(c). These are then combined together to find the absolute magnitude of the gradient at each point as shown in Figure 1(d) and 2(d). The gradient magnitude is given by:

$$\mod G = \sqrt{G_x^2 + G_y^2} \tag{1}$$

The process of edge detection is followed by segmentation, normalization and thinning.

2) Segmentation: In this module we first horizontally segment out the chassis number from the image background. To do that we scan the image row-wise once from top to bottom and then from bottom to top, and discard all the rows till we have a row with some white pixels as shown in Figure 1(e) and 2(e). The pixels which represent the chassis number characters have intensity value of 255 (white) while the background has a value of 0 (black). Similar process is undertaken for vertical segmentation for which scanning is done column-wise. The column containing all black pixels is part of gaps between the rectangular boxes of the chassis code. Identifying these regions, we separate each character of the chassis number in such a way that every individual character is segmented in rectangular boxes as shown in Figure 1(f) and 2(f).

3) Normalization: The segmented characters of the chassis numbers are in rectangular boxes which have different sizes, so we normalize it to a fixed size of pixels by scaling. The rectangular boxes are then removed by discarding a few number of rows and columns along the boundary of the boxes on the normalized images. The decision of how many rows and columns should be discarded to get rid of box without distorting the character, was based on study of hundreds of such images. Thus, after the box removal process we have only the individual characters which have to be thinned in the next stage as shown in Figure 1(g) and 2(g).

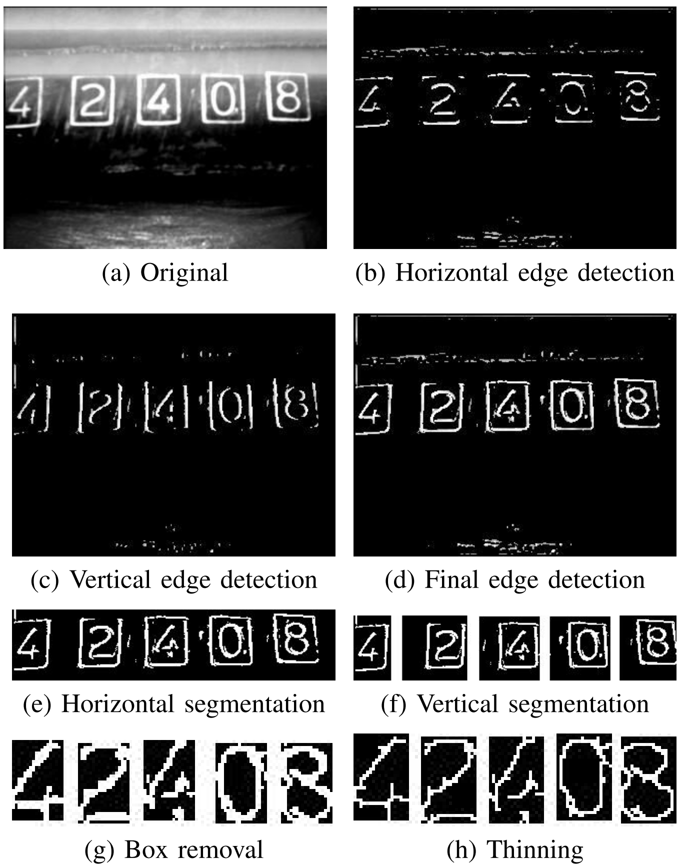
4) Thinning: Morphological thinning is performed on individual image for reducing characters to single pixel thickness while preserving the full length of those lines (i.e. pixels at the extreme ends of the lines should not be affected) as shown in Figure 1(h) and 2(h). These characters are now given to a feed forward back propagation artificial neural network for training/identification.

B. OCR using ANN

The use of artificial neural network (ANN) in OCR applications can dramatically simplify the code and improve quality of recognition while achieving good performance. Another benefit of using neural network in OCR is extensibility of the system i.e. ability to recognize more character sets than initially defined. In the character recognition algorithm using neural networks, the weights of the neural network were adjusted by training it using back propagation algorithm. The size of each character is 28-by-18 pixels which are arranged column wise to give $504 * 1$ arrays as input. In order to train the neural network, we have created different sets each containing digits from 0 to 9. This is called Block training. The complete network was implemented as a library, which was statically tied to the project. This helped to isolate the neural network code from the rest of the preprocessing and segmentation code. It also helped to reduce the memory required for the program.

IV. Experimental Results and Analysis

The trained ANN was used to identify the characters of the VIN. The ANN decides the final output on the basis of a threshold. We have used three values of threshold viz. 0.75, 0.65 and 0.55. Only if the maximum output value of the network exceeds the threshold value, the character corresponding to that node is considered recognised. Table I shows the output of the neural network for each of the five characters of the VIN shown in Figure 1(a) and 2(a), with a threshold value of 0.75. The output1 represents the output of ANN when the first character was given as the input, output2 represents the second character of the chassis and so on. It is found that with a threshold of 0.75 we achieve a wrong identification rate (*WIR*) of zero and a correct identification rate (*CIR*) of 95.49%. Table II shows the performance of the OCR for different values of the threshold, in terms of *CIR*, *WIR* and nil identification rate (*NIR*). The major achievement of the neural network is significantly high value of *CIR* with zero *WIR*. Zero *WIR* is very important, because an incorrectly identified VIN can result into false alarms and can also be disastrous if a crime is involved.

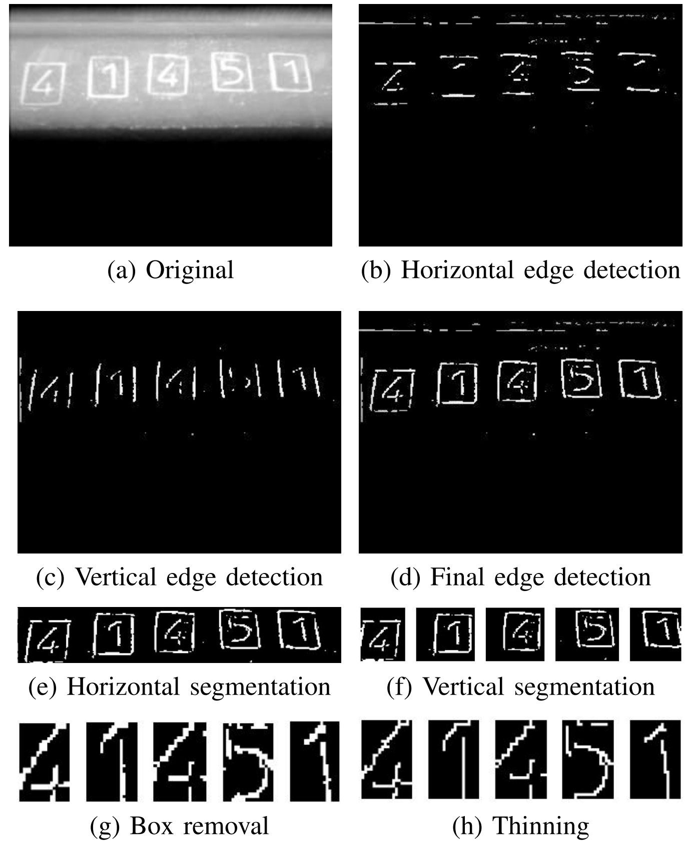


(a) Original (b) Horizontal edge detection (c) Vertical edge detection (d) Final edge detection (e) Horizontal segmentation (f) Vertical segmentation (g) Box removal (h) Thinning

Fig. 1. Example of Proposed Pre-processing for OCR

V. Conclusion

The proposed chassis number-recognition algorithm based on OCR using artificial neural networks is well suited for intelligent vehicular systems. The major achievements of the proposed method is that it gives considerably high value for correct identification rate *(CIR)* along with zero wrong identification rate *(WIR)*. This is crucial because even a single instance of wrong identification of the VIN may lead to incorrect interpretations, which can be catastrophic in certain applications.



(a) Original (b) Horizontal edge detection (c) Vertical edge detection (d) Final edge detection (e) Horizontal segmentation (f) Vertical segmentation (g) Box removal (h) Thinning

Fig. 2. Example of Proposed Pre-processing for OCR

TABLE I

Outputs of Neural Network for VIN 42408

Node	output1	output2	output3	output4	output5
$node0$	-0.2493	-0.1243	-0.1077	-0.2458	-0.2293
$node1$	0.0530	0.8408	0.0764	-0.0478	-0.1295
$node2$	-0.0926	0.0202	0.1024	0.0260	0.0226
$node3$	0.8036	0.0064	1.1335	-0.0630	0.1589
$node4$	0.0792	-0.0249	0.0329	-0.0981	-0.0275
$node5$	-0.1157	0.0873	-0.0487	0.0309	0.1936
$node6$	0.1704	0.4389	0.0468	-0.0404	0.1618
$node7$	0.0691	0.1371	0.4311	-0.0303	1.0477
$node8$	-0.0136	-0.1917	-0.0889	0.0382	-0.1045
$node9$	-0.0775	-0.3680	0.2563	0.9857	-0.2769

Acknowledgements

This work was supported by Microsoft Research India under the MSRI PhD Fellowship Award 2008.

References

[1] Haralick R., ”Performance Characterization in Image Analysis: Thinning, A Case in Point”, Pattern Recognition Letters, vol. 13, 1992, pp. 5-12.

[2] Wu V., Manmatha R. and Riseman E. M., ”TextFinder: An Automatic System to Detect and Recognize Text in Images”, IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 21 No. 11, 1999, pp. 1224-1229.

[3] Li, H., Doermann, D. and Kia, O., ”Automatic Text Detection and Tracking in Digital Video’, IEEE Transactions on Image Processing, Vol. 9. No. 1, 2000, pp. 147-157.

TABLE II

PERFORMANCE OF THE PROPOSED OCR

Threshold	CIR	WIR	NIR
0.75	95.49	0	4.51
0.65	89.19	8.11	2.7
0.55	78.3	21.7	0

[4] Mani, N. and Srinivasan, B., "Application of Artificial Neural Network Model for Optical Character Recognition", Proc. IEEE Int. Conf. on Systems, Man and Cybernetics:'Computational Cybernetics and Simulation', 1997, pp. 2517-2520.

[5] 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.

[6] 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.

[7] T. D. Duan, T. L. Hong 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.

[8] D. Llorens, A. Marzal, V. Palazon, and J. M. Vilar, "Car license plates extraction and recognition based on connected components analysis and HMM decoding," in Lecture Notes on Computer Science, vol. 3522, J. S. Marques et al., Eds. New York: Springer-Verlag, 2005, pp. 571-578.

[9] F. Martn, M. Garca, and L. Alba, "New methods for automatic reading of VLP's (Vehicle License Plates)," in Proc. IASTED Int. Conf. SPPRA, Jun. 2002.

[10] 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, Neural Netw. Signal Process., 2000, vol. 2, pp. 614-623.

[11] 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.

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

[13] 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 Neural Netw. and Appl., 1998, pp. 212-217.

[14] 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.