



Journal of Applied Sciences

ISSN 1812-5654

science
alert

ANSI*net*
an open access publisher
<http://ansinet.com>

OCR Based Pixel Fusion

Rami Al-Hmouz
Faculty of Engineering, King Abdulaziz University, P.O. Box 80204,
Jeddah 21589, Saudi Arabia

Abstract: Character recognition is the process that allows the automatic identification of character images, which is generally referred as Optical Character Recognition (OCR). The characters are either handwritten or typed. This study proposed a novel OCR approach based on the likelihood functions of pixels, which were obtained by averaging a trained set of character images. A Bayesian fusion process for all pixel probabilities decides the recognition of characters. Further tests using Support Vector Machine (SVM) classifier were carried out on characters with the same shape. This method was used to test noisy images and achieved an accuracy of 97.95%, thus, outperforming other OCR methods.

Key words: Optical character recognition, optical character recognition, fusion, Bayes theorem, pixel likelihood function

INTRODUCTION

The OCR system was developed over the past 50 years and is now commercially available as software packages. OCR has many applications, including data entry, signature identification and License Plate Recognition (LPR). OCR performance primarily depends on the quality of the input image, most of the existing OCR systems work with a very constrained character image and are still unable to provide a reliable accuracy under various conditions, thus, no comparisons can be made with human reading capabilities (Cheung *et al.*, 2001). The outputs of OCR comprise character scores of how likely the recognized character appears (Fig. 1).

LPR system is a well-explored problem. The performance of these systems is heavily dependent on environmental setup such as quality of cameras, special lighting and other forms of environmental control. A typical LPR system structure is composed of four modules (Al-Hmouz and Challa, 2010):

- Image acquisition-acquire images from video sequences and feed them into the system
- License plate extraction-locate and extract the plate from the acquired image
- Character segmentation-separate characters from the extracted license plate
- Character recognition-identify segmented characters

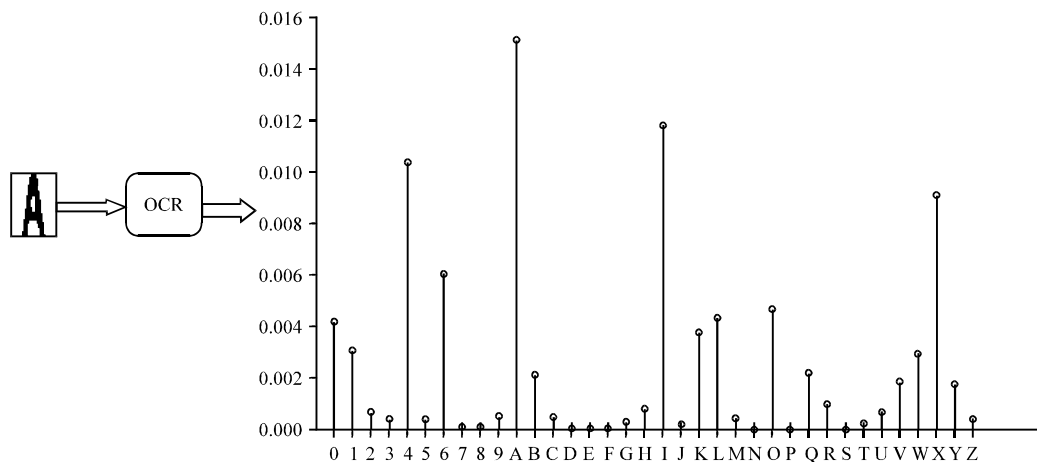


Fig. 1: Outputs of OCR

Neural Networks (NN) provide satisfactory answers to the OCR in LPR, but they have drawbacks in speed, complexity and training requirements. Template matching is a minimization of squared error approach that does not provide a probabilistic answer and can be computationally expensive. Probability is the theoretically sound framework in which an uncertainty problem is treated. In the probabilistic framework, a likelihood function is built to model the characteristics of the problem. It is then either maximized or inverted to provide the most likely solution (Csato *et al.*, 2003). In this study, a probabilistic solution to the problem is presented. The likelihood functions of pixels are computed based on statistical features of trained data set and then character is recognized using the famous Bayesian inference. OCR is discussed in the context of LPR. The likelihood functions of pixels are formed based on statistical features of data; characters are then recognized using Bayes' theorem.

SURVEY OF OCR APPROACHES

Various OCR methods have been explored, including template matching, Neural Network (NN), Hausdorff distance, Support Vector Machine (SVM), Hidden Markov Models (HMMs) and the probabilistic model. In the template matching and Hausdorff distance methods, the character image is compared to templates (character images) and the image that has the minimum distance or the highest match is considered to have the same features of the character image (Fig. 2).

The distance between image I and image J can be measured by Euclidean distance, Hausdorff distance (Shuang-Tong and Wen-Ju, 2005; Juntanasub and Sureerattanan, 2005), Chamfer matching (Barrow *et al.*, 1977) and cross correlation. The correlation measure is achieved through Normalized Cross Correlation (NCC) as shown in Eq. 1. Here, the image J that maximizes NCC is considered the recognized character:

$$NCC = \frac{\sum_{i=x}^{x+M-1} \sum_{j=y}^{y+N-1} I(i,j)J(i-x,j-y)}{\sqrt{\sum_{i=x}^{x+M-1} \sum_{j=y}^{y+N-1} |I(i,j)|^2 \sum_{i=x}^{x+M-1} \sum_{j=y}^{y+N-1} |J(i,j)|^2}} \quad (1)$$

The NCC has values between -1 (no correlation) and 1 (high correlation). Intuitively, when images I and J are the same characters, the maximum value of NCC occurs when I and J are exactly on top of each other. Therefore, the values of x and y that satisfy this condition are monitored when compared with all characters (J). Equation 1 can be simplified in vectors, as shown in Eq. 2:

$$NCC = \frac{I^T J}{|I||J|} \quad (2)$$

Template matching is a simple and easy method and has been used in LPR (Comelli *et al.*, 1995). Ko and Kim (2003) used hierarchical template matching by extracting the concentrated common features from the first candidates and then running template matching again for

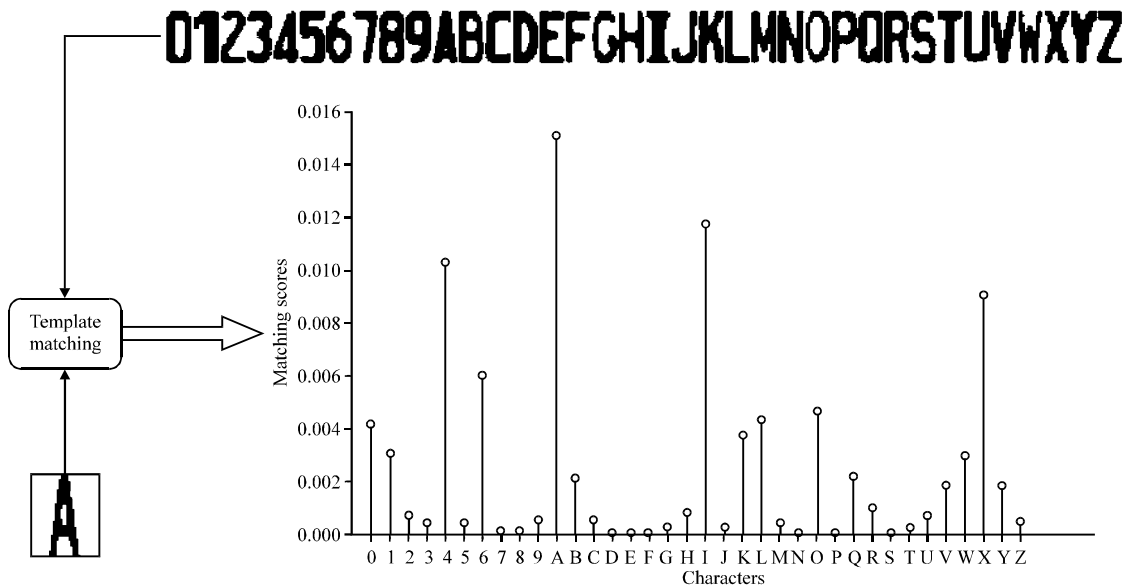


Fig. 2: Template matching

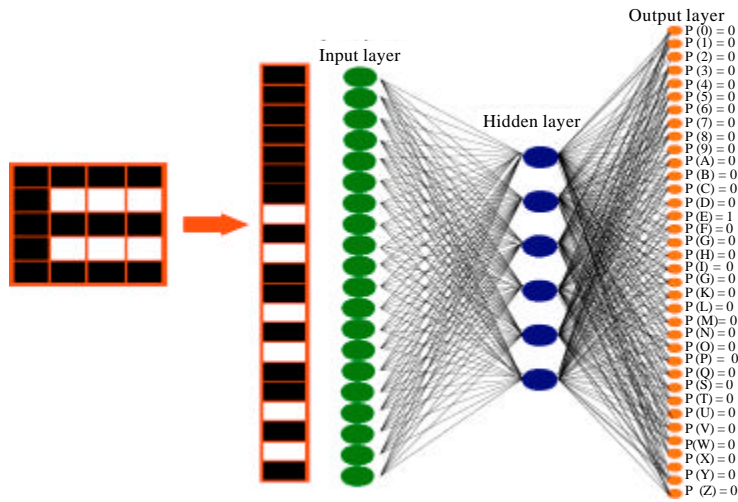


Fig. 3: Neural network (training phase)

the second stage. Template matching and distance methods are sensitive to noise and cannot handle the rotated image. Feature based character recognition is also investigated in the context of LPR. Two features from the character are extracted (Wang and Lee, 2003): Contour crossing and peripheral background. Contour crossing counts the number of strokes after it is divided into sub-images and the peripheral background calculate the outer area from the character to the image boundary. The feature vector of character mean and standard deviation was fed to a discrimination function that evaluates the differences between the input character and the rest of the characters. Support Vector Machine (SVM) based recognition is investigated by Kim *et al.* (2000). Four SVMs were used to classify characters and numerals, results were then fed into 10 SVMs for numerals (0-9) and into 26 SVMs for characters (A-Z). A recognition rate of 97% was reported. SVM with SIFT descriptors was also investigated by Yang *et al.* (2011), however SVM classifier is still very sensitive to noise. The HMM was also used for character recognition (Duan *et al.*, 2005); however, the result was still not convincing compared with other methods.

The classification of segmented characters was examined using NN (Wei *et al.*, 2001). In general, the NN has the ability to learn, as desired outputs are chosen based on sample inputs. The NN also has the ability to generalize, in which reasonable outputs are produced for inputs that have not been tested. The character recognition application is exploited from supervised learning. Characters are fed into the input layer of the NN and the desired outputs are obtained. The characteristic features (pixel values) are propagated through the neurons. The training process performs an iterative

minimization of errors between the input and the output over a training set. Figure 3 describes the NN's structure, where the image is converted to one vector and fed into the input layer and the corresponding neuron in the output is set to 1, while the rest are set to 0. The main drawbacks of the training process are over fitting and its time-consuming nature. The NN is trained on a training set, while the error is recorded on a validation set. In order to avoid over fitting, the training stops when the error on the validation set is at its minimum. The performance is estimated on a test set. Cross-validation can be conducted by placing two sets for validation and then averaging the error for both sets. The previous method can be repeated several times for several NN parameters, such as the number of nodes in the hidden layer and the number of iterations needed to stop the training process. Each time the error in the validation set is monitored, the weights of the training set that produced the lowest error are chosen to be the weights of the trained NN. However, the NN needs to be trained on a large number of noisy characters in order to obtain reasonable results.

The Probabilistic Neural Network (PNN) was developed by Specht (1990) and provides a solution to classification problems using Bayesian classifiers and the Parzen Estimators. PNNs are very effective for pattern recognition. However, the LPR character recognition problem is a simple OCR problem and can be addressed by a full probabilistic approach. A probabilistic approach using Bernoulli's trial was presented by Aboura and Al-Hmouz (2007).

All character recognition classifiers successfully operated in controlled environments in which the illuminations and weather conditions were controlled and

the locations of the cameras were pre-determined. When the character reached the recognition phase, it presented a very clear image that could be easily recognized. However, the real scenario was on the opposite side, wherein the character could be blurry or in low resolution and the effects of illumination existed, as well as noise that accumulated from the previous stages.

PIXEL FUSION APPROACH

In LPR systems, character images are converted into black and white. All information regarding character images is found in pixels with logic of 0 or 1. However, noise produced from the conversion process and from previous LPR stages may affect in-pixel values, where 0 may become 1 and vice versa. The inversion of pixel values may lead to incorrect recognition of characters. In order to reduce the effect of noise on characters and to find the standard character images, a training set of characters is averaged and their likelihood functions are constructed from statistical features of the averaged characters (Fig. 4).

The pixel's likelihood functions have been constructed for all characters by averaging the same set of characters. The value of 1 or 0 in the character images implies that all trained characters for that particular location are 1 and 0 respectively. However, when the value is between 0 and 1 (grey), it implies that noise are added in some characters in that location. If the value was closer to 0, it signifies that the probability for the pixel to be 0 for that particular location is higher than 1 and vice versa if the value is closer to 1.

Each pixel in the averaged character produces a distribution with mean m and standard deviation σ . The distribution of most pixels follows the normal distribution that have m and σ . The distributions of pixels represent pixel likelihood functions with different m_s and σ_s based in their locations in the averaged image. Let s_{ij} be a pixel in image S , then the probability (likelihood function) of pixel s_{ij} at location (i, j) given the character C is given in Eq. 3:

$$p\left(\frac{s_{ij}}{C}\right) = \frac{1}{\sqrt{2\pi}\sigma_{ij}} \exp\left(-\frac{(s_{ij} - m_{ij})^2}{2\sigma_{ij}^2}\right) \quad (3)$$

where, s_{ij} is the pixel value (0 or 1), m_{ij} is the mean of pixel at location (i,j) for character C and σ_{ij} is the standard deviation of pixel (i,j) for character C . Making the assumption of conditional independence of the pixel values for a given character image, the probability $P(C/S)$ for a character image C , given a character image S is shown in Eq. 4:

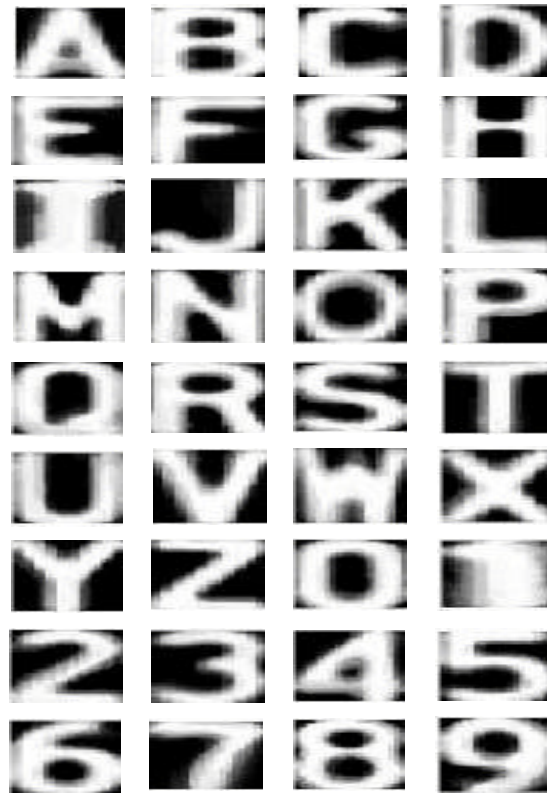


Fig. 4: The averaged character images

$$P\left(\frac{C}{S}\right) = \frac{P\left(\frac{S}{C}\right)P(C)}{\sum_{C=A}^{C=9} P\left(\frac{S}{C}\right)P(C)} \quad (4)$$

where:

$$P\left(\frac{S}{C}\right) = \prod_{i,j} \frac{P\left(\frac{s_{ij}}{C}\right)}{P\left(\frac{s_{ij}}{C}\right) + P\left(\frac{\bar{s}_{ij}}{C}\right)}$$

$P(S/C)$ was computed using a Bayesian fusion of all pixel probabilities of image S . $P(C)$ is the prior probability for the character C ; for example in LPR systems, some characters in the plate could have certain locations in which the posterior $P(C/S)$ increased or decreased for such information. However in our test, all characters are equally likely ($P(C) = 1/36$):

$$\sum_{C=A}^{C=9} P\left(\frac{S}{C}\right)P(C)$$

is a normalization term to make the posterior as a valid probability. The Maximum A Posteriori probability (MAP) of C was used to estimate the character image S . C that

maximizes $P(C/S)$ was considered the recognized character as shown in Eq. 5:

$$\max_{C \in \{A, B, \dots, 9\}} \left\{ P\left(\frac{S}{C}\right) \right\} \quad (5)$$

RESULTS

The likelihood functions have been constructed from a noisy training set and the method is tested on the test set. There were 1,500 character images in the training set and 3,030 in the test set that have been extracted from noisy license plates under various illuminations. Table 1 shows the accuracy of the recognition for each character using the proposed method.

The all over accuracy of the system was 95.05%. The same evaluation set was tested using feed forward NN methods and the probabilistic method in Al-Hmouz and Challa (2010), the achieved accuracies were 93.60 and 94.16%, respectively. Our method achieves better performance in the case of well-extracted and cleaned characters. As in most methods, character combinations such as 1/I, 2/Z, 5/S and O/0/D/Q can be mistakenly recognized because they have almost the same shape in the presence of noise. Two solutions were presented to eliminate such confusion. First, the same logic was applied in some parts of the image that distinguished the confusing characters from each other. A similar approach was proposed (Shuang-Tong and Wen-Ju, 2005). The second solution, an SVM classifier is used as it achieves good results in the case of two classes 1/I, 2/Z, 5/S, O/0, O/D, O/Q, 0/D, 0/Q and D/Q.

Eliminating the confusion: The same probabilistic approach described above was used in some parts of the image when confusion occurred. Figure 5, shows how 2 and Z was distinguished by focusing the pixel fusion algorithm in the areas of the boxes. The likelihood of the pixels in those areas had already been computed and was applied to the image that was initially recognized as either 2 or Z.

Similarly, Fig. 6 shows how the remaining confusions were resolved. This refinement was the equivalent of moving a magnifying glass, in the form of a pixel fusion algorithm, to specific parts of the image. Although, the likelihood function was the same, the scoring changes and the choice were between two classes rather than 36. A reliable rate with this refinement reached 96.37%. However, there were still many confused characters.

Vapnik (1995) proposed SVM, which is a technique used for classification and regression analysis. In classification, two groups of data sets are trained to be

Table 1: Recognition accuracies of characters

Character	Probability (%)	Character	Probability (%)
0	85.22	1	90.08
2	81.94	3	97.70
4	97.37	5	93.75
6	95.53	7	94.29
8	81.33	9	95.73
A	98.76	B	94.03
C	98.99	D	83.70
E	94.70	F	98.85
G	93.10	H	100.00
I	96.43	J	97.30
K	99.23	L	100.00
M	100.00	N	96.35
O	95.59	P	100.00
Q	81.13	R	96.09
S	94.44	T	97.47
U	93.26	V	96.15
W	98.18	X	98.25
Y	96.15	Z	96.29



Fig. 5: Pixel fusion applied to eliminate confusions for characters 2 and Z

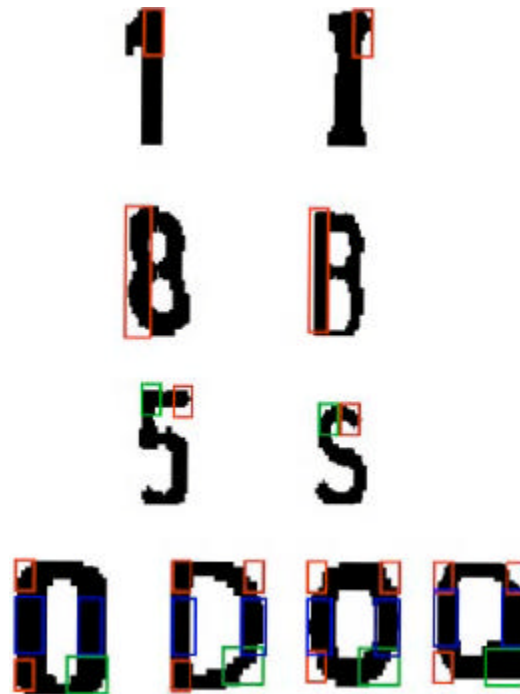


Fig. 6: Eliminating confusion of characters

Table 2: Reported OCR accuracies

Accuracy (%)	Reference
94.7	Kim <i>et al.</i> (2000)
95.2	Duan <i>et al.</i> (2005)
98.6	Wang and Lee (2003)
91.0	Comelli <i>et al.</i> (1995)
90.0	Cheung <i>et al.</i> (2001)
92.0	Juntanasub and Sureerattanan (2005)
95.0	Wei <i>et al.</i> (2001)
98.0	Shuang-Tong and Wen-Ju (2005)

labeled into one of two categories. The parameters that are produced from the training process are then used to classify a new set of data into one of the two categories. After the probability scores were computed from the pixel fusion process, the scores were represented in a vector of 36 entries. The SVM classifier was performed when the highest scores were presented for 0, 1, 2, 5, 8, B, D, I, O, Q, S and Z. Intuitively in character recognition, if the character is recognized as 2, the next highest probability score should be for the character Z and SVM classifier would confirm whether it was 2 or Z.

Applying pixel fusion and SVM in the case of confused characters, a success rate of 97.95% was achieved. Generally, SVM classifier can be used not only for the confused characters but also when the scores of the two consecutive top posterior probabilities $P(S/C)$ of some characters are close.

Table 2 shows OCR accuracies which were reported in the literature. Although, the reported OCR approaches were tested on a different data set and number of character images that have been used to record their accuracies is quite low compared to our test data set. The pixel fusion approach is competing with existing OCR approaches.

CONCLUSIONS

We introduced a very simple solution for the OCR problem. The OCR approach is presented based on pixel information and the likelihood functions of characters are constructed by averaging noisy character samples. The decision on the recognition is based on the MAP of Bayes' theorem. This method outperforms other numerical methods, such as NNs, in both accuracy and complexity. Two solutions for the confused characters are presented when the recognized characters are 1, I, 2, Z, 5, S, O, 0, D and Q. Firstly; pixel fusion is applied on certain parts of the confused characters in which the recognition of the character at the end might change according to the likelihood functions of the focusing parts. Secondly, SVM achieved good output performance in the case of two classes.

ACKNOWLEDGMENT

This article was funded by the Deanship of Scientific Research (DSR), King Abdulaziz University, Jeddah. The authors, therefore, acknowledge with thanks the technical and financial support of DSR.

REFERENCES

- Aboura, K. and R. Al-Hmouz, 2007. Probabilistic license plate optical character recognition. Technical Report, Department of Computer Systems Engineering, University of Technology Sydney, Australia.
- Al-Hmouz, R. and S. Challa, 2010. License plate localization based on a probabilistic model. *Machine Vision Appl.*, 21: 319-330.
- Barrow, H., J.M. Tenenbaum, R.C. Boles and H.C. Wolf, 1977. Parametric correspondence and chamfer matching: Two new techniques for image matching. *Proceedings of the 5th International Joint Conference on Artificial Intelligence*, August 22-25, 1977, Morgan Kaufmann Publishers Inc., San Francisco, CA., USA., pp: 659-663.
- Cheung, A., M. Bennamoun and N.W. Bergmann, 2001. An Arabic optical character recognition system using recognition-based segmentation. *Pattern Recogn.*, 34: 215-233.
- Comelli, P., P. Ferragina, M.N. Granieri and F. Stabile, 1995. Optical recognition of motor vehicle license plates. *IEEE Trans. Veh. Technol.*, 44: 790-799.
- Csato, L., M. Opper and O. Winther, 2003. Tractable inference for probabilistic data models. *Complexity*, 8: 64-68.
- Duan, T.D., T.L.H. Du, T.V. Phuoc and N.V. Hoang, 2005. Building an automatic vehicle license-plate recognition system. *Proceedings of the International Conference on Computer Science Research*, February 21-24, 2005, Can Tho, Vietnam, pp: 59-63.
- Juntanasub, R. and N. Sureerattanan, 2005. Car license plate recognition through hausdorff distance technique. *Proceedings of the 17th IEEE International Conference on Artificial Intelligence*, November 14-16, 2005, Hong Kong, pp: 647-651
- Kim, K.K., K.I. Kim, J.B. Kim and H.J. Kim, 2000. Learning-based approach for license plate recognition. *Proceedings of the IEEE Signal Processing Society Workshop on Neural Networks for Signal Processing X*, Vol. 2, December 11-13, 2000, Sydney, Australia, pp: 614-623.
- Ko, M.A. and Y.M. Kim, 2003. License plate surveillance system using weighted template matching. *Proceedings of 32nd Applied Imagery Pattern Recognition Workshop*, October 15-17, 2003, Washington, DC., USA., pp: 269-274.

- Shuang-Tong, T. and L. Wen-Ju, 2005. Number and letter character recognition of vehicle license plate based on edge hausdorff distance. Proceedings of the 6th International Conference on Parallel and Distributed Computing, Applications and Technologies, December 5-8, 2005, Dalian, China, pp: 850-852.
- Specht, D.F., 1990. Probabilistic neural networks. Neural Network, 3: 109-118.
- Vapnik, V.N., 1995. The Nature of Statistical Learning Theory. 1st Edn., Springer-Verlag, New York, USA.
- Wang, S.Z. and H.J. Lee, 2003. Detection and recognition of license plate characters with different appearances. Proceedings of the IEEE Intelligent Transportation Systems, Vol. 2, October 12-15, 2003, China, pp: 979-984.
- Wei, D.W., Y. Li, M. Wang and Z. Huang, 2001. Research on number plate recognition based on neural networks. Proceedings of the IEEE Signal Processing Society Workshop, Neural Networks for Signal Processing XI, September 12-12, 2001, North Falmouth, MA., USA., pp: 529-538.
- Yang, M.N., X.J. Li and X.H. Zhang, 2011. Robust description method of SIFT for features of license plate characters. Inform. Technol. J., 10: 2189-2195.