A Naive Bayes Classifier for Character Recognition

Conference Paper · June 2016

1 author:

Khalid Aboura
The American University of Armenia

A Naive Bayes Classifier for Character Recognition

Khalid Aboura
College of Business Administration
University of Dammam, Saudi Arabia
Email: kaboura@uod.edu.sa

***Abstract*—We describe work done some years ago that resulted in an efficient Naive Bayes classifier for character recognition. The original idea was to develop a probabilistic solution for a well known recognition problem where implemented solutions consisted mostly of Neural Network variants and distance based algorithms known as Template Matching. After much effort trying to develop a parametric model that would lead to reliable recognition, a breakthrough occurred when the realization was made that one ought to use all the pixels information contained in the image. The thought was triggered by a recall of the definition of the likelihood function in its most fundamental form. However, to use all pixels information in a parametric model would lead to a very complex approach. Unless, and the deduction was automatic, an assumption of independence was made. For the author who was not versed in all machine learning approaches, a rediscovery of the Naive Bayes approach was made. The new method was simple, easy to implement and produced reliable results. We review the recognition problem and introduce the Naive Bayes approach.**

I. Introduction

The problem considered was the recognition of characters in a license plate image, referred to as an Optical Character Recognition (OCR) problem. License plate recognition (LPR), or automatic number plate recognition (ANPR) is the use of video captured images for automatic identification of a vehicle through its license plate. The applications are numerous and LPR is a well known problem. The image analysis in LPR consists of three steps; (i) the localization of the license plate in the image, (ii) the extraction of the characters from the localized license plate region, and (iii) the recognition of those characters. Plate localization is an important step in LPR. It aims to locate the license plate of the vehicle in an image. Several methods have been proposed such as morphological operations [1], [2], edge detection [3]–[6], corner detection [7], sliding concentric windows [8], fuzzy logic [9], [10], [12], spatial variance method [11], Hough transform [13], [14], neural networks [15]–[19], mean shift algorithm [20], Fourier transform [21]–[24], Gabor transform [25], Wavelet transform [26], [27], the generalized symmetry transform [28], the genetic algorithm [29], [30], and the adaptive boosting (AdaBoost) algorithm [31]. The second step in LPR is to extract the characters from the localized license plate region. The most common method used is the projection method. This methods is applied after the thresholding of the image. The value of each image pixel is reduced from a multi-dimensional value in the case of colors, to the binary values 0 or 1, reducing the image to a black and white image. This operations reduces the dimension of the problem and brings out features of the image that can be used to solve the problem. The projection method counts the number of foreground pixels vertically and horizontally in the license plate area to separate and extract the characters [2], [14], [33], [34]. Other methods and variants include [16], [35], [36], thin window scanning [32], local vector quantization [25], scale shape analysis [37], Laplacian transform [38], Hough transform [39] and Markov Random fields [40].

II. Character Recognition

The problem we considered was the recognition of characters in LPR. The extraction of characters results in a number of selected regions in the image. These regions contain the characters of the license plate and are processed for recognition by an Optical Character Recognition (OCR) algorithm. This is a well known problem and the most common OCR approaches used in LPR are the correlation-based template matching [41], [44]–[46], and neural networks [9], [10], [19], [47]–[49]. Other methods are feature based [42], [43], [50], use pattern mapping [36] or are based on the Hausdorff distance [2], [51]. Binary classifiers [52] are also used as well as the Hidden Markov model (HMM) [14], [53].

A. Neural Networks

In LPR, Neural networks (NN) are used to localize the license plate in the image and to recognize the extracted characters of the plate. Neural networks mimic the ability to learn of biological neural systems [54]. For specific tasks such as pattern recognition, a NN is trained to recognize. This is done by feeding it a set of inputs to which the outputs are known. Training data is usually taken from historical records. The NN processes the inputs one by one and compares the resulting outputs against the desired outputs. Errors are calculated and weights which control the strength of network connections are adjusted at each iteration. The training is stopped once the NN reaches a satisfactory level of recognition. The set of final weights is used for processing new data. There are forms of unsupervised training in NN, but most applications in LPR train the network to infer the relationship between the image inputs, and the corresponding outputs. Neural networks require a considerable amount of training and do not guarantee convergence to a solution. NN cannot be considered theoretically sound. It is their success in solving some problems that are difficult to parameterize in a parsimonious way that seems to justify their use.

DOI: 11.ICCITR.2016.1.10

B. Template Matching

Template Matching is the other approach used in character recognition in LPR. It is distance based, comparing the image in question to image templates until part of it matches. There are many variants. In its simplest form, the image, in its binary form (Figure 1), is compared with same size parts of the template image (Figure 2) using a suitable metric. The metric can be the euclidian distance or a correlation measure between the pixels of the image and the template. For example, the cross-correlation, a statistical measure used by Horowitz [55] and Pratt [56] for image recognition, can be a metric for template matching. If $F_I(x, y)$ and $F_T(x, y),\ x = 1, \ldots, N_x,\ y = 1, \ldots, N_y$ are the pixel values of the digitized image and part of the template it is being compared to, respectively, then a normalized cross-correlation measure is

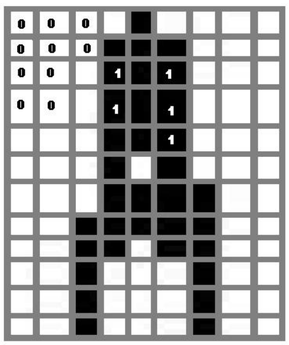

Fig. 1. Binary image with foreground pixel =1 and background pixel = 0

$$R(u, v) = \frac{\sum_x \sum_y F_I(x, y) F_T(x-u, y-v)}{[\sum_x \sum_y F_I^2(x, y)]^{1/2} [\sum_x \sum_y F_I^2(x-u, y-v)]^{1/2}}$$

The template matching approach is combined with other methods in character recognition. However, it remains a method based on the minimization of a distance between two images.

Fig. 2. LPR template for character recognition

III. Probability Modeling

Neural networks provide answers to the OCR problem, but they have drawbacks in speed, complexity and training requirements. Template matching is a minimization of distance approach that does not always provide an efficient 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. The drawback is that a probability approach can be intractable if there are too many parameters. Despite attempts at developing probability solutions, the only noticeable probability based research direction at the time was that of the probabilistic neural networks, for example [57]–[59]. The probabilistic neural network (PNN) was developed by Donald Specht [60], [61] and provides a solution to classification problems using Bayesian classifiers and the Parzen Estimators. It is a class of neural networks which combine statistical pattern recognition and feed-forward neural networks technology. It is characterized as having very fast training times and it produces outputs with Bayes posterior probabilities [62]. PNN are very effective for pattern recognition [63]. However, the LPR character recognition problem is a simple OCR problem and the feeling was that it can be addressed by a simple probabilistic approach. Despite converging to a Bayes optimal solution, PNNs are in effect neural networks.

A. Statistical Features

To solve the problem probabilistically is to treat features, or statistics, from the input image, using a probability model. The input character image to the OCR module is usually a binary image such as the one in Figure 1. Thresholding is the operation of turning the original image into a binary image. Many thresholding methods exist, such as the Otsu method [64] and the bi-threshold procedure called Hysteresis thresholding [65]. Thresholding is usually used in the first steps of LPR. Depending on the method, the operation brings out some features of the images while burrying others into the background. It can at times produce very thin extracted characters. This problem can be avoided by applying the Hysteresis thresholding, with proper parameters, on the original image, in the region that corresponds to the extracted characters. This provides for good input data to the OCR system. In [66], a new thresholding method is introduced that performs better in LPR problems.

1) Historical Data: The historical data of extracted characters is made of two sets; training data and validation data. These are images of license plate characters that have been extracted from images of vehicles, using the first two steps LPR. They are binary images that have been cleaned, cropped and normalized (Figure 3). The characters are visually inspected one by one, and classified manually in the 36 possible classes {A,B,C, ..., X,Y,Z,0,1, ...,8,9}. Each set of characters is then split into a training set and a validation set.

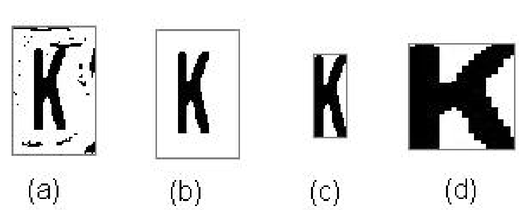

Fig. 3. Extracted character (a), cleaned (b), cropped (c) and normalized (d)

2) Statistics of the Extracted Character: Examples of statistics of an extracted character are the fill percentage and the projected foreground. The fill percentage is the proportion of foreground pixels in the binary image. If F is the matrix of the digitized binary image, then the fill percentage is $\sum_x \sum_y F(x,y)/(N_x N_y)$, where N_x and N_y are the height and width of the image in pixels. Figure 4 shows the mean within one standard deviation of the fill percentage of the training set.

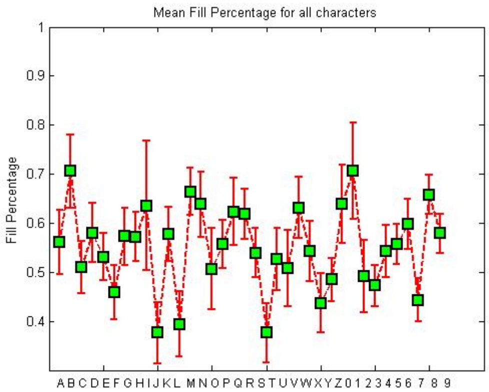

Fig. 4. Mean fill percentage for training set

The projected foreground is the normalized projection of the foreground on the x and y axes of the image, $\sum_y F(x,y)/N_y,\ x = 1,\ldots,N_x$ and $\sum_x F(x,y)/N_x,\ y = 1,\ldots,N_y$, respectively. These projected histograms seem to provide distinguishing information, as seen in Figure 5 where the mean of these statistics is plotted, within one standard deviation.

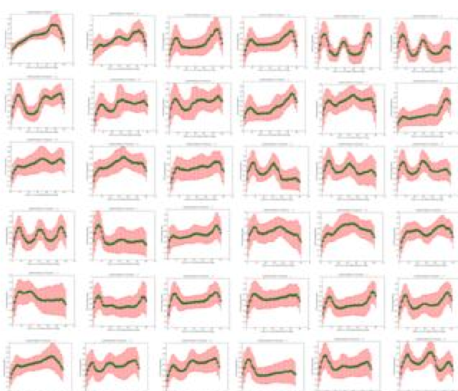

Fig. 5. Historical means of projected foreground for the 36 characters

These statistics can be used for the recognition of characters. In a preliminary study, we used the *minimization of squared errors* to attempt to recognize the characters using the normalized projection foreground:

$\mathbf{Minimize}_{C=A,B,\ldots,X,Y,Z,0,1,\ldots,8,9}$

$$\sum_x \Big(\sum_y F(x,y)/N_y - \mu_x^C\Big)^2 + \sum_y \Big(\sum_x F(x,y)/N_x - \mu_y^C\Big)^2$$

where μ_x^C and μ_y^C, $C = A,B,\ldots,X,Y,Z,0,1,\ldots,8,9$ are the historical means from the training set (Figure 5). The approach failed to return a good recognition. In an attempt to remedy to a possible lack of information from the statistics, the projected foreground was augmented with the distance of the pixels to the the side of the image, thus incorporating the information about the locations of the foreground pixels. The minimum squared errors method failed again, an indication that these features of the image do not offer information to fully distinguish among the characters.

B. The Probabilistic Approach

Let Z be the random variable that represents the statistical feature of a character image. For example, Z can be the fill percentage seen above. Note that Z need not, and often isn't a univariate. The probabilistic approach starts by building a probability model for that feature in the form of $\text{Prob}(Z|C)$, $C = A,B,\ldots,9$. For each image in the training set, the value of Z is computed. Data analysis tools are used, along with any engineering/prior knowledge to arrive at the probability model $\text{Prob}(Z|C)$. Seen as a function of the event C, that is the character is C, the probability model $\text{Prob}(Z|C)$ is known as the likelihood function $\mathcal{L}(C) = \text{Prob}(Z|C)$. This likelihood function is at the heart of the probabilistic approach. If this model is built properly, and the statistical feature Z offers enough information about the character's class, the probabilistic approach will be effective. An example of such likelihood function would be the Normal distribution for the data of Figure 6(a), and the Beta distribution for Figure 6(b). The data is from the fill percentage of the training sets for $C = 3$ and $C = 0$, respectively.

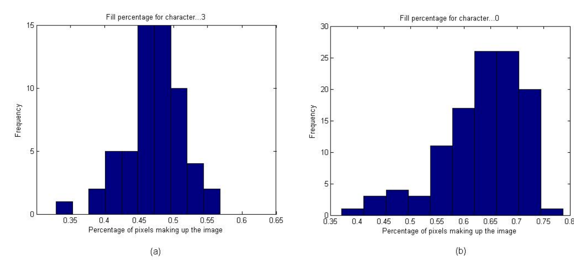

Fig. 6. Mean of projected foreground for training set

Given a likelihood model, the probabilistic approach proceeds as follows. Let z be the value of Z for an image being analyzed. Then the probability that the character is C, given the data z is

$$\text{Prob}(C|Z=z) = \frac{\text{Prob}(Z=z|C)\text{Prob}(C)}{\sum_{S=A}^{S=9}\text{Prob}(Z=z|S)\text{Prob}(S)}$$

for $C = A,B,\ldots,9$. $\text{Prob}(C)$ is the model built with any prior knowledge about what C might be. It is called the prior distribution. $\delta = \sum_{S=A}^{S=9}\text{Prob}(Z=z|S)\text{Prob}(S)$ is a normalizing constant. $\text{Prob}(C|Z=z)$ is the posterior distribution of C. The solution is given in the selected character $\hat{C}$, mode of the posterior distribution,

$$\text{Prob}(\hat{C}|Z=z) = \mathbf{Max}_{C=A,B,\ldots,9}\ \text{Prob}(C|Z=z).$$

If there is no prior knowledge as to what C might be, then the discrete uniform distribution is used where $\text{Prob}(C) = 1/36$ for $C = A, B, \ldots, 9$.

This probabilistic approach is often referred to as a *Bayesian* approach. A non Bayesian approach would simply maximize the likelihood function, ignoring the prior component, and select $\hat{C}$ such that

$$\text{Prob}(Z = z|\hat{C}) = \mathbf{Max}_{C=A,B,\ldots,9}\ \text{Prob}(Z = z|C).$$

As one can see, both these statistical approaches rely on the likelihood function. These simple operations are at the heart of many probabilistic predictions, classifications and inferences. While seemingly simple in principle, their success depends on the proper selection of the random variable Z and the probability model $\text{Prob}(Z|C)$.

IV. The Naive Bayes Classifier

However, finding the random variable Z and the probability model $\text{Prob}(Z|C)$ so that the solution is computationally tractable is not a trivial task. Much time was devoted to the research. Until we arrived at the conclusion that the values of the pixels in the binary image hold all the information needed to recognize the character in an image that has been cleaned, cropped and normalized (Figure 3). We therefore defined Z as the multidimensional vector of the values of all the pixels in the image. Each of these values is either 0 or 1, the input image being binary. For each pixel, we applied the Bernoulli probability model $\theta_i^{Z_i}(1-\theta_i)^{1-Z_i}$, Z_i being the value of Z at pixel i. Making the assumption of conditional independence of the pixel values given an image, we construct the likelihood function

$$\mathcal{L}(C) = \text{Prob}(Z|C) = \prod_{i=1}^{|Z|} \theta_i^{Z_i}(1-\theta_i)^{1-Z_i}$$

where $|Z|$ is the cardinal, or vector size, of Z. This independence assumption was made on an instinct, proving to be a tremendous leap to an efficient solution in the form of a Naive Bayes classifier. Once the assumption was made, it remained to estimate the parameters. To estimate the proportion θ_i for pixel i, a number of approaches are available. But given that the sizes of the historical sets are relatively large, in the order of 200 images per character $C = A, B, \ldots, X, Y, Z, 0, 1, \ldots, 8, 9$, the estimates converge to the average

$$\hat{\theta}_i = \sum_{j=1}^{N_C} x_{i,j}/N_C$$

where N_C is the size of the training set for character C, and $x_{i,j} = \{0 \text{ or } 1\}$ is the value of pixel i for image j of the training set. This is done for each character C. For simplicity of exposition, we used θ_i, when in fact it is a $\theta_i(C)$ that differs for each C. From a computational point of view, the assessment of the likelihood parameters is very simple. For each character C, all the images of the training set are added, then divided by the size of the set N_C, automatically providing a matrix of estimates $[\hat{\theta}_i]_{i=1}^{|Z|}$. This is a simple operation, inexpensive computationally, that replaces the training of a neural network. It needs to be done only once, and the estimates matrices are used subsequently to recognize the characters.

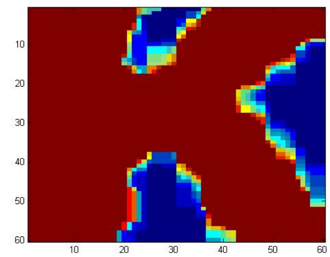

Fig. 7. Likelihood image of character K

Figure 7 shows the matrix $[\hat{\theta}_i]_{i=1}^{|Z|}$ for character K. Each value of the pixel i of the image in Figure 7 is the estimate $0 \leq \hat{\theta}_i \leq 1$ for character K. It is the estimate of the parameter of the Bernoulli model for pixel i, for character K. In essence, this method builds a likelihood function value for each character C, that results from a matrix, where the values of 1 signify that the corresponding pixels are always present in the foreground of C (in brown in Figure 7), the values 0 mean that the corresponding pixels are always background in C (in dark blue in Figure 7), and the values in between correspond to pixels i that are present in C with a probability $\hat{\theta}_i$. One observes that the K in the image is not a perfect one, as the extracted images of characters in LPR are most often taken at angles and subject to many sources of noise and deformation. In addition, the colors of the image do not show all the nuances in the values. But the matrix $[\hat{\theta}_i]_{i=1}^{|Z|}$ is computed with double precision accuracy and provides accurate estimates of the probabilities of the foreground existence. Figure 7 is included for the sake of illustration of the method.

A. Algorithm

Let z be the realization of the statistic Z for a binary image that has received similar cleaning, cropping and resizing as have the images of the training set. Then

$$\text{Prob}(C|z) = \frac{\text{Prob}(z|C)\text{Prob}(C)}{\sum_{S=A}^{S=9} \text{Prob}(z|S)\text{Prob}(S)}$$

where

$$\text{Prob}(z|C) = \prod_{i=1}^{|Z|} \hat{\theta}_i^{z_i}(1-\hat{\theta}_i)^{1-z_i}$$

noting that the appropriate $\hat{\theta}_i$'s correspond to a given C. This posterior probability distribution ranks the characters $A, B, \ldots, X, Y, Z, 0, 1, \ldots, 8, 9$ for their likelihood of being the character in the image being treated. One introduces prior knowledge in the form of $\text{Prob}(C)$ for each character C. Such knowledge can vary from country to country for example, where the position of a character in the license plate may indicate whether it is a letter or a number, for example. If such knowledge does not exist, or is hard to embody in a formal model, then the characters are declared to be equally probable apriori, letting Prob$(C) = 1/36$ in this case. This reduces, from a solution point of view, to maximizing the likelihood function. To further improve the speed and increase the accuracy, we note that maximizing the product of two bounded positive values is equivalent to maximizing their sum. Therefore we implemented the following:

$$\max_{C=A,B,\ldots,9} \mathrm{f}(C) = \sum_{i=1}^{|Z|} \{\hat{\theta}_i^{z_i} + (1-\hat{\theta}_i)^{1-z_i}\}$$

where $f(C)$ is the score function defined by the sum.

This method yielded excellent results. As most methods in this category, it fails to distinguish, as such, fully between some characters like 2 and Z, 5 and S, 1 and I, B and 8, and O,0,D and Q. However, using a similar logic and applying it exclusively to parts of the image, we reach a 97% reliability in [67]. The tests conducted performed well on both types of data mixed together, as indicated in table I.

TABLE I
OCR Reliability Results

Character	Reliability	Character	Reliability
A	0.9888	S	0.9385
B	0.9259	T	0.9828
C	0.9841	U	1.0000
D	0.9552	V	0.9306
E	0.9359	W	0.9898
F	1.0000	X	1.0000
G	0.9608	Y	1.0000
H	1.0000	Z	0.9722
I	0.9500	0	0.8636
J	1.0000	1	0.8864
K	1.0000	2	1.0000
L	1.0000	3	0.9649
M	1.0000	4	0.9899
N	0.9529	5	0.9623
O	0.9630	6	0.9756
P	1.0000	7	0.9560
Q	0.9231	8	0.8889
R	0.9545	9	0.9670

V. Conclusion

This was a breakthrough in solving our recognition problem, part of a much larger set of problems. The LPR optical character recognition falls within the realm of pattern recognition. While such problems often require sophisticated approaches due to the large number of patterns and unforseen ones, making it hard to use parsimonious probability models, the recognition problem we dealt with is much simpler. For the author who was not versed in all machine learning approaches, a rediscovery of the Naive Bayes approach was made. The author introduced a simple, and very inexpensive method to solve a relatively important problem.

References

[1] J-W. Hsieh, S-H. Yu and Y-S. Chen, Morphology-based license plate detection from complex scenes. Proceedings of the 16th International Conference on Pattern Recognition, Vol. 3, pp. 176- 179, 2002.

[2] F. Martin, M. Garcia and J. L. Alba, New Methods For Automatic Reading of VLP's (Vehicle License Plates). Signal Processing Pattern Recognition and application, 2002.

[3] H. Bai, J. Zhu, and C. Liu, A fast license plate extraction method on complex background. IEEE Proceedings on Intelligent Transportation Systems, vol. 2, pp. 985-987, 2003.

[4] D. Chanson and T. Roberts, License plate recognition system. Department of Electrical and Electronic Engineering, Manukau Institute of Technology, Auckland, 2002.

[5] B. Hongliang and L. Changping, A hybrid License Plate Extraction Method Based On Edge Statistics and Morphology. Proceedings of the 17th International Conference on Pattern Recognition (ICPR04), 2004.

[6] B. Enyedi, L. Konyha, C. Szombathy and K. Fazekas, Strategies for Fast License Plate Number Localization. 46th International Symposium Electronics in Marine, ELMAR-2004, Zadar, Croatia, 16-18 June 2004.

[7] Z. Qin, S. Shi, J. Xu and H. Fu, Method of License Plate Location Based on Corner Feature. Proceedings of the 6th World Congress on Intelligent Control and Automation, Dalian, China, June 21-23, 2006.

[8] C. Anagnostopoulos, I. Anagnostopoulos, G.Tsekouras, G. Kouzas, V. Loumos and E. Kayafas, Using Sliding Concentric Windows for License Plate Segmentation and Processing. IEEE Workshop on Signal Processing Systems Design and Implementation, pp. 337- 342, Nov. 2005.

[9] S.-L. Chang, L.-S. Chen, Y.-C. Chung and S.-W. Chen, Automatic License Plate Recognition. IEEE Transactions on Intelligent Transportation Systems, Vol. 5, No. 1, March 2004.

[10] 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. IEEE International Conference on Neural Networks, 1995.

[11] Y.Cui and Q. Huang, Automatic license extraction from moving vehicles. Proceedings of the International Conference on Image Processing, Vol. 3, pp. 126 - 129, 26-29 Oct. 1997.

[12] N. Zimic, J. Ficzko, M. Mraz, and J. Virant, The Fuzzy Logic Approach to the Car Number Plate Locating Problem. Proc. IIS, pp. 227230, 1997.

[13] V. Kamat and S. Ganesan, An efficient implementation of the Hough transform for detecting vehicle license plates using DSP, Proceedings of Real-time Technology and Applications Symposium, CA USA, IEEE Computer, Society Press, 1995.

[14] T. Duc Duan, T.L. Hong Du, T. Vinh Phuoc and N. Viet Hoang, Building an Automatic Vehicle License Plate Recognition System. Proceedings International Confererence on Computer Science, RIVF05, Can Tho, Vietnam, pp. 5963, Feb. 2005.

[15] E. R. Lee, P. K. kim and Hang Joon Kim, Automatic recognition of a Car License Plate Using Color Image Processing. Proceedings of international Conference on Image Processing, pp.301- 305, 1994.

[16] K.K.Kim, K.I.Kim, J.B.Kim and H. J.Kim, Learning-based Approach for License Plate Recognition. Proceedings of the 2000 IEEE Signal Processing Society Workshop, Sydney, NSW, Australia, Neural Networks for Signal Processing X, Vol.2, pp 614-623, Dec. 2000.

[17] X. Jianfeng, L. Shaofa and C. Zhibin, Color Analysis For Chinese Car Plate Recognition. Proceedings of the 2003 IEEE Changsha International Conference on Robotics, Intelligent Systems and Signal Processing, China, October 2003.

[18] F. Porikli and T. Kocak, Robust License Plate Detection Using Covariance Descriptor in a Neural Network Framework, avss, p. 107, IEEE International Conference on Video and Signal Based Surveillance (AVSS'06), 2006.

[19] W. Wei, Y. Li, M. Wang and Z. Huang, Research on Number-Plate Recognition Based on Neural Networks. Proceedings of the 2001 IEEE Signal Processing Society Workshop, Neural Networks for Signal Processing XI, North Falmouth, MA, USA, pp. 529-538, 2001.

[20] W. Jia, H. Zhang, X. He and M. Piccardi, Mean Shift for Accurate License Plate Localization. Proceedings of the 8th International IEEE Conference on Intelligent Transportation Systems Vienna, Austria, September 13-16, 2005.

[21] B.D. Acosta, Experiments in Image Segmentation for Automatic US License Plate Recognition. Master's thesis. Virginia Polytechnic Institute and State University, 2004.

[22] R, Parisi, E.D. Di Claudio, G. Lucarelli and G. Orlandi, Car Plate Recognition by Neural Networks and Image Processing. Proceedings of the 1998 IEEE International Symposium on Circuits and Systems, ISCAS '98. Vol. 3, pp. 195-198, 1998.

[23] K. Aboura, Stochastic Statistical Analysis of the DFT for Localization in an Image. Proceedings of the 2013 Pan African International Conference on Science, Computing and Telecommunications, PACT 2013, July 15 18, Lusaka, Zambia, 2013.

[24] R. Al-Hmouz anf K. Aboura, License Plate Localization Using a Statistical Analysis of Discrete Fourier Transform Signal. Computers and Electrical Engineering, Vol. 40, No. 3, pp. 982-992, 2014.

[25] F. Kahraman, B. Kurt, M. Gkmen, License Plate Character Segmentation Based on the Gabor Transform and Vector Quantization. ISCIS, pp. 381-388, 2003.

[26] D. Liu and M. Xie, Detecting License Plate based on Wavelet Transform from Complex Scene. Proceeding (449) Circuits, Signals, and Systems, 2004.

[27] C.-T. Hsieh, Y.-S. Juan and K.M. Hung, Multiple License Plate Detection for Complex Background. Proceedings. The 19th International Conference on Advances Information Networking and Applications, 2005.

[28] D.-S. Kim S.-I. Chien, Automatic Car License Plate Extraction Using Modified Generalized Symmetry Transform and Image Warping. Proceedings of the IEEE International Symposium on Industrial Electronics, Vol.3, pp. 2022-2027, 2001.

[29] S. Yohimori, Y. Mitsukura, M. Fukumi, N. Akamatsu and N. Pedrycz, License Plate Detection System by using Threshold Function and Improved Template Matching Method. IEEE Annual Meeting of the Fuzzy Information, Processing NAFIPS '04, Vol.1, pp 357-362, 27-30 June 2004.

[30] J.R. Parker and P. Federl, An Approach To Licence Plate Recognition. Vision Interface '97, Kelowna, B.C., May 20-22, 1997.

[31] L. Dlagnekov, Video-based Car Surveillance: License Plate, Make, and Model Recognition. Master's thesis. University of California, San Diego, 2005.

[32] H. A.-Q. Maarif and S. Sardy, Plate Number Recognition by Using Artificial Neural Network. Prosiding Semiloka Teknologi Simulasi dan Komputasi serta Aplikasi, 2006.

[33] J. Kong, X. Liu, Y. Lu, and X. Zhou, A Novel License Plate Localization Method Based On Textural Feature Analysis. IEEE International Symposium on Signal Processing and Information Technology, 2005.

[34] Zhang Sanyuan, Zhang Mingli and Ye xiuzi, Car Plate Character Extraction under Complicated Environment. International Conference on Systems, Man and Cybernetics, 2004.

[35] V. Shapiro and G. Gluhchev, Multinational License Plate Recognition System: Segmentation and Classification. Proceedings of the 17th International Conference on Pattern Recognition, ICPR 2004, Vol. 4, pp. 352-355, 23-26 Aug. 2004.

[36] C. A. Rahman, W. Badawy and A. Radmanesh, A Real Time Vehicles License Plate Recognition System. Proceedings of the IEEE Conference on Advanced Video and Signal Based Surveillance (AVSS03), 2003.

[37] H. Hontani and T. Koga, Character Extraction Method without Prior Knowledge on Size and Information. Proceedings of the IEEE International Vehicle Electronics Conference (IVEC01), pp. 67-72, 2001.

[38] F. Yang, Z. Ma and M. Xie, A Novel Approach for License Plate Character Segmentation. 1st IEEE Conference on Industrial Electronics and Applications, 2006.

[39] Y. Zhang and C. Zhang, A New Algorithm for Character Segmentation of License Plate. Proceedings of the IEEE Intelligent Vehicles Symposium, 2003.

[40] Y.Cui and Q. Huang, Character Extraction of License Plates from Video. Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, pp. 502-507, June 1997.

[41] M.-A. Ko and Y.-M. Kim, License Plate Surveillance System Using Weighted Template Matching. Proceedings of the 32nd Applied Imagery Pattern Recognition Workshop, pp. 269-274 , Oct. 2003.

[42] S.-Z. Wang and H.-J. Lee, Detection and Recognition of License Plate Characters with Different Appearances. Intelligent Transportation Systems, Proceedings IEEE, vol. 2, pp. 979-984, 2003.

[43] Christoph Busch, Ralf Dorner, Christian Freytag and Heike Ziegler, Feature Based Recognition of Traffic Video Streams for Online Route Tracing. Vehicular Technology Conference,48th IEEE, Ottawa, Ont., Canada, Vol. 3, pp. 1790-1794, 18-21 May 1998.

[44] P. Comelli, P. Ferragina, M. N. Granieri and F. Stabile, Optical Recognition of Motor Vehicle License Plates. IEEE Transactions on Vehicular Technology, Vol. 44. no. 4, Nov. 1995.

[45] K. Yamaguchi, Y. Nagaya, K. Ueda, H. Nemoto and M. A. Nakagawa, A Method for Identifying Specific Vehicles Using Template Matching. Proceedings of the 1999 IEEE/IEEJ/JSAI International Conference on Intelligent Transportation Systems, Japan, pp. 8-13, 1999.

[46] 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. 23092319, Nov. 2000.

[47] M.M.M. Fahmy, Automatic Number-plate Recognition : Neural Network Approach. Proceedings of VNIS94 Vehicle Navigation and Information System Conference, 31 Aug - 2 Sept, 1994.

[48] J. Matas and K. Zimmermann, Unconstrained Licence Plate and Text Localization and Recognition. Proceedings of the 8th International IEEE Conference on Intelligent Transportation Systems Vienna, Austria, September 13-16, 2005.

[49] A. da Rocha Gesualdi and J. Manoel de Seixas, Character Recognition in Car License Plates Based on Principal Components and Neural Processing. Proceedings of the VII Brazilian Symposium on Neural Networks, pp. 206-211, 2002.

[50] M.-A. Ko and Y.-M. Kim, A simple OCR method from strong perspective view. Proceedings of the 33rd Applied Imagery Pattern Recognition Workshop, pp. 235-240, Oct. 2004.

[51] T. Shuang-tong and L. Wen-ju, Number and Letter Character Recognition of Vehicle License Plate Based on Edge Hausdorff Distance. Proceedings of the Sixth International Conference on Parallel and Distributed Computing, Applications and Technologies (PDCAT05), 2005.

[52] F. Odone, Experiments on a License Plate Recognition System, Technical Report No. DISI-TR-06-17, DISI, Universita degli Studi di Genova, Italy, 2006.

[53] D. Llorens, A. Marzal, V. Palazon, and J. M. Vilar, Car License Plates Extraction and Recognition Based on Connected Components Analysis and HMM Decoding. Lecture Notes on Computer Science, Vol. 3522, J. S. Marques et al., Eds. New York: Springer-Verlag, 2005.

[54] D. Patterson, *Artificial Neural Networks*. Singapore: Prentice Hall, 1996.

[55] M.Horowitz, Efficient use of a picture correlator. J. Opt. Soc. Am, vol. 47, pp. 327, 1957.

[56] W. K. Pratt, Correlation Techniques of iIage Registration. IEEE Trans., vol AES-10, pp. 353-358, 1974.

[57] C. Anagnostopoulos, E. Kayafas, and V. Loumos, Digital Image Processing and Neural Networks for Vehicle License Plate Identification. J. Elect. Eng. [Online]. 1(2), pp. 27, 2000. Available: http://www.medialab.ntua.gr/people/canag/journals.php

[58] C. Anagnostopoulos, I. Anagnostopoulos, V. Loumos and E. Kayafas, A License Plate-Recognition Algorithm for Intelligent Transportation System Applications. IEEE Transactions on Intelligent Transportation Systems, Vol. 7, No. 3, Sept. 2006.

[59] Y. Hu, F. Zhu, and X. Zhang, A Novel Approach for License Plate Recognition Using Subspace Projection and Probabilistic Neural Network. Lecture Notes on Computer Science, Vol. 3497, pp. 216221, J. Wang, X. Liao, and Z. Yi, Eds. New York: Springer-Verlag, 2005.

[60] D. F. Specht, Probabilistic Neural Networks for Classification, Mapping or Associative Memory. IEEE International Conference on Neural Networks, Vol.1, pp. 525-532, 1988.

[61] D. F. Specht, Probabilistic Neural Networks. Neural Networks, Vol. 3, pp. 109-118, 1990.

[62] C. M. Bishop, Neural Networks for Pattern Recognition. Oxford University Press, New York, pp. 1723, 1997.

[63] J. F. Wang, C. C. Huang, C. H. Wu, Y. S . Lu and J. Y. Lee, A Speaker Dependent Connected Mandarin Speech Recognition System with Incremental Learning Ability. Proceedings of 4th International Symposium on IC Design, Manufacture and Applications, September 1991.

[64] Otsu N., A Threshold Selection Method from Gray-Level Histogram. IEEE Trans. System Man Cybernetics, Vol. 9, No. 1, pp. 62-66, 1979.

[65] J. Canny, A computational approach to edge detection. IEEE TPAMI, Vol. 8, No. 6, pp. 79698, 1986.

[66] K. Aboura, Automatic Thresholding of License Plate. International Journal of Automation and Control, Vol. 2, Nos. 2-3, pp. 213-231, 2008.

[67] K. Aboura and R. Al-Hmouz, Probabilistic License Plate Optical Character Recognition. Technical Report, Computer Systems Engineering Department, University of Technology Sydney, Australia, 2007.