Journal of Advances in Computer Research
Quarterly pISSN: 2345-606x eISSN: 2345-6078
Sari Branch, Islamic Azad University, Sari, I.R.Iran
(Vol. 6, No. 4, November 2015), Pages: 27-38
www.jacr.iausari.ac.ir

Car License Plate Recognition using Color Features of Persian License Plates

Amir Ebrahimi✉[1], Abdollah Amirkhani[2], Abolghasem A. Raie[1], Mohammad R. Mosavi[2]

1) Dept. of Electrical Engineering, Amirkabir University of Technology, Tehran 15875-4413, Iran
2) Dept. of Electrical Engineering, Iran University of Science and Technology, Tehran, Iran

amir.ebrahimi66@aut.ac.ir; amirkhani@ieee.org; raie@aut.ac.ir; m_mosavi@iust.ac.ir

Received: 2015/02/02; Accepted: 2015/04/19

Abstract

Car license plate recognition is addressed in this paper. Given the development of intelligent transportation systems, it is absolutely essential to implement a strong license plate recognition system. Efforts were made to put forward a novel reliable method for car license plate recognition in Iran. Each license plate recognition system comprises three main parts. The first part is the license plate detection stage. The blue color feature of the license plate margin along with Scale-Invariant Feature Transform (SIFT) algorithm were used for this purpose. The accuracy of the presented method over the database was approximately 90% in less than a second. License plate morphological features were utilized upon character segmentation. Using these features, areas with sizes close to that of the characters of a license plate may be searched. The accuracy of this method was almost 95%. A probabilistic neural network together with a Support Vector Machine (SVM) was employed at the character recognition stage. For this stage, an accuracy of nearly 97% in 55 milliseconds for each license plate was achieved.

Keywords: *License plate recognition; SIFT; SVM; Intelligent transportation systems*

1. Introduction

The booming auto industry has led to increasing urban and non-urban transportation rendering traffic supervision and extracting traffic parameters by human a very difficult, sometimes impossible task. The growing trend of auto manufacture and these autos appearing in cities have caused numerous problems especially for auto recognition regarding different applications such as traffic control, monitoring roads and highways and ensuring their security, etc. In case new technologies are not used for solving these problems, a great deal of financial and human resources should be brought into play and it is likely that this, in the end, would be unable to fully meet our requirements for solving the plethora of problems. That is why the concept of Intelligent Transportation Systems (ITS) has been brought up. The automatic license plate recognition system is one of the fundamental components of an ITS. Car license plate recognition in public parking lots, controlling and receiving toll for entry into the traffic plan area, receiving road and highway toll, and stolen car recognition are among the applications of the automatic license plate recognition system [1].

The car license plate recognition system brings about the possibility of automatically extracting the license plate number from the video or the still image by the computer, allowing it to be used in form of separate characters. The car license plate number recognition involves three main steps: car license plate detection, license plate character segmentation and character recognition. This paper covers all the three parts of a car license plate recognition system. Various works have been carried out on car license plate recognition, whether on foreign or Persian license plates. In [1] and [2], a comprehensive review of a great number of existing methods was presented. Persian license plate recognition remains a challenge [3-5].

This paper aims at implementing an accurate rapid algorithm for license plate recognition, in a manner that it can be used in real-time applications. The rest of the paper is organized as follows: At first, all the three main steps of car license plate recognition, i.e. car license plate detection, license plate character segmentation and character recognition are described in detail. Further, implementation results are presented and a conclusion is drawn.

2. License Plate Detection

The methods based on combination of the statistical and morphological properties have good results in extracting the license plate from the background image. In [6], smoothed and normalized edge operators on gray image are used to obtain the horizontal and vertical edge maps. Then, statistical analysis of edges was utilized to identify the edges of rectangular plate. Next, this method in a hierarchical process was carried out again at different scales. Ultimately, the final decision was made based on connected component analysis (CCA).

The authors claim to have an accuracy of 99.6% over 9825 images. Many of the other plate recognition algorithms [7, 8] have followed a similar trend although these methods assume that the images of vehicle license plates have sharp and vertical edges. Thus, if the image of vehicle license plates has no clear and sharp edges or it is rotated, the results will not be reliable.

High contrast between the characters and the background [9] is used to identify the location of the plates that have black characters on a white background. While, other methods assume that density of edges in vehicle license plate are more than the other regions in images in which contrast between the characters and the background is high. In [10, 11], for instance, N-Row distance method is employed for counting the available edges of the image. High edge-density areas are more likely to have plates. Similarly, in [12], a Block-based method was used where the blocks with high amount and variance of edge were selected as the area with license plate.

In many countries, the vehicle license plate is standard for all cars. And, color and background are fixed. So, many algorithms can use color information to identify the position of the vehicle license plate [13, 14]. However, if the lighting conditions change, the color of vehicle license plates also changes. Therefore, methods that only depend on the color will have not final appropriate accuracy.

The proposed method is based on finding the blue margin of the Persian license plates. The standard format of Persian license plates contains this blue margin and it may be found in all car license plates [15]. In the invented method that embraces the main idea behind this paper, the entire blue regions of the image are initially extracted. These blue regions include the license plate region in addition to other blue points. Then, all of the blue regions obtained from the image are labeled. These regions are compared with the blue strip at the car license plate margin in terms of dimensions and aspect ratio in the database image. In case their dimensions are not close to those of a license plate margin, they are eliminated and a number of candidates finally remain. Among the remaining candidates, the license plate margin features should be sought. In the proposed system, this is done using SIFT algorithm [16]. Once the position of the license plate margin is found, the entire license plate region is also obtained considering the dimensions and the format of standard license plates [15]. In what follows, the above-mentioned stages will be described in detail.

2.1 Extracting Blue Regions

Consider Figure 1 to be the image fed to the license plate recognition system. At first, the RGB components of the color image are separated. Component B being higher than other image components in a certain region is the blueness criterion of that region. Extracting these regions gives Figure 2.

Figure 1. Main image

In this figure, the parts with white color pertain to regions where component B of the image has a higher value compared with other components. Other parts of the image are shown in black. The degree to which the blue color component is higher than others depends on the brightness of database images.

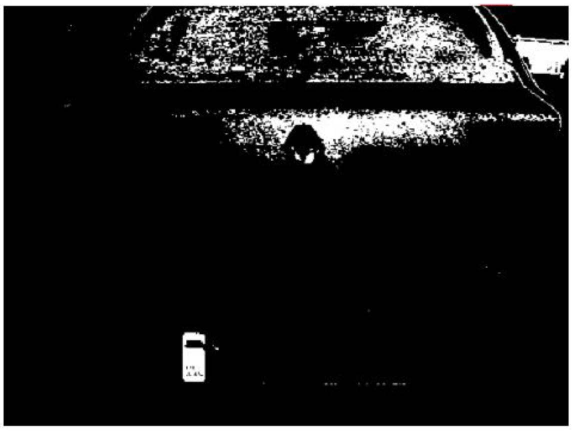

Figure 2. Extracting blue regions from the main image

2.2 Eliminating Unlikely Candidates

In this section, the blue regions obtained from the previous section that are not in the size of an actual license plate margin are eliminated in view of the database and the license plate sizes existing in it. These sizes should be introduced to the system by system user and in consideration of the imaging conditions. Once the unlikely parts are eliminated, only the acceptable blue color regions are distinguished in the following image.

2.3 Searching using SIFT Algorithm

A number of candidates may be obtained from the previous stage. Among the remaining candidates, SIFT algorithm conducts a search. SIFT is a machine vision algorithm employed for searching and describing the local feature of an image's key points for object recognition. In this method, describing an appropriate feature of an object plays a very important role in identifying and finding that object in different images. Hence, all of these features should be robust to image scale, noise, and brightness, so that identification and recognition would be performed reliably. Moreover, SIFT algorithm's feature descriptor is unchangeable in relation to rotation and scale. The key points of SIFT should thus be extracted from a sample license plate margin and be used for finding license plates in the database.

Figure 3. Eliminating unlikely candidates and labeling the license plate blue margin

The output of this algorithm yields the result as to which of the blue color regions are considered by SIFT descriptors to be close to the blue strip at the license plate margin. In this manner, non-license plate options are eliminated and thereby, merely the blue strip of the license plate margins in the images remains. Therefore, the proposed system is capable of simultaneously detecting several license plates in an image. In the end, in light of the detected blue strips and the dimensions of the car license plate in comparison with these blue strips, the car license plate is extracted as Figure 4.

Figure 4. Recognition of the entire license plate using the blue margin

It is noteworthy that SIFT was also previously utilized for car license plate detection. For instance, SIFT was used in [17] for the detection of the entire license plate in the image given the saved training samples. Since the processing operation is performed on the entire image, the license plate extraction speed is very low.

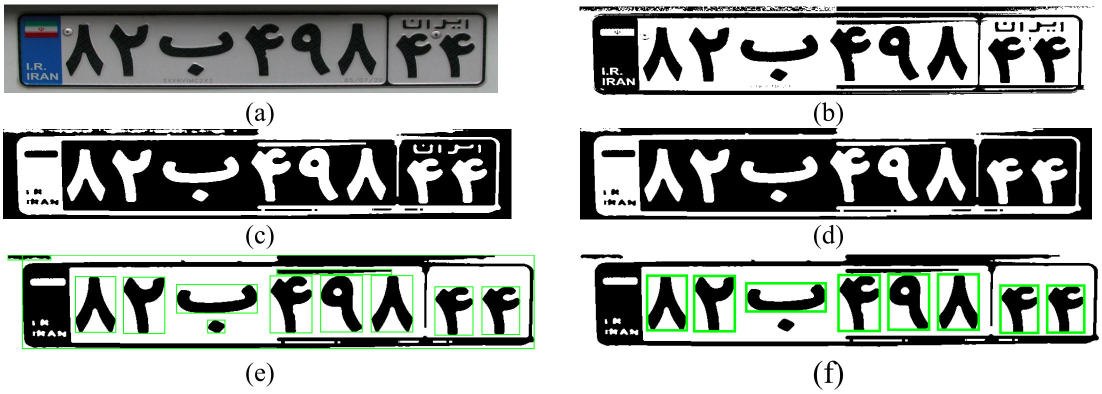

Figure 5. (a) Main image (b) Binarized image (c) Inversed image with noise cancellation (d) Eliminating small objects from the image (e) Labeling the objects in the image (f) All of the acceptable objects in terms of size

3. Character Segmentation

The most common method used to segment characters is horizontal and vertical projection of the binary image of license plate [11, 18, 19]. The idea used is to calculate the sum of the dark sides of the license plate in horizontal and vertical directions. Thus, the gap between the characters is obtained and characters are separated. Mathematical morphology method is used for segmenting characters in [19] that can be effective for low quality images. In this method, morphological algorithms such as thinning and thickening are used.

In order to segment characters, a method similar to that of [20] is adopted. In this method, a local binarization is at first performed on the license plate using Otsu's method for Figure 5.a [21]. This binarization results in Figure 5.b. The image is then inversed and a median filter is applied to the image for noise cancellation (Figure 5.c). Further, the small objects in the image are also eliminated and the image would be as Figure 5.d. The objects in the license plate image are then labeled. Figure 5.e shows all labeled objects. Considering the size of the characters, the only acceptable objects are demonstrated in Figure 5.f.

4. Character Recognition

The most important classifier used for character recognition is Neural Network. Various neural networks have been used for this purpose. In [22], a multilayer neural network with an accuracy of 98.5% was used. In [23], the self-organizing neural networks is employed to identify the characteristics of noisy, deformed, broken or defective characters with an accuracy of 95.6% on a large database. Due to high-speed training, probabilistic neural networks are widely used in the recognition of vehicle license plate characters. Remarkable accuracy of 99.5% was also reported in 2005 [24]. In recent years, the use of SVM in License Plate Recognition characters has been very popular. For example, in [18], an accuracy of 97.2% was reported in Korea on vehicle license plate characters using SVM.

The advantages of SVM have been highlighted in various papers so far. Diverse solutions have been presented to render SVM multi-class and the advantages and disadvantages of each have been pointed out. Misclassification is an important issue in character recognition. It means that characters with similar shape might be misclassified in classification step. A solution called MLP-SVM, with proper accuracy is put forward in [25]. In this method, the first and the second largest output of a neural network are selected as most probable classes and then, an SVM makes the final decision between them. In this method, first, the dataset is broken into three subsets as Set1, Set2 and Set3. Set1 is used for training the neural network and recognizing the most similar classes. Set2 is used to test the neural network and select the samples of similar classes which have largest output of the neural network. These samples are used for training an SVM. Also, Set3 is used for testing the overall system. The diagram of this method is shown in Figure 6.

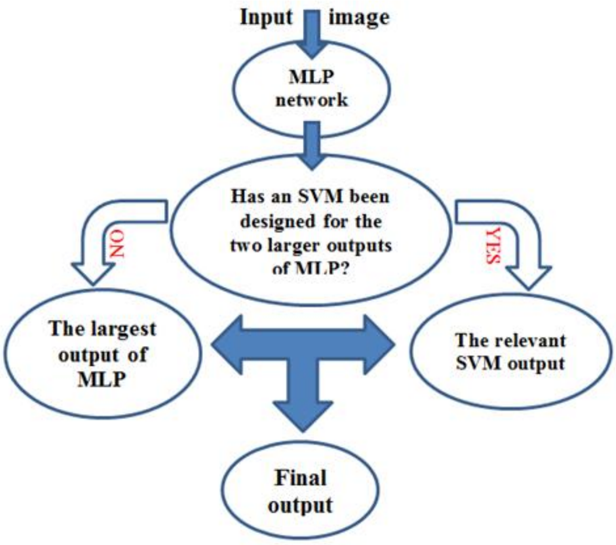

Figure 6. Schematic of the proposed method in [25]

As is observed, for each character, at most, one neural network and one SVM are executed, with an acceptable computation time. Since the output of the neural network is not probabilistic, there is no guarantee for the improvement of accuracy by the SVM. The major defect of this method is that the neural network output is not probabilistic and two larger outputs do not necessarily represent two outputs with higher probability.

In the proposed method, the most probable outputs will initially be obtained using a probabilistic classifier. The final decision will then be made by an SVM [26]. This method which is called PNN-SVM, has a training process which is simpler, more conceptual, and more intuitive compared with that of multi-layer neural networks. Furthermore, the number of its parameters is much fewer than those of multi-layer neural network. The probabilistic classifier used here is a probabilistic neural network [27]. Suppose we have N patterns and each one has a d-dimensional feature vector according to K classes. PNN consists of d input units, and all of them are connected to all N pattern units. Each pattern unit is connected to one of Koutput units. First, each training sample x_{ji} is divided on $\left(\sum_{i=1}^{d} x_{ji}^2\right)$ for normalization purpose. Then, network's weights are equated to training samples:

$$w_{jk} = x_{jk} \tag{1}$$

For a test sample, after normalization, the pattern units calculate inner product:

$$z_k = w_k^t . x \tag{2}$$

And then, emits a nonlinear function of z_k. Here, function $e^{\frac{z_k - 1}{\sigma^2}}$ with optimized value $\sigma = 0.26$ (by using trial and error) is used. Each output unit sums the contributions from all pattern units connected to it. The class with larger output is selected as the winner. Test algorithm is as below:

ALGORITHM

1 ***begin initialize*** $k = 0, x = test\ pattern$
2 *do* $k \leftarrow k + 1$
3 $z_k = w^t . x$
4 $g_c \leftarrow g_c + \exp(\frac{z_k - 1}{\sigma^2})$
5 ***until*** k=K
6 ***return*** $class = \arg\max g_i(x)$
7 ***end***

5. Empirical Results

5.1 Test Bed

The license plate recognition system of this paper is implemented on a computer with a 2.67GHz Core2 Quad CPU and 3.5GB of RAM. MATLAB2010 is used for simulation. The database of auto images comprises 516 images obtained from Bani Nik Pardazesh Company [28] and contains various images of autos with Farsi license plates in real conditions, i.e. in the day, at night, at different distances and angles. The license plate is shadowed or muddy in a number of images making license plate recognition difficult [15].

5.2 Results

Color-based features and the properties of SIFT are used for license plate detection. The accuracy of this method is approximately 90% in less than a second. At the character segmentation stage, the characters are properly segmented using edge detection, noise cancellation, and morphological operations. This method was evaluated for character segmentation over the database of car license plate images comprising 516 license plates. The result was the successful classification of 95.7% of characters. The most important obstacles to successful character segmentation in the conducted tests were shadow, steep angle, muddiness, and rivet effect. The average processing time required for classifying the characters of each license plate was 137 milliseconds. Figures 7 and 8 depict sample images where the employed method has succeeded or failed.

At the character recognition stage, a 15×20×24 structure was considered for the MLP-SVM method and 1200 characters were used for training (50 samples for each class). Identical training data and SVM parameters were utilized for all classifiers. The difference between MLP-SVM and PNN-SVM classifiers lies merely in the classification type of the most probable characters before putting SVM to use.

Figure 7. Successful character segmentation using the proposed segmentation method

Table 1 shows the accuracy and speed of implemented methods on the test data. All the methods in this table use SVM. The second column is accuracy and the third column is the run time for the whole test samples. In this table, DAG method [29] uses SVM in a hierarchical structure. ML-SVM [23] is similar to PNN-SVM and the only difference is using Maximum Likelihood model instead of PNN. In NSVM [30], for N classes, from N(N-1)/2 possible binary SVM classifiers N/2 are selected randomly. Outputs of them are given to a neural network as input and finally, the output of this neural network is the final output of this method. As shown in the fourth column of Table 1, PNN-SVM exhibits better processing speed accuracy compared with the other methods.

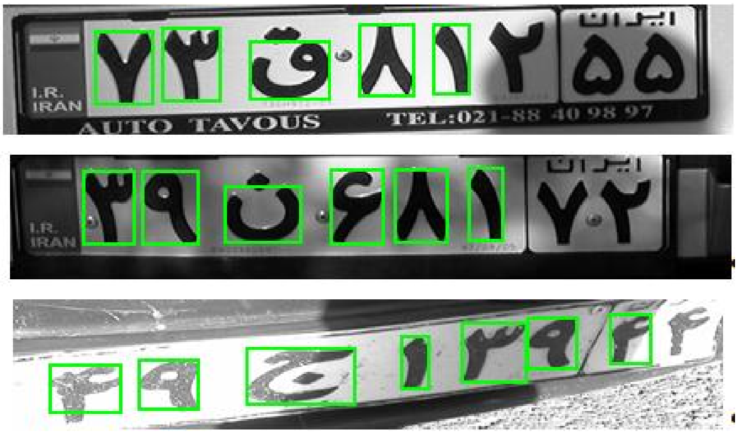

Figure 8. Unsuccessful character segmentation using the proposed segmentation method

PNN networks are trained in a fraction of the time required for training conventional multi-layer neural networks with error back-propagation algorithm. These networks enjoy a parallel structure; hence, using parallel processing techniques increases the output obtaining speed. Additionally, the classes or number of patterns of each class may be increased by adding the necessary nodes. It should be noted that neural networks have a relatively great number of parameters for training.

Table 1. Accuracies and recognition speeds obtained for several methods.

Method	Accuracy (%)	Testing time (s)	Recognition time (ms)
PNN-SVM	96.70	67.89	57.37
ML-SVM [25]	95.87	118.04	99.76
MLP-SVM [26]	95.86	109.93	92.90
PNN [31]	96.23	60.41	51.05
DAG [29]	94.89	127.96	108.14
NSVM [30]	91.94	120.37	101.73

The sample presented here is an appropriate one in that it displays a proper response over the database. This simple is definitely neither the best one nor an optimal neural network. Instances of images pertaining to the errors of the PNN-SVM method are shown in Figure 9. As the characters of this figure demonstrate, a number of errors cannot be easily distinguished even with the eye.

Also, the PNN has performed better when one character was to be chosen as the output and with the increase in the number of more probable characters, the PNN has been more successful although the MLP-SVM ultimately exhibits the same performance. The difference between the two classifiers of MLP-SVM and PNN-SVM is in the way of separating the most probable characters before applying the SVM.

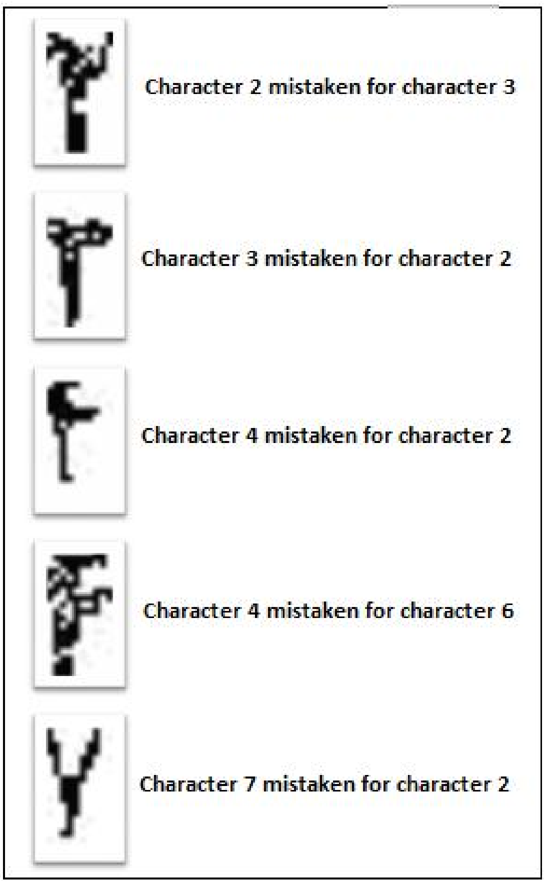

Figure 9. Mistakenly classified characters in the PNN-SVM method

Figure 10 shows the percentage of the presence of the correct character when choosing more than two probable characters for all the two mentioned approaches. As can be observed, the PNN-SVM method has had a better performance in recognizing the probable character, compared with the MLP-SVM method.

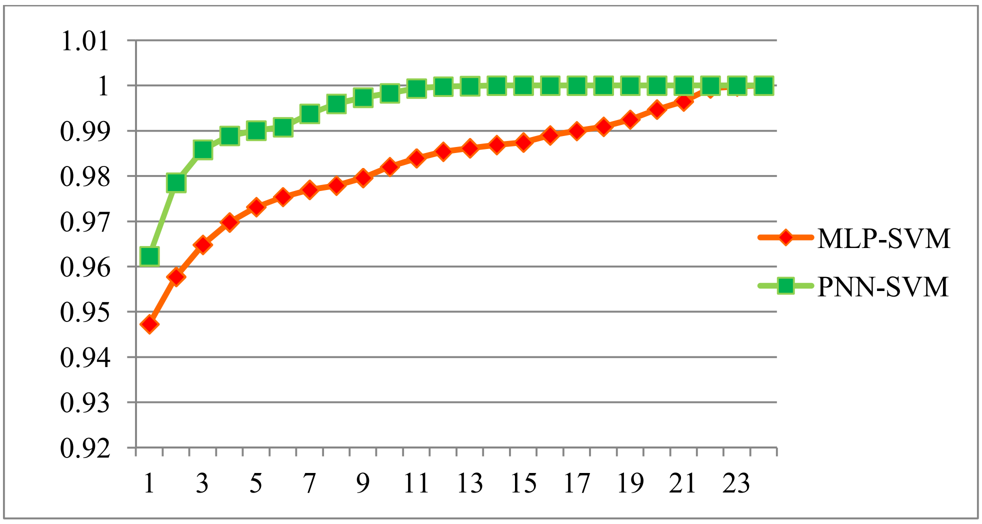

Figure 10. The probability of the presence of the correct character when choosing more than two probable characters

6. Conclusion

A novel method for license plate recognition of Persian cars was put forward. The license plate position recognition of the proposed method made use of the color features of the license plate as well as SIFT algorithm. Employing these two simultaneously increases the speed of license plate detection compared with the case where SIFT alone is used. The accuracy of this method over the database of car license plate images comprising 516 license plates is almost 90% in less than a second. At the character segmentation stage, using edge detection, noise cancellation, and morphological operations have brought about the successful segmentation of 95.7% of the characters. At the character recognition stage, a probabilistic neural network together with SVM was utilized with an accuracy of 96.70% for recognizing each character in nearly 55 milliseconds for each license plate. Noise, shadow, complex background, and rivet are the major causes of error at the stages of license plate recognition. This suggests that a series of preprocessing operations aiming at eliminating the above-mentioned items can enhance the accuracy of the license plate recognition system by increasing the computational load.

References

[1] C. Anagnostopoulos, I. Anagnostopoulos, I. D. Psoroulas, V. Loumos and E. Kayafas, "License Plate Recognition from Still Images and Video Sequences: A Survey", IEEE Transactions on Intelligent Transportation Systems, Vol. 9, No. 3, pp. 371-391, 2008.

[2] S. Du, M. Ibrahim, M. Shehata and W. Badawy, "Automatic License Plate Recognition (ALPR): A State-of-the-Art Review", IEEE Transactions on Circuits and Systems for Video Technology, Vol. 23, No. 2, pp. 311-325, 2013.

[3] M. Akhtari, K. Faez, "The Application of a CICA Neural Network on Farsi License Plates Recognition", in Proceedings of IEEE, 10th International Conference on Hybrid Intelligent Systems, Atlanta, GA, pp. 205-208, 2010.

[4] A. E. Ghahnavieh and A. Raie, "License Plate Character Recognition Using Multiclass SVM", Journal of American Science, Vol. 8(1s), pp. 38-42, 2012.

[5] A. A. Shahraki, A. E. Ghahnavieh and S. A. Mirmahdavi, "License Plate Extraction from Still Images", in Proceedings of IEEE, 4th International Conf. on Intelligent Systems, Modelling and Simulation (ISMS), Bangkok, Thailand, pp. 45-48, 2013.

[6] B. Hongliang and L. Changping, "A hybrid license plate extraction method based on edge statistics and morphology", in Proceedings of the Pattern Recognition, 17th International Conference on (ICPR), Vol.2, pp. 831-834, 2004.

[7] D. Zheng, Y. Zhao, and J. Wang, "An efficient method of license plate location," Pattern Recogn. Lett, Vol. 26, pp. 2431-2438, 2005.

[8] P. V. Suryanarayana, S. K. Mitra, A. Banerjee, and A. K. Roy, "A morphology based approach for car license plate extraction", Annual IEEE in INDICON, pp. 24-27, 2005.

[9] P. Comelli, P. Ferragina, M. N. Granieri, and F. Stabile, "Optical recognition of motor vehicle license plates", IEEE Transactions on Vehicular Technology, Vol. 44, pp. 790-799, 1995.

[10] J. Cano and J. C. Pérez-Cortés, "Vehicle License plate segmentation in natural images", ALLEMAGNE: Springer, Vol. 26-52, 2003.

[11] J. Barroso, E. L. Dagless, A. Rafael, and J. Bulas-Cruz, "Number plate reading using computer vision", in Proceedings of the IEEE International Symposium on Industrial Electronics ,Vol. 3, pp. 761-766, 1997.

[12] L. Hsi-Jian, C. Si-Yuan, and W. Shen-Zheng, "Extraction and recognition of license plates of motorcycles and vehicles on highways", the 17th International Conference on the Proceedings of Pattern Recognition, Vol. 4, pp. 356-359, 2004.

[13] L. Miao, F. Wang, and H. Wang, "Automatic license plate detection based on edge density and color model", in Control and Decision Conference, Chinese, pp. 3718-3721, 2009.

[14] K. Deb and J. Kang-Hyun, "HSI color based vehicle license plate detection", ICCAS , pp. 687-691, 2008.

[15] A. E. Ghahnavieh, M. Enayati and A. A. Raie, "Introducing a large dataset of Persian license plate characters", Journal of Electronic Imaging, Vol. 23, No. 2, 2014.

[16] D. G. Lowe, "Object recognition from local scale-invariant features", in Proceedings of IEEE, the International Conference on Computer Vision, Vol. 2, pp. 1150-1157, 1999.

[17] Morteza Zahedia and Seyed Mahdi Salehi, "License Plate Recognition System Based on SIFT Features", Procedia Computer Science, Vol. 3, pp. 998-1002, 2011.

[18] K. K. Kim, K. I. Kim, J. B. Kim, and H. J. Kim, "Learning-based approach for license plate recognition", in Neural Networks for Signal Processing X, Proceedings of IEEE Signal Processing Society Workshop, pp. 614-623 vol. 2, 2000.

[19] S. Nomura, K. Yamanaka, O. Katai, H. Kawakami, and T. Shiose, "A novel adaptive morphological approach for degraded character image segmentation", Pattern Recognition, Vol. 38, pp. 1961-1975, 2005.

[20] S. Nomura, K. Yamanaka, O. Katai, H. Kawakami, and T. Shiose, "A novel adaptive morphological approach for degraded character image segmentation", ELSEVIER: Pattern Recognition, Vol. 38, No. 11, pp. 1961-1975, 2005.

[21] Otsu, N., "A Threshold Selection Method from Gray-Level Histograms", IEEE Transactions on Systems, Man, and Cybernetics, Vol. 9, No. 1, pp. 62-66, 1979.

[22] 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", in Proceedings of IEEE, the International Conference on Neural Networks, Vol.5, pp. 2232-2236, 1995.

[23] C. Shyang-Lih, C. Li-Shien, C. Yun-Chung, and C. Sei-Wan, "Automatic license plate recognition", IEEE Transactions on Intelligent Transportation Systems, Vol. 5, pp. 42-53, 2004.

[24] Y. Hu, F. Zhu, and X. Zhang, "A novel approach for license plate recognition using subspace projection and probabilistic neural network Vol. 3498. Berlin, ALLEMAGNE: Springer, 2005.

[25] A. Bellili, M. Gilloux, P. Gallinari, "An MLP-SVM Combination Architecture for Offline Handwritten Digit Recognition", International Journal on Document Analysis and Recognition, Vol. 5, No. 4, pp. 244-252, 2003.

[26] A. E. Ghahnavieh, and A. A. Raie, "Enhancing the License Plates Character Recognition Methods by Means of SVM", in Proceedings of IEEE, 22nd Iranian Conference on Electrical Engineering, Tehran, Iran, 2014.

[27] Richard O. Duda, Peter E. Hart and David G. Stork, Pattern classification, Wiley, Second edition, Chapter 4, pp. 15-18, 2001.

[28] www.baninick.com

[29] John C. Platt, Nello Cristianini and John Shawe-Taylor, "Large margin DAGs for multiclass classification", MIT Press, pp. 547-553, 2000.

[30] H. Nemmour and Y. Chibani, "N-SVM combination and tangent vectors for handwritten alphanumeric character recognition", 5th International Symposium on I/V Communications and Mobile Network (ISVC), pp. 1-4, 2012.

[31] Y. Hu, F. Zhu, and Xianda Zhang, "A novel approach for license plate recognition using subspace projection and probabilistic neural network", Advances in Neural Networks, Lecture Notes in Computer Science, Vol. 3497, pp. 216-221, 2005.