Journal of Academic and Applied Studies
Vol. 2(4) April 2012, pp. 1-11
Available online @ www.academians.org
ISSN1925-931X

License Plate Recognition System for Persian Vehicles

Mohammad Kadkhoda[1], Ebrahim Zarei Zefreh[2]

[1] Mathematics & Informatics Group, Academic Center (ACECR), Tarbiat Modares University, Tehran, Iran
[2] Faculties of Mathematics Science, Tarbiat Modares University, Tehran, Iran

ABSTRACT
The license plates of vehicles are the most important information items for recognition vehicles. This paper proposes an intelligence license plate recognition (ILPR) system. We deal with problematic from field of image processing in construction of intelligence license plate recognition system for Persian vehicles (ILPRSPV). We used Image Processing and back propagation neural network for recognition the numbers and characters.The proposed method tested on 500 vehicles images which captured from different distance, different angle and variable illumination conditions. The rate of success plate location recognition is about 87%, success numbers and the rate of success character segregating is about 95% and the rate of success numbers and character recognition of complete registration is about 91%.

Key words: License plate recognition (LPR), Neural network, Image processing.

INTRODUCTION

Intelligent license plates reading (ILPR) systems have been developed as a major tool for ticket of vehicles at car parking facilities, tracking vehicles during traffic law enforcement, security control of restricted areas and other related applications [1,2]. These systems attempt to facilitate the problem of identification of cars, via various techniques. Image processing is one of this it, which deal with images or video sequences taken from vehicles. License plate number is a unique property that can take into account for identifying all vehicles. The latest number plate formats are seen on Iranian lanes is similar below:

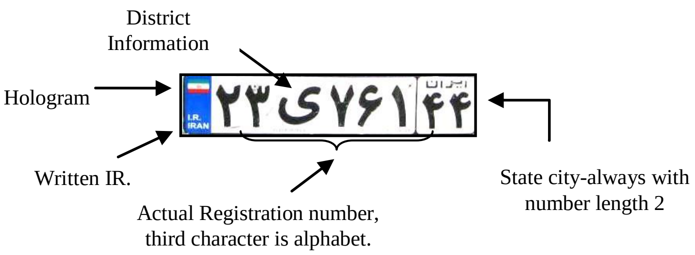

Fig 1: Persian Vehicles

An ILPR system can works online or offline; offline ILPR system, captures the vehicle images and stores them in a centralize data server for further processing, i.e. for interpretation of vehicle license plates, In contrast, an online ILPR system, the localization

Journal of Academic and Applied Studies
Vol. 2(4) April 2012, pp. 1-11
Available online @ www.academians.org **ISSN**1925-931X

and interpretation of license plates take place instantaneously from the incoming still frames, enabling real-time tracking of moving vehicles through the surveillance camera.

Although for human is read car license plate easy way with a simple observation, but the reading for a system is hard. An ILPR system mainly consists of three major parts; license plate detection (LPD), character segmentation and character recognition [3]. Due to diversity of parameters involved in car images, the first step, i.e. license plate detection is the most crucial task among these steps. We review the major works in this area which reported in image processing literature. Then our proposed method for detecting the Persian license plates is introduced with these details.

Our works is mainly related with third parts, but we have done all steps. For extracting the area we done; get the plat images that are taken by digital camera (a), then convert color image to gray image (b). we should derivate from output of stage b and choose Sobel operation in there, because it has less complexity and high speed compared to other methods, next grow output of previous stage c by dilation operation. This operation product filled objects that we use to recognize license plates location (d). In this stage, we number these objects and selected one with distinct conditions (e), the coordinate and dimension of specific object is given, we can crop it from image and its equivalent images. This object crop is obtained from gray images. Now we find the angle of this object with horizontal pivot and it rotates in opposite direction with same angle and converts to binary image. By extracting the numbers and character from license plate we can use of recognition the numbers and characters with artificial neural network. We present a back propagation neural network for detection the kind of symbol in every segment of Persian license plate.

In section 2 described basic point for image processing of license plate location: image segmentation, mathematics morphology and image coding. In section 3, described artificial neural network (ANN), the back propagation (BP) Algorithm, forward pass process in BP algorithm and error propagation in BP algorithm. We done system design in section 4 and show how its work do and give the discussion and conclusions.

IMAGE PROCCESSING

1 Thresholding

Thresholding method is used to separate object and background, which is divided image into two modes [3]. The way to resolve both categories is by assigning a thresholding value T. Each point (x, y) which have value $f(x, y) > T$ is called point object, and each point (x, y) which have value $f(x, y) \leq T$ is called background object. A thresholded image g(x, y) is defined as:

$$g(x, y) = \begin{cases} \text{object} & \text{if } g(x, y) > T \\ \text{background} & \text{if } g(x, y) \leq T \end{cases}$$

T is a constant and is called global thresholding. Thresholding is a technique widely used in image segmentation [4].

Journal of Academic and Applied Studies
Vol. 2(4) April 2012, pp. 1-11
Available online @ www.academians.org
ISSN1925-931X

2 Image Segmentation

Thresholding is the main process towards image segmentation. The thresholded image has only the characters that belong to the license plates of a vehicle including numbers and alphabetic chararecters [3]. To recognize these characters, an ANN is designed to give an answer for only one character. So, it is important to segment all characters in a vehicle license plate image. The segmentation process is implemented in prepared software using the information that the characters are different black pixels, and the character number is known for a country [5].

3 Mathematics Morphology

Mathematical morphology (MM) is a theory and technique for the analysis and processing of geometrical structures, based on set theory, lattice theory, topology, and random functions. Mathematical morphology is a tool for extracting image components that are useful in the representation and description of region shape, such as boundaries, skeletons, and the convex [3]. The primary morphological operations are dilation and erosion. More complicated morphological operators can be designed by means of combining erosions and dilations [5, 6].

Dilation is an operation that "grows" or "thickens" objects in a binary image. The specific manner and extent of this thickening is controlled by a shape referred to as a Structuring Element (SE). Computationally, structuring elements typically are represented by a matrix of 0's and l's.

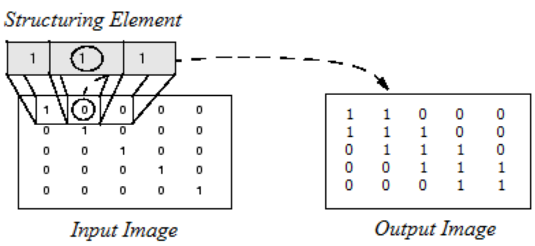

Fig 2: Morphological Dilation of a Binary Image.

Morphological dilation function sets the value of the output pixel to 1 if one of the elements in the neighborhood defined by the structuring element is one.

Erosion "shrinks" or "thins" objects in binary image. As dilation, the manner and extent of shrinking is controlled by a structuring element.

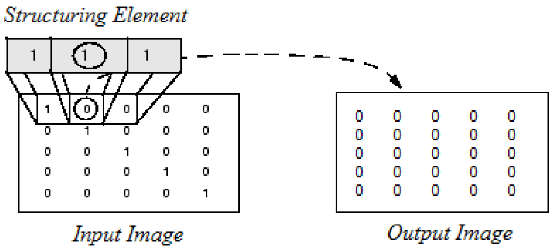

Fig 3: Morphological Erosion of a Binary Image.

Morphological erosion function sets the value of the output pixel to 1 if all of the elements in the neighborhood defined by the structuring element is one.

Journal of Academic and Applied Studies
Vol. 2(4) April 2012, pp. 1-11
Available online @ www.academians.org **ISSN**1925-931X

Erosion and dilation themselves have very few practical uses. Usually, they are combined into another two operations [4]:

a. Opening is the combination of erosion-dilation with the same SE. This operation can erase white holes on dark objects.
b. Closing is the combination of dilation-erosion with the same SE. This operation can remove black holes on white objects.

4 Image Coding

In an image processing application, it is necessary to code the images according to aim. There are many coding techniques being used in coding, such as chain code, run code and crack code [7]. In addition to these, sometimes some feature extraction techniques are used, such as moment invariants or Fourier Descriptors. According to the simplicity of the process, it is possible to code images in binary [1]. In this study, the characters are coded in binary to make them ready as an input to ANN. Two samples binary coded character image is given below.

Fig 4: Sample images code for number 2 and letter ی (input of ANN).

As seen in Fig 5, each character is coded 30*60 pixels. These 1200 bits are given as input vector set of ANN sequentially for each character. The binary technique is used to code the output of ANN expressing each character. It is a known fact that binary codes are strong through the noise.

25 digits

الف	1	0	0	0				0	0	0	0
ب	0	1	0	0				0	0	0	0
ت	0	0	1	0				0	0	0	0
۸	0	0	0	0				0	0	1	0
۹	0	0	0	0				0	0	0	1

Fig 5: The Representation Of Characters In Binary (Output Of ANN)

ARTIFICIAL NEURAL NETWORK

An Artificial Neural Network (ANN) usually is a mathematical or computational model that tries to simulate the structure and/or functional aspects of biological neural networks [7]. It consists of an interconnected group of artificial neurons and processes information using a connectionist approach to computation. ANN has some components:

1. Neuron is a unit consists of input, weight, bias, adder and transfer function.
2. Weight defines the position of hyper plane in input space.

Journal of Academic and Applied Studies
Vol. 2(4) April 2012, pp. 1-11
Available online @ www.academians.org
ISSN1925-931X

3. Bias is used to change the position of hyperplane in order not to across the zero hyperspace.
4. Transfer function gives a thresholding value to activate neuron. Some of transfer functions are illustrated on Fig 6.

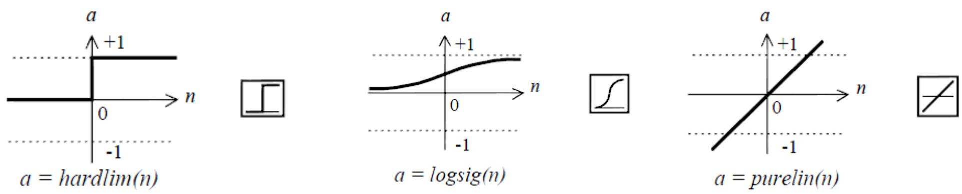

Fig 6: Some graphics of transfer function

The process of tuning the weights to the correct values –training- is vehiclеried out by passing a set of examples of input-output pairs through the model and adjusting the weights in order to minimize the error between the answer the network gives and the desired output. Once the weights have been set, the model is able to produce answers for input values which were not included in the training data. The models do not refer to the training data after they have been trained; in this sense they are a functional summary of the training data [9, 8]. These mentioned processes for back propagation algorithm are given step by step below.

5 The He Back Propagation (BP) Algorithm

The back propagation is a widely used algorithm, and it can map non-linear processes. It is a feed forward network with the one or more hidden layers. The elementary architecture of the back propagation network has three layers. There are no constraints about the number of hidden layers. Back propagation is a systematic method for training multilayer artificial neural networks. It has a mathematical foundation that is strong if not highly practical.

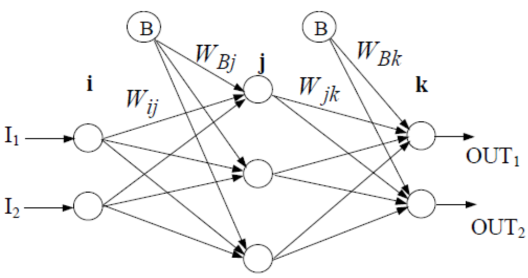

Fig 7: A sample multi-layer feedforward net structure

6 Forward Pass Process In BP Algorithm

NET values can be obtained by multiplying inputs and related weights. To calculate NET values out of perceptrons in hidden layer; the formula (1 and 2) given below is used.

$$NET = \Sigma\, I_i\, W_{ij} + W_{Bj}^{NEW} \quad (1)$$
$$OUT = 1/(1+e^{-NETj}) \quad (2)$$

Journal of Academic and Applied Studies
Vol. 2(4) April 2012, pp. 1-11
Available online @ www.academians.org
ISSN1925-931X

To calculate the OUT values, NET values are applied to an activation function. Similarly, the outs of hidden units are considered as inputs of the next hidden layer units or output layer units. W_{Bj}^{NEW} and W_{Bk}^{NEW} values present biases. The OUT and NET are calculated by using formulas (3) and (4).

$$NET = \Sigma\, I_k W_{jk} + W_{Bk}^{NEW} \quad (3)$$
$$OUT = 1/(1+e^{-NETk}) \quad (4)$$

7 Error Propagation In BP Algorithm

The outputs of neural network model are obtained from the output layer units. The difference between target values and actual values are considered as system error. The obtained error values are propagated back to connection weights. This process is applied using the following equations. Firstly, from output layer to last hidden layer:

$$\delta_k = f'(NET_k)(TARGET_k - OUT_k) \quad (5)$$
$$\delta_k = OUT_k\,(1 - OUT_k)\,(TARGET_k - OUT_k) \quad (6)$$
$$\Delta W_{kj}(n+1) = \eta \delta_k OUT_k + \alpha.[\Delta W_{kj}(n)] \quad (7)$$
$$W_{kj}^{NEW} = W_{kj}^{OLD} + \Delta W_{kj}(n+1) \quad (8)$$

Where $TARGET_k$ presents desired output value. The η is learning rate, α is momentum coefficient, $f(NET_k)$ presents activation function, N presents iteration number and ΔW is change of related weight. This term is added to old weight of the related connection to obtain a new one. Secondly, from hidden layer to input layer:

$$\delta_j = f'(NET_j).\ \Sigma \delta_k W_{kj} \quad (10)$$
$$\delta_j = OUT_j\,(1 - OUT_j) - \Sigma \delta_k W_k \quad (11)$$
$$\Delta W_{ji}(n+1) = \eta.\delta_j .OUT_j + \alpha.[\Delta W_{ji}(n)] \quad (12)$$
$$W_{ji}^{NEW} = W_{ji}^{OLD} + \Delta W_{ji}(n+1) \quad (13)$$

The bias effect the activation function in order to force the learning process, therefore the speed of learning process increases. Biases are recomputed as following, for the biases of output layer, where the letters and symbols have similar meanings;

$$\Delta W_{BK}(n+1) = \eta \delta\, k + \alpha\, [\Delta W_{BK}(n)] \quad (14)$$
$$W_{BJ}^{NEW} = W_{BJ}^{OLD} + \Delta W_{BK}(n+1) \quad (15)$$

and for the biases of hidden layer:

$$\Delta W_{Bj}(n+1) = \eta.\delta\, j + \alpha.[\Delta W_{Bj}\,.(n)] \quad (16)$$
$$W_{BJ}^{NEW} = W_{BJ}^{OLD} + \Delta W_{BK}(n+1) \quad (17)$$

The training of the neural network model, as may be understood from the previous process is vehicleried out in two steps. The first step is called forward pass that is composed of calculation for NET and OUT values. The second step is called backward pass that is composed of error propagation throughout connection weights. The iterative process is repeated until a satisfactory learning level is achieved i.e. the differences between TARGET and OUT are minimized [9, 8].

SYSTEM DESIGN

The purpose of this paper was to build a system which recognizes license plates from cars at a gate, at the entrance of a parking area. The system is designed by Image Processing and Neural Network toolbox of the MATLAB.
The License Plate Recognition system divided in to three following parts [10]:

1- Extracting the area consist of license plates from Image.
2- Extracting the numbers and character from area consist of license plates.
3- Recognition of the numbers and character with neural network.

The following diagram (Fig 8) shows the purpose algorithm in more detail:
Now, we describe the three main part of system.

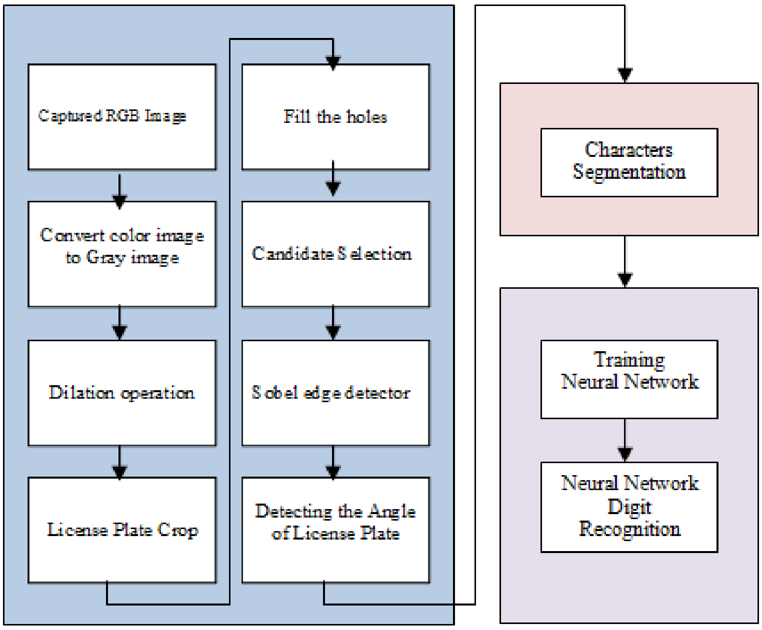

Fig 8: Diagram of LPR Recognition

8 Extracting the Area Consist Of License Plates from Image

a. Image Acquisition

The plat images are taken by digital camera with considering the following assumptions (Fig 9 No. 1):

- The image size is taken to be 1280 *1024 pixels.
- The position of plate in image is centered.
- Image is taken 2.5-5 meters.
- Lighting is given proportionally.

b. Sobel edge Detector

Journal of Academic and Applied Studies
Vol. 2(4) April 2012, pp. 1-11
Available online @ www.academians.org **ISSN**1925-931X

To reach the object outside (object around), we should derivative from output of stage b. We choose Sobel operation because it has less complexity and high speed compared to other methods (Fig 9 No. 3).

c. Mathematic Morphology

In this stage, we grow objects in output of previous stage c by dilation operation (Fig 9 No. 4). Then we fill these holes by opening operation (Fig 9 No. 5). This operation product filled objects that we use to recognize license plates location.

d. Candidate Selection

In this stage, we number these objects and select one with following conditions (Fig 9 No. 5):

1- The form of object is rectangular such that its length is three time its width.
2- The proportion of white pixels to black pixels taken to be bigger than 82 percent.

Notice, if there are several objects with the above given in e conditions then we select an object with more space and if there is no object then we select the biggest object.

e. License Plates Crop

Given the coordinate and dimension of specific object, we can crop it from image and its equivalent image. This object crop is obtained from gray image. Now we find the angle of this object with horizontal pivot and it rotates in opposite direction with same angle and converts to binary image (Fig 9 No. 6).

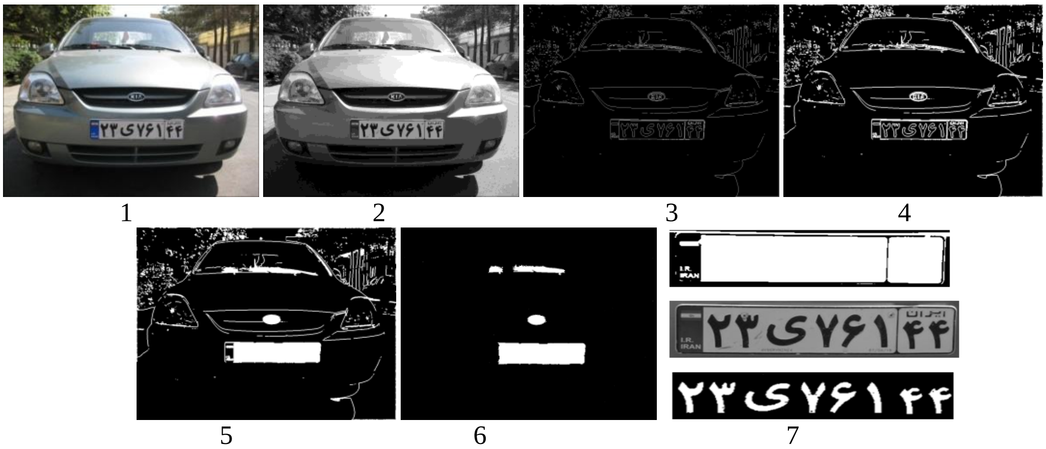

Fig 9: Extracting the area consist of license plates from Image

9 Extracting The Numbers And Character From License Plate

In order to separate numbers of binary image of previous stage, we add number of 1's per image columns and plot it. Now in order to recognize numbers, we should move left to right on the chart. When we receive the value that is equal my initial starting value, then these two points are start and end of first number. We do these operations eight times to recognize eight numbers.

Journal of Academic and Applied Studies
Vol. 2(4) April 2012, pp. 1-11
Available online @ www.academians.org **ISSN**1925-931X

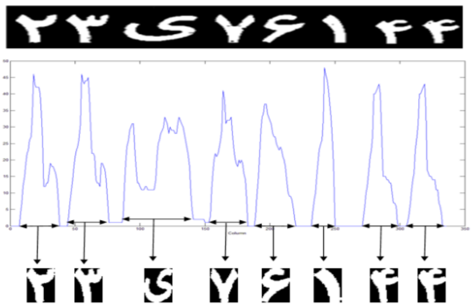

Fig 10: Extracting the numbers and character from license plate

10 Recognition The Numbers And Characters With Neural Network

a. Training Neural Network

In order to train the Neural Network, we use 330 images (thirty images for numbers and characters). These images are without noise and collection by other algorithm from license plate images.

Fig 11: Thirty images for number two and character ی

b. Neural Network Digit Recognition

In this stage, we give segregated binary images of stage 2 to Neural Network inputs. Neural Network compares these images with training images and selects the desired target output.

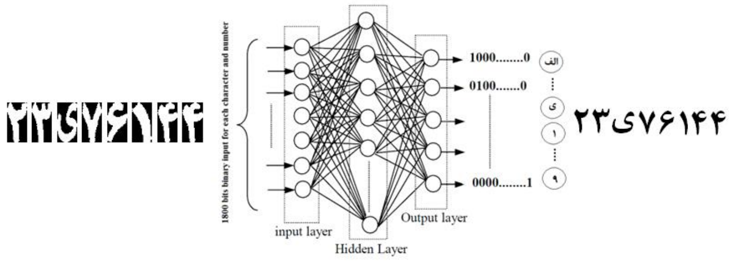

Fig 12: Neural Network Digit Recognition

CONCLUSION

We have discussed the challenges of license plate recognition in the Iranian vehicles. And presented an overview of a system designed to meet those challenges. The test where made on images taken from various sources like internet is done in MATLAB toolbox image processing.

The proposed method tested on 500 vehicles images which captured from different distance, different angle and variable illumination conditions. The following table shows performances for five samples of hundred license plates. The last row shows calculations for average performances: successful plate location about 87%, successful characters segmentation about 95% and successful recognition of complete registration plates about 91%.

Future work will have refining the design and the algorithms to provide commercially useful license plate recognition in the Iranian lane. Context results can improve efficiency algorithm for more picture.

Table1. Performances for five samples

Sample Number	success location	segmentation	success recognition
100	90%	97%	94%
100	83%	94%	88%
100	91%	96%	91%
100	85%	93%	90%
100	86%	95%	92%
average	87%	95%	91%

REFERENCES

[1] G. J. Awcock, R. Thomas, Applied Image Processing, *McGraw Hill Inc.*, 1996.

[2] S. Draghici, A neural network based artificial vision system for license plate recognition, *International Journal Neural Systems*, pp. 113–126, 1997.

[3] W. Jia, H. Zhang, X. He, Region-based license plate detection, *Journal of Network and Computer Applications*, pp. 1324–1333, Elsevier 2007.

[4] R. C. Gonzales, R. E. Woods, S. L. Eddins, Digital Image Processing using MATLAB, *Prentice Hall Upper Saddle River*, 2004.

[5] Bo Li, Z. Zeng, J. Zhou, H. Dong, An Algorithm for License Plate Recognition Using Radial Basis Function Neural Network, *Proceedings of Conference ISCSCT in LiZZD*, pp. 569-572, 2008.

[6] Image Processing Toolbox For use with MATLAB 16.

[7] Y. Huang, C. Chen, Y. Chang, F. Sandnes, An intelligent strategy for checking the annual inspection status of motorcycles based on license plate recognition, *Expert Systems with Applications*, pp. 9260–9267, Elsevier 2009.

[8] N. H. Sheikh Abdullah, M. Khalid and R. Yusof, License Plate Recognition using Multi-cluster and Multilayer Neural Networks, *Proceeding of SITI Conference* IEEE, pp. 1818-1823, 2006.

[9] V. Koval, V. Turchenko, V. Kochan, A. Sachenko, G. Markowsky, Smart License Plate Recognition System Based on Image Processing Using Neural Network, *Second IEEE International Workshop on Intelligent Data Acquisition and Advanced Computing Systems Technology and Applications*, pp. 123-127, 2003.

[10] K. Kim, J. Kim, H. Kim, License-Based Approach for License Plate Recognition, *IEEE Transactions on Intelligent Transportation Systems*, pp. 614-623, 2000.