# TWO NEURAL NETWORKS FOR LICENSE NUMBER PLATES RECOGNITION

**[1]A. AKOUM, [1]B. DAYA , [2]P. CHAUVET**

[1]Lebanese University, Institute of Technology, P.O.B. 813 - Saida - LEBANON

[2]Institute of Applied Mathematics, UCO, BP10808 – 49008 Angers - FRANCE

**ABSTRACT**

A license plate recognition system is an automatic system that is able to recognize a license plate number, extracted from an image device. Such system is useful in many fields and places: parking lots, private and public entrances, border control, theft and vandalism control. In our paper we designed such a system. First we separated each digit from the license plate using image processing tools. Then we built a classifier, using a training set based on digits extracted from approximately 350 license plates. Once a license plate is detected, its digits are recognized, displayed on the User Interface or checked against a database. The focus is on the design of algorithms used for extracting the license plate from an image of the vehicle, isolating the characters of the plate and identifying characters. Our approach is considered to identify vehicle through recognizing of its license plate using two different types of neural networks: Hopfield and the multi layer perceptron "MLP". A comparative result has shown the ability to recognize the license plate successfully. The experimental results have shown the ability of Hopfield Network to recognize correctly characters on license plate more than MLP architecture which has a weaker performance. A negative point in the case of Hopfield is the processing time.

**Keywords:** *License Number Identification, Image Processing, License Plate Locating, Segmentation, Feature Extraction, Character Recognition, Artificial Neural Network.*

## 1. INTRODUCTION

License plate recognition applies image processing and character recognition technology to identify vehicles by automatically reading their license plates Optical character recognition has always been investigated during the recent years, within the context of pattern recognition [1], [2]. The broad interest lies mostly in the diversity and multitude of the problems that may be solved (for different language sets), and also to the ability to integrate advanced machine intelligence techniques for that purpose; thus, a number of applications has appeared [3], [4].

The steps involved in recognition of the license plate are acquisition, candidate region extraction, segmentation, and recognition. There is batch of literature in this area. Some of the related work is as follows: [3] has developed sensing system, which uses two CCDs (Charge Coupled Devices) and a prism to capture the image. [8] has proposed method heart extracting characters without prior knowledge of their position and size. [7] has discussed the recognition of individual Arabic and Latin characters. Their approach identifies the characters based on the number of black pixel rows and columns of the character and comparison of those values to a set of templates or signatures in the database. [10], [3] have used template matching. In the proposed system high resolution digital camera is used for heart acquisition.

The intelligent visual systems are requested more and more in applications to industrial and deprived calling: biometrics, ordering of robots [12], substitution of a handicap, plays virtual, they make use of the last scientific projections in vision by computer [13], artificial training [14] and pattern recognition [15].

The present work examines the recognition and identification -in digital images- of the alphanumeric content in car license plates. The images are obtained from a base of abundant data, where variations of the light intensity are common and small translations and or rotations are permitted. Our approach is considered to identify vehicle through recognizing of it license plate using, two processes: one to extract the block of license plate from the initial image containing the vehicle, and the second to extract characters from the licence plate image. The last step is to recognize licence plate characters and identify the vehicle. For this, and on the first level, we use the Hopfield networks with 42x24 neurons as the dimension of each character. The network must memorize all the Training Data (36 characters). For the validation of the network, we have built a program that can read the sequence of characters, split each character, re-size it and finally display the result on a Notepad editor.

A comparison with another type of neural networks, the multi layer perceptron "MLP" is much appreciated to evaluate the performance of each network.

The rest of the paper is organized as follows: In Section 2, we present the real dataset used in our experiment. We give in section 3 the description of our algorithm which extracts the characters from the license plate. Section 4 gives the experimental results the recognizing of characters using two types of neural networks architecture. Section 5 contains our conclusion.

## 2. DATABASES

The database (350 Images with license plates) contains images of good quality (high-resolution: 1280x960 pixels resizes to 120x180 pixels) of vehicles seen of face, more or less near, parked either in the street or in a car park, with a negligible slope.

Let us note that in our system we will divide in a random way the unit of our database into two:

1) A base of Training on which we regulate all the parameters and thresholds necessary to the system so as to obtain the best results.
2) A base T by which we will test all our programs.

The images employed have characteristics which limit the use of certain methods. In addition, the images are in level of gray, which eliminates the methods using color spaces.


*Figure 1: Some examples from the database training.*

## 3. LICENSE PLATE CHARACTERS EXTRACTING

Our algorithm is based on the fact where an area of text is characterized by its strong variation between the levels of gray and this is due to the passage from the text to the background and vice versa (see fig. 1.). Thus by locating all the segments marked by this strong variation and while keeping those that are cut by the axis of symmetry of the vehicle found in the preceding stage, and by gathering them, one obtains blocks to which we consider certain constraints (surface, width, height, the width ratio/height,…) in order to recover the areas of text candidates i.e. the areas which can be the number plate of the vehicle in the image.


*Figure 2: Selecting of license plate.*

*Figure 3: Extracting of license plate.*

We digitize each block then we calculate the relationship between the number of white pixels and that of the black pixels (minimum/maximum). This report/ratio corresponds to the proportion of the text on background which must be higher than 0.15 (the text occupies more than 15% of the block).

First, the block of the plate detected in gray will be converted into binary code, and we construct a matrix with the same size block detected. Then we make a histogram that shows the variations of black and white characters.

To filter the noise, we proceed as follows: we calculate the sum of the matrix column by column, then we calculate the min_sumbc and max_sumbc representing the minimum and the maximum of the black and white variations detected in the plaque.

All variations which are less than 0.08 * max_sumbc will be considered as noises. These will be canceled facilitating the cutting of characters.

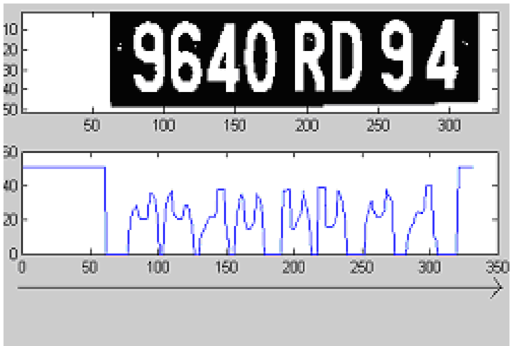



*Figure 4: Histogram to see the variation black and white of the characters.*

To define each character, we detect areas with minimum variation (equal to min_sumbc). The first detection of a greater variation of the minimum value will indicate the beginning of one character. And when we find again another minimum of variation, this indicates the end of the character. So, we construct a matrix for each character detected.

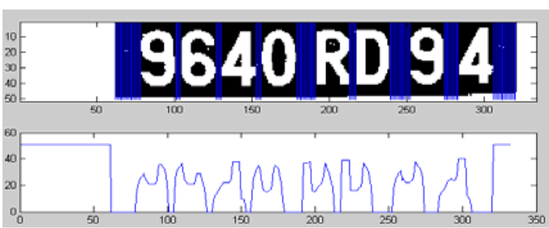



*Figure 5: The characters are separated by several vertical lines by detecting the columns completely black.*

The Headers of the detected characters are considered as noise and must be cut. Thus, we make a 90 degree rotation for each character and then perform the same work as before to remove these white areas.

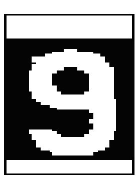


*Figure 6: Extraction of one character*

.

A second filter can be done at this stage to eliminate the small blocks through a process similar to that of extraction by variations black white column.

*Figure 7: Rotation 90 degrees of the character.*

Finally, we make the rotation 3 times for each image to return to its normal state. Then, we convert the text in black and change the dimensions of each extracted character to adapt it to our system of recognition (neural network type Hopfield-type and MLP).

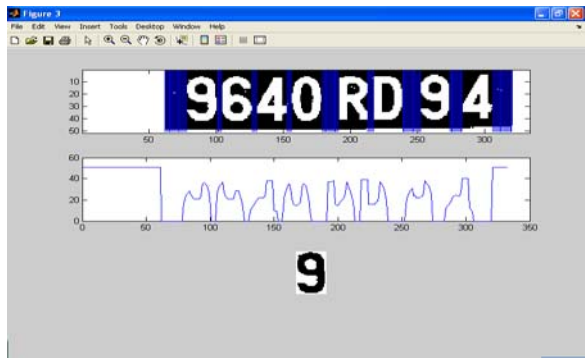



*Figure 8: Representation of the histogram of the alphanumeric chain and extraction of a character from the number plate.*

## 4. RECOGNIZING OF CHARACTERS USING OUR APPROACH NEURAL NETWORKS

The character sequence of license plate uniquely identifies the vehicle. It is proposed to use artificial neural networks for recognizing of license plate characters, taking into account their properties to be as an associative memory. Using neural network has advantage from existing correlation and statistics template techniques [5] that allow being stable to noises and some position modifications of characters on license plate.

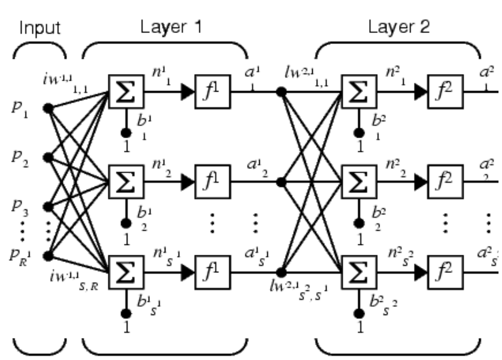



*Figure 9: Structure of MLP neural network for an image with R pixels.*

Our approach is considered to identify vehicle through recognizing its license plate using, Hopfield networks with 42x24 neurons as the dimension of each character. The network must memorize all the Training Data (36 characters). For the validation of the network we have built a program that reads the sequence of characters, to cut each character and resize it and put the result on a Notepad editor. A comparison with an MLP network is very appreciated to evaluate the performance of each network.

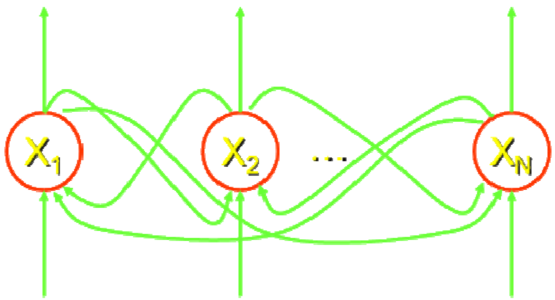



*Figure 10: Structure of Hopfield neural network for an image with N pixels.*

For this analysis a special code has been developed in MATLAB [6].

Our Software is available to do the following:

1) Load a validation pattern.

2) Choose architecture for solving the character recognition problem, among these 6 architectures:

- "HOP112": Hopfield architecture, for pictures of 14x8 pixels (forming vector of length 112).
- "HOP252": Hopfield architecture, for pictures of 21x12 pixels (forming vector of length 252).
- "HOP1008": Hopfield architecture, for pictures of 42x24 pixels (forming vector of length 1008)
- "MLP112": Multi Layer Perceptron architecture, for pictures of 14x8 pixels (forming vector of length 112).
- "MLP252": Multi Layer Perceptron architecture, for pictures of 21x12 pixels (forming vector of length 252).
- MLP1008": Multi Layer Perceptron architecture, for pictures of 42x24 pixels (forming vector of length 1008).
- 2) Calculate time of processing of validation (important for "real applications").

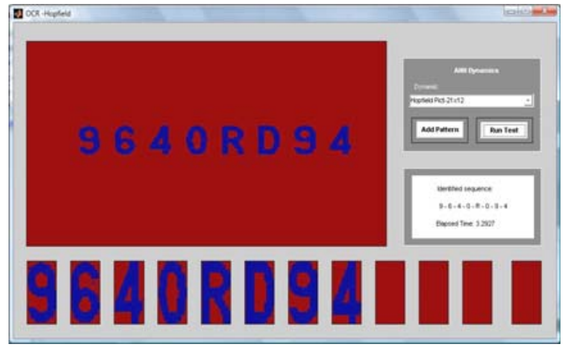


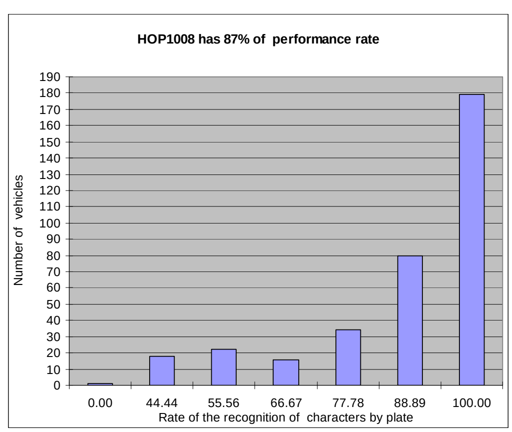

*Figure 11: The graphic interface of our program which recognizes alphanumeric containing in the plate and posts the result in text form.*

For our study, we used 3 kinds of Hopfield Networks (1008, 252 and 112 neurons) and 3 kinds of MLP Networks, always with one hidden layer (1008-252-1; 252-64-1 and 112-32-1). In the case of MLPs, we train one MLP per character; it means that there are 36 MLPs for doing the recognition. In the case of Hopfield there is only one network that memorizes all the characters.

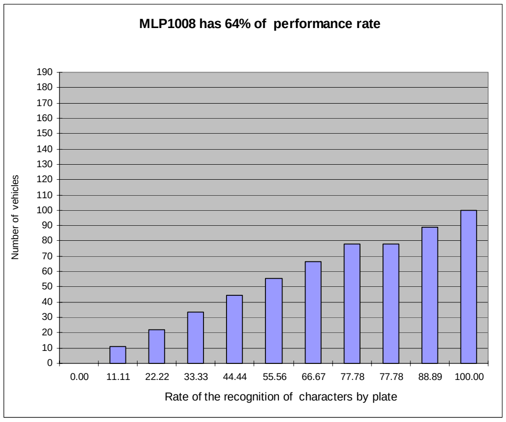
Table 1 shows the performance of each neural architecture for the six different cases.

*Table 1: The performance of each Neural Network Architecture (MultiLayer Perceptron and Hopfield).*

| Neural Network | Number of neurons | Total Symbols | Total Errors | Perf (%) |
|---|---|---|---|---|
| HOP | 1008 | 1130 | 144 | 87 % |
| MLP | 1008 | 1130 | 400 | 64 % |
| HOP | 252 | 1130 | 207 | 84% |
| MLP | 252 | 1130 | 255 | 80 % |
| HOP | 112 | 1130 | 342 | 69 % |
| MLP | 112 | 1130 | 355 | 68 % |

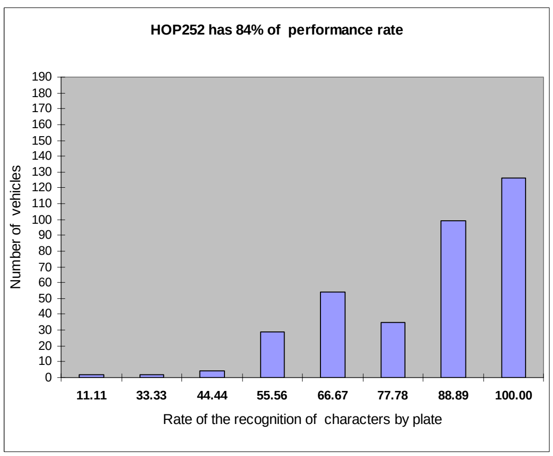



*(a)*

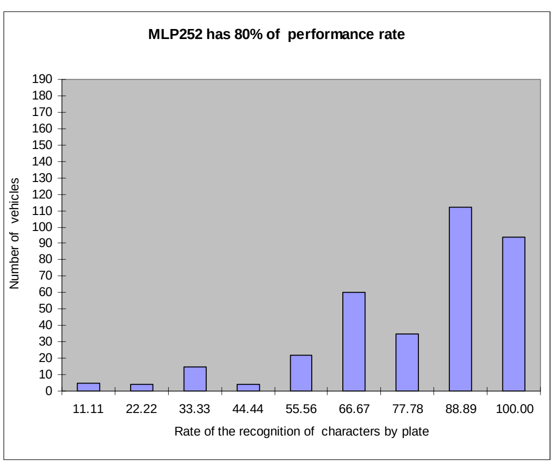



*(b)*

*Figure 12: Histogram representing the number of vehicles among 350 images in function of the recognition rates for the HOP1008 architecture (a) and the MLP1008 architecture (b).*

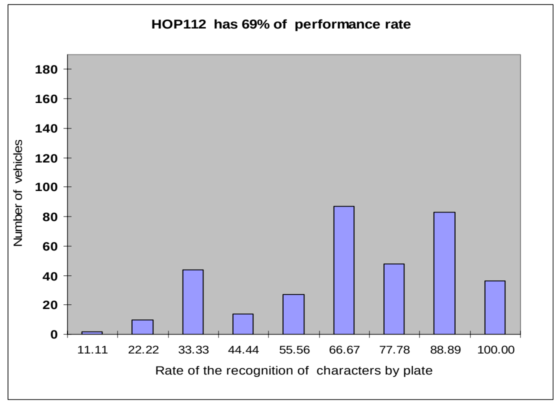



*(a)*

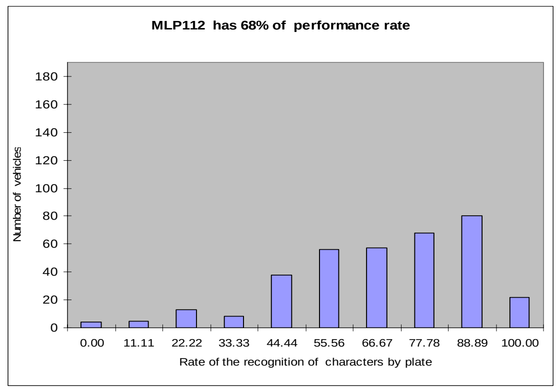



*(b)*

*Figure 13: Histogram representing the number of vehicles among 350 images in function of the recognition rates for the HOP252 architecture (a) and the MLP252 architecture (b).*

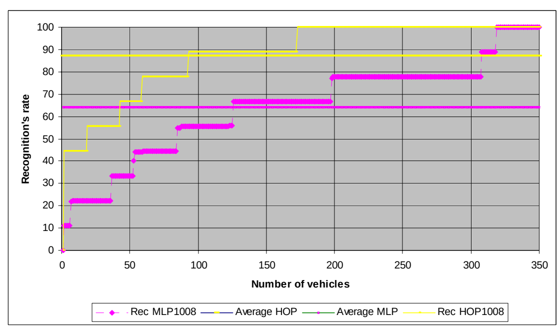



*(a)*

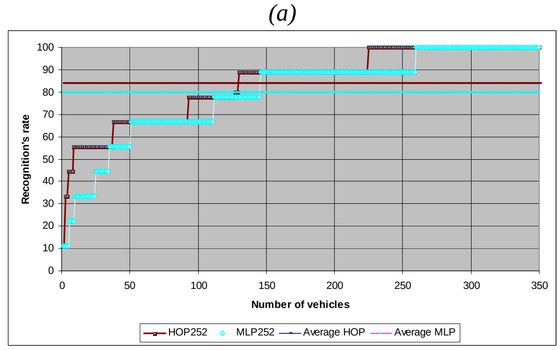



*(b)*

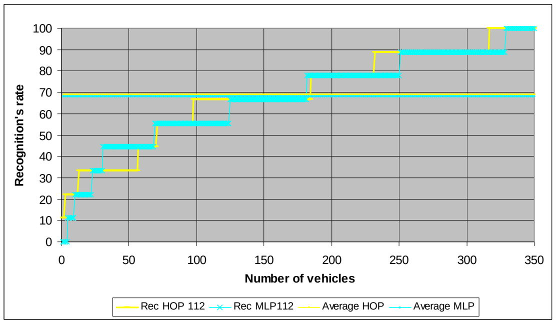
*Figure 14: Histogram representing the number of vehicles among 350 images in function of the recognition rates for the HOP112 architecture (a) and the MLP112 architecture (b).*




*(a)*




*(b)*




*(c)*

*Figure 15: Comparing the recognition rates of characters between Hopfield and MLP architectures.*

Tables 2 and 3 "see appendix" show all the recognitions for all the patterns. The first column corresponds to the file's name of the plate number; the second column, to the plate number observed with our bare eye and columns 3 to 5 represent the plate number that each architecture has recognized. The last row corresponds to the average processing time that takes for each network.

In the case of the Hopfield recognition, when the network doesn't reach a known stable state, it gives the symbol "?".

Tables 4, 5 and 6 "see appendix" specify the errors for each case (we can consider "?" and "-" has not an error), because it means that Hopfield Network can not recognize a symbol that has not been memorized.

### 4.1 RESULTS

Hopfield Networks have demonstrated better performance 87% than MLPs regarding OCR field. A negative point in the case of Hopfield is the processing time, in the case of pictures of 42x24 pixels (90 seconds average, versus only 3 seconds in the case of pictures of 21x12 pixels). It can be observed also that cases "HOP1008" and "HOP252" don't present an appreciable difference regarding performance.

A strange case corresponds to case "MLP1008", in which MLP architecture has the lower performance, and maybe decreasing the number of neuron in the hidden layer we can obtain better results, but anyway the size (bytes in disc) of each MLP is too big for considering it a good option (154 MB the 36x3 networks).

## 5. CONCLUSION

In this work, a system for recognizing the number off a license plate was designed. For this goal, we used 400 images of license plates, from which we received 3200 images of digits.

Our algorithm of license plate recognition, allows to extract the characters from the block of the plate, and then to identify them using artificial neural network. The experimental results have shown the ability of Hopfield Network to recognize correctly characters on license plate with probability of 87% more than MLP architecture which has a weaker performance of 80%.

The proposed approach of license plate recognition can be implemented by the police to detect speed violators, parking areas, highways, bridges or tunnels. Also the prototype of the system is going to be integrated and tested as part of the sensor network being developed by other intelligent systems used in our CEDRE project.

## 6. ACKNOWLEDGEMENTS

This research was supported by the CEDRE project (07SciF29/L42).

## 7. REFERENCES

[1] F. Ahmed and A.A.S. Awwal., 1993. An Adaptive Opto-electronic Neural Network for Associative Pattern Retrieval. *Journal of Parallel and Distributed Computing*, 17(3), pp. 245-250.

[2] J. Swartz, 1999. "Growing 'Magic' of Automatic Identification", *IEEE Robotics & Automation Magazine*, 6(1), pp. 20-23.

[3] Park et al, 2000. "OCR in a Hierarchical Feature Space", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(4), pp. 400-407.

[4] Omachi et al, 2000. Qualitative Adaptation of Subspace Method for Character Recognition. *Systems and Computers in Japan*, 31(12), pp. 1-10.

[5] B. Kroese, 1996. An Introduction to Neural Networks, *Amsterdam, University of Amsterdam*, 120 p.

[6] J. Stephen, 2002. Chapman. MATLAB *Programming for Engineers*, 2nd Edition, Brooks/Cole Publishing Company.

[7] J. Cowell, and F. Hussain, 2002. "A fast recognition system for isolated Arabic characters", *Proceedings Sixth International Conference on Information and Visualisation, England*, pp. 650-654.

[8] H. Hontani, and T. Kogth, 2001. "Character extraction method without prior knowledge on size and information", *Proceedings of the IEEE*

*lnternational Vehicle Electronics Conference* (IVEC'OI). pp. 67-72.

[9] C. Nieuwoudt, and R. van Heerden, 1996. "Automatic number plate segmentation and recognition", *In Seventh annual South African workshop on Pattern Recognition*, l April, pp. 88-93.

[10] M., Yu, and Y.D. Kim, 2000. "An approach to Korean license plate recognition based on vertical edge matching", *IEEE International Conference on Systems, Ma, and Cybernerics*, vol. 4, pp. 2975-2980.

[12] B. Daya & P. Chauvet, 2005. A Static Walking Robot to Implement a Neural Network System. *Journal of Systems Science*, Vol. 31, pp. 27-35.

[13] B. Daya & A. Ismail, 2006. A Neural Control System of a Two Joints Robot for Visual conferencing. *Neural Processing Letters*, Vol. 23, pp.289-303.

[14] Prevost, L. and Oudot, L. and Moises, A. and Michel-Sendis, C. and Milgram, M., 2005. *Hybrid* generative/discriminative classifier for unconstrained character recognition. *Pattern Recognition Letters, Special Issue on Artificial Neural Networks in Pattern Recognition*, pp. 1840-1848.

[15] M. Hanif, S. and Prevost, L. and Belaroussi, R. and Milgram, M, 2007. "A Neural Approach for Real Time Facial Feature Localization", *IEEE International Multi-Conference on Systems, Signals and Devices*, Hammamet, Tunisie.

## APPENDIX

*Table 2: The recognition for all the patterns with different numbers of neurons (Hopfield Network).*

| Seq | Real plate nb (eye) | HOP1008 | HOP252 | HOP112 |
|---|---|---|---|---|
| p1 | 9640RD9 | 9640R094 | 9640R094 | 9640R?94 |
| p10 | 534DDW77 | 534DD?77 | 534DD?77 | 534DD?77 |
| p11 | 326TZ94 | 326TZ94 | 326TZ94 | 325TZ9? |
| p12 | 6635YE93 | 66J5YES? | B??5YE?? | B???YE?? |
| p13 | 3503RC94 | 3503RC94 | 35O3RC94 | 3503RC94 |
| p14 | 7874VT94 | 7874VT94 | 7874VT94 | 7874VT94 |
| p15 | 3015TA61 | 3015TA61 | 3015TA61 | 3Q15TA61 |
| p16 | 655PZR75 | 655PZR75 | 665PZR75 | 655???75 |
| p17 | 1416XZ93 | A416XZ83 | ?416XZ8? | ????XZ?? |
| p18 | 957PGK75 | 957PGK75 | 9S7PGK75 | 957PG?75 |
| p19 | 75N5088F | 75N5086F | 75N5088F | 75N50??? |
| p2 | 437LPB75 | 43VLFBV5 | 4?VLPBV E | ???L?B?? |
| p20 | 3593SC94 | 3593SC94 | 3?93SC94 | 3??3??94 |
| p23 | - 347DEX92 | 034ZDEX9 ? | 934?DEX9 Z | ?S?`?DEX9 ? |
| p24 | 703PJA75 | V0JFJAV5 | V0?PJAV5 | ???FJA?5 |
| p25 | 5641SB94 | 5641SB94 | ?641SB94 | ?641SB9? |
| p26 | 5641SB94 | 5641SB94 | ?641SB94 | ??41SB94 |
| p27 | 7255VD94 | 7255V094 | 7255V094 | 7255VO9? |
| p28 | 775SH94 | 775SH94 | ?75SH94 | 775SHS? |
| Ti me | -- | 90 sec | 3 sec | 2 sec |

*Table 3: The recognition for all the patterns with different numbers of neurons (multi layer perceptron network).*

| Seq | Real plate nb (eye) | MLP1008 | MLP252 | MLP112 |
|---|---|---|---|---|
| p1 | 9640RD94 | 964CR094 | 56409D94 | 2B40PD34 |
| p10 | 534DDW77 | 53CZD677 | 53CDD877 | 53WDD977 |
| p11 | 326TZ94 | 32SSZ8C | 326T794 | 328TZ3C |
| p12 | 6635YE93 | 8695YE8O | BE3SYE98 | E535YEB3 |
| p13 | 3503RC94 | 35CO3C94 | 5503RC94 | 3503PC24 |
| p14 | 7874VT94 | F674VT94 | 7574V394 | 7S74VT34 |
| p15 | 3015TA61 | 3C15TAS1 | 5015TA51 | 3015TAS1 |
| p16 | 655PZR75 | 6556ZR75 | 65597835 | 555PZP75 |
| p17 | 1416XZ93 | BCB5XZ8O | S236XZ9J | RC15XZB3 |
| p18 | 957PGK75 | 957FGK75 | 95785K75 | N57PGK75 |
| p19 | 75N5088F | 75N5666F | 75N50557 | 75N50JSF |
| p2 | 437LPB75 | CORLZ8R5 | CJRL88RE | R3TLFBT5 |
| p20 | 3593SC94 | 8598SG90 | 3593SC94 | S5BSBQ54 |
| p23 | -347DEX92 | 6ZCXZEX9Z | 2327UFX9Z | 93CPDEX3Z |
| p24 | 703PJA75 | R0OFJ0R5 | R698JAR8 | TQ3FJAT5 |
| p25 | 5641SB94 | 5661S69C | 5541S592 | 5S41SB39 |
| p26 | 5641SB94 | B601S69C | S641S594 | 5S41SB34 |
| p27 | 7255VD94 | 7255VOS4 | 7255VD94 | 7255VS5C |
| p28 | 775SH94 | 725SHS4 | 775SH5C | YY5SH9C |
| Time | -- | 25 sec | 3 sec | 2 sec |

*Table 4: Hop1008-Errors.*

| | | | | | |
|---|---|---|---|---|---|
| 0&D | ?&W | J&3 | S&9 | ?&3 | A&1 |
| Z&7 | ?&2 | V&7 | J&3 | F&P | V&7 |
| 5&6 | 6&5 | S&9 | S&9 | H&M | 6&- |
| 8&B | ?&1 | Z&2 | ?&1 | A&4 | 0&Q |
| S&9 | H&M | 6&- | ?&2 | ?&R | ?&2 |
| V&L | L&7 | 7&5 | O&0 | ?&1 | T&7 |
| 4&- | Z&2 | 4&- | ?&1 | 5&- | N&W |
| Z&2 | ?&7 | O&0 | 4&- | V&7 | F&P |
| 8&9 | 6&8 | V&7 | F&P | V&7 | 0&- |
| 0&D | S&9 | V&7 | J&3 | F&P | V&7 |
| 0&1 | ?&7 | ?&3 | 8&B | ?&7 | A&4 |
| 8&9 | 8&9 | 6&8 | 0&Q | ?&1 | O&0 |
| Z&2 | 6&M | ?&7 | 7&3 | 3&P | P&V |
| 0&D | Z&2 | V&7 | ?&M | 0&D | 0&- |
| I&1 | ?&1 | 9&- | A&1 | J&3 | V&W |
| V&7 | J&3 | U&0 | ?&1 | ?&1 | A&4 |

*Table 5: MLP1008-Errors.*

| | | | | | |
|---|---|---|---|---|---|
| C&0 | 0&D | C&4 | Z&D | 6&W | S&6 |
| O&3 | C&0 | O&3 | 3&R | F&7 | 6&8 |
| 5&6 | 8&9 | O&3 | F&P | 6&0 | 6&8 |
| R&7 | 8&3 | 8&3 | G&C | 0&4 | 6&- |
| O&3 | F&P | 0&A | R&7 | 6&4 | 6&B |
| S&9 | 2&7 | S&9 | 6&0 | 6&8 | 6&0 |
| F&P | 0&A | R&7 | 5&6 | 6&5 | 6&4 |
| C&4 | 8&9 | C&4 | X&7 | 6&8 | 9&3 |
| B&Y | 6&- | G&6 | 9&3 | 8&P | 0&A |
| R&7 | C&4 | 6&0 | C&4 | C&4 | 6&0 |
| 8&9 | 5&3 | X&K | Z&2 | C&4 | 6&8 |
| 9&3 | C&4 | 8&Q | C&4 | 8&3 | 8&9 |
| 6&Q | S&9 | G&7 | 6&0 | G&7 | G&S |
| S&P | 5&3 | 6&M | 6&9 | C&4 | 6&- |
| 0&M | B&7 | 7&3 | 8&P | H&V | V&L |
| O&0 | 8&C | S&9 | C&4 | T&7 | O&3 |
| 8&E | G&S | R&7 | 6&8 | O&9 | B&1 |
| 6&8 | C&4 | 8&E | 8&E | 8&E | C&- |
| G&7 | 6&- | 6&0 | F&P | 0&1 | 0&- |
| 6&- | 8&3 | Z&D | O&1 | L&4 | 9&3 |
| 6&0 | 6&0 | C&4 | C&4 | L&4 | 6&- |
| G&7 | O&3 | 8&0 | 6&8 | C&4 | Z&P |
| S&T | 8&9 | C&4 | 8&6 | 9&3 | 8&9 |
| C&0 | S&6 | 6&P | B&1 | C&4 | B&1 |
| 6&8 | C&4 | O&3 | R&7 | Z&P | 8&B |
| Z&3 | C&4 | X&7 | Z&D | Z&2 | R&7 |
| C&4 | B&5 | 0&4 | 6&B | C&4 | O&D |
| C&4 | 8&9 | C&4 | X&7 | R&7 | O&3 |
| 6&B | C&4 | 2&7 | S&9 | 6&0 | 6&8 |
| 8&6 | G&7 | 6&9 | Z&B | C&4 | 6&M |
| 6&1 | R&7 | O&3 | 8&3 | 0&D | 8&B |
| 6&B | 6&9 | B&1 | R&1 | 5&S | 8&P |
| 6&0 | B&1 | 8&R | 6&R | 8&9 | C&4 |
| C&4 | G&S | 0&A | 8&9 | C&4 | 6&8 |
| C&4 | S&9 | 8&P | S&T | 8&9 | 9&3 |
| Z&2 | Z&D | 6&M | H&R | Z&2 | Z&2 |
| L&7 | 7&5 | 9&3 | B&1 | O&3 | 6&9 |
| 6&0 | 5&S | 0&D | Z&2 | C&4 | C&4 |
| G&7 | C&4 | 0&M | 8&P | 0&D | 6&- |
| B&1 | B&1 | 6&0 | Z&2 | X&K | 0&D |
| C&4 | C&4 | 6&W | C&4 | T&1 | 0&1 |
| Z&2 | 6&8 | B&Y | G&7 | 6&8 | 5&3 |
| C&4 | O&3 | R&7 | Z&P | 8&B | R&7 |
| S&5 | O&7 | 8&C | B&1 | C&4 | 5& |

*Table 6: Hop252-Errors.*

| | | | | | |
|---|---|---|---|---|---|
| 0&D | ?&W | B&6 | ?&6 | ?&3 | ?&9 |
| S&5 | ?&3 | V&7 | V&7 | E&5 | ?&5 |
| ?&5 | ?&5 | 0&D | ?&7 | O&0 | 8&9 |
| ?&7 | O&0 | 8&9 | A&4 | B&6 | S&9 |
| 9&1 | V&7 | ?&3 | ?&3 | 8&B | V&7 |
| ?&R | S&9 | ?&4 | U&Q | 8&9 | 8&9 |
| ?&8 | J&3 | H&M | 8&9 | ?&4 | ?&- |
| P&V | V&L | L&7 | 7&5 | O&0 | S&5 |
| ?&S | 0&D | V&7 | E&5 | ?&7 | N&M |
| ?&7 | ?&- | ?&1 | 5&- | N&W | I&1 |
| ?&W | Z&2 | V&7 | J&3 | ?&8 | O&0 |
| ?&3 | O&0 | 6&5 | ?&1 | 8&9 | ?&3 |
| 9&- | ?&7 | Z&2 | V&7 | ?&3 | V&7 |
| A&4 | V&7 | ?&3 | V&7 | ?&6 | 6&5 |
| ?&B | H&M | 8&- | O&3 | ?&P | S&5 |
| A&4 | O&0 | A&4 | ?&1 | J&3 | ?&1 |
| ?&4 | 6&8 | 0&Q | ?&1 | ?&7 | O&0 |
| ?&7 | H&R | ?&M | ?&7 | 7&3 | 3&P |
| ?&7 | ?&7 | ?&7 | ?&1 | ?&7 | ?&7 |
| 0&D | W&- | A&4 | ?&E | 4&- | ?&2 |
| ?&1 | ?&- | ?&D | ?&L | ?&1 | A&4 |
| ?&4 | ?&- | ?&3 | V&7 | V&7 | E&5 |