[Downloaded from www.aece.ro on Thursday, March 28, 2024 at 17:24:03 (UTC) by 34.238.138.162. Redistribution subject to AECE license or copyright.]

An Effect of Noise in Printed Character Recognition System Using Neural Network

Sandel GHEORGHITA[1], Radu MUNTEANU[1], Adrian GRAUR[2]
[1]*Faculty of Electrical Engineering, Technical University of Cluj-Napoca, 400020, Romania*
[2]*Stefan cel Mare University of Suceava, 720229, Romania*
sandugeorge@yahoo.com;radu.munteanu@mas.utcluj.ro;adriang@eed.usv.ro

***Abstract*—In this article we present the implementation of a neural network model trained with a high noise level using a backpropagation algorithm and the experimental results for printed character recognition, based on the idea of using the primary information by reorganising it in a different format. The values obtained at the outputs of each network are processed by using analysis algorithms designed for this purpose. The suggested model is made up of two neural networks and two analysis modules. In M1 Module we designed a value analysis algorithm for all the outputs of the two neural networks in order to select the best values provided by the networks. The M2 Module also contains a designed algorithm, which assesses the data based on the fact that the highest values are directly correlated with the probability of correctly identifying the characters entered into the networks.**

Results are obtained for noise of up to 50% applied to the input data. The values obtained at the outputs of the two modules emphasises the increase of the printed character recognition level up to 89.1% for the M1 module and up to 89.8% for the M2 module, the number of errors decreasing vis-à-vis the RNA2 network response from 12.5% to 10.9%, and 10.2%, respectively. In order to set up the hidden layer of 90 neurons, a value of 92% was obtained at the output of the M2 analysis module.

The performed model increased the printed character recognition rate by using the same primary information in a different manner. The validity and functionality of the suggested model are confirmed by experimental results.

***Index Terms*—backpropagation, character recognition, neural networks, noise perturbation, training algorithm.**

I. INTRODUCTION

The article presents an architecture based on neural networks and analysis blocks designed to improve the performance of a character recognition system.

The neural networks are parallel structures made up of interconnected neurons. These networks have an increased performance and a fast response in data processing. For character recognition, Martin and Pittman used networks trained with the backpropagation algorithm [1]. The same algorithm was also suggested by Fukumi and others in [2]. Very good results for the recognition of graphical or handwritten shapes can also be achieved with the help of fuzzy neural network architectures that use neurons and fuzzy logic and with convolutional neural network (CNN) [3].

Applying a controlled amount of noise during training may improve convergence and generalization performance. State of the art outliers (data samples with a high noise level) scores are not standardized and often difficult to interpret [4-5].

II. NEURAL NETWORK ARCHITECTURE

The neural networks are architectures that use several layers with interconnected neurons. The standard neuron has N weighted inputs and one output. The result of adding up all x_i inputs and the w_{ij} ratios are transferred at output through a *f*() non-linear activation function described by the equation (1):

$$y_i = f\left[\sum_{i=1}^{N} w_{ij} x_i - T\right] \tag{1}$$

where i=1…N represents the N inputs and T represents the neuron's inner limit.

In general, there is an input layer, a hidden layer and an output layer that use activation functions with the purpose of limiting the variation area of its output at a preset area (limit, sigmoid or Gauss function).

Because sigmoid functions have the non-linearity and differentiality advantage, for the neurons of the hidden layer the *logsig* non-linear activation function described in the equation (2) is used. For the output layer, the same equation is used because this offers values in 0 - 1 area [6-7].

$$\varphi(v) = \frac{1}{1 + e^{(-\gamma v)}} \tag{2}$$

Since the training of a network is in general slow and it requires thousands or tens of thousands of ages, learning acceleration methods are used. The most common methods are the momentum and the application of a variable learning rate described in equations (3) and (4):

$$\Delta w_{ij}(p) = \Delta w_{ij}(p) + \alpha \cdot \Delta w_{ij}(p-1) \tag{3}$$

$$\eta_{ij}(p) = \begin{cases} u \cdot \eta_{ij}(p-1), \operatorname{sgn}(\Delta w_{ij}(p)) = \operatorname{sgn}(\Delta w_{ij}(p-1)) \\ d \cdot \eta_{ij}(p-1), \operatorname{sgn}(\Delta w_{ij}(p)) = -\operatorname{sgn}(\Delta w_{ij}(p-1)) \end{cases} \tag{4}$$

where Δw_{ij} *(p)* represents the gradient of the weights and $\eta_{ij}(p)$ represents the learning rate [10-11].

For the character recognition problem, the backpropagation algorithm with the variable learning rate method is used. The method consists of using a learning rate for each weight, and the parameters are adapted at each iteration depending on the consecutive signals of the gradients. The learning process stops when the established error level is reached .

III. SUGGESTED MODEL

Two neural networks have been designed, marked with RNA1 and RNA2. The network's architecture contains an

Digital Object Identifier 10.4316/AECE.2013.01011

1582-7445 © 2013 AECE

[Downloaded from www.aece.ro on Thursday, March 28, 2024 at 17:24:03 (UTC) by 34.238.138.162. Redistribution subject to AECE license or copyright.]

input layer, a hidden layer and an output layer. The inputs of the input layer are 35 for RNA1 and 40 for RNA2. The number of neurons from the hidden layer is 30 and the output layer contains 26 neurons, each for every character of the alphabet. All neurons from the hidden layer are completely connected with the input layer and with the output layer, as shown in Fig. 1. The transfer function used for the hidden layer and for the output layer is *logsig*.

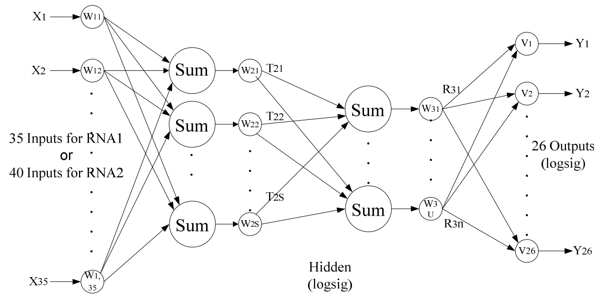


Figure 1. Architecture used for RNA1 and RNA2

In Fig. 2 a diagram of the suggested model is presented.

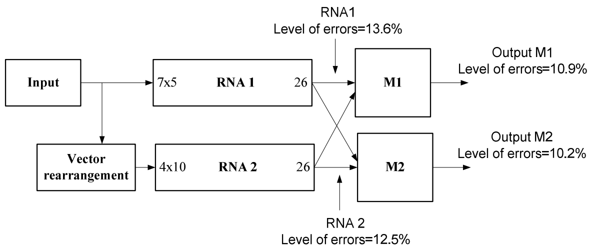


Figure 2. Architecture with two neural networks and analysis modules

The first RNA1 network has a number of 35 inputs, the input vector having the shape of a matrix of 5x7 bits and the second RNA2 network has a number of 40 inputs, the vector used at input having the shape of 10x4 bits.

For the network's training the *traingda* algorithm is used. The momentum parameters and the training rate have been improved for this type of printed character recognition application [12-14]. Training with noise can also improve generalization in feedforward networks. The training has been performed in several phases, using sets of data with different level of disturbance in 0.1-0.99 area (from 10% to 99%).

IV. Model Testing and Experimental Results

Tests have been performed by using the randn() function as a noise source from the Matlab environment, for both characters and background [15]. The charts from Fig. 3 and Fig. 4 present the distribution of these values:

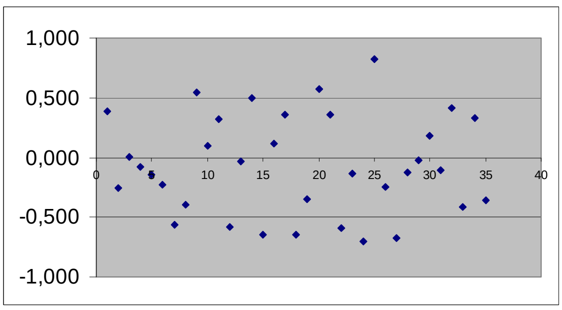


Figure 3. The distribution of values for the "A" character

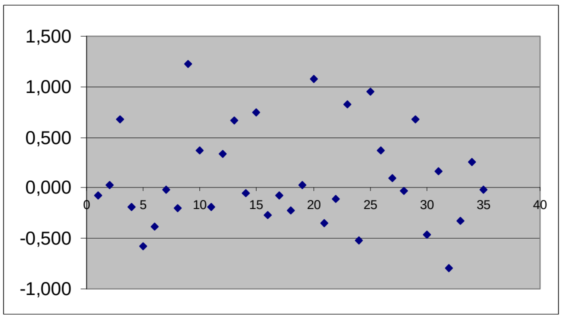


Figure 4. The distribution of values for the "C" character

It has been found that the distribution of the noise overlapping the test characters contains values within the area of -1.000 - +1.500. Bits of "0" and of "1" are modified from 0% up to 100%, the noise having a major impact over all characters and implicitly over the neural network's recognition level.

In Fig. 5, there are presented the undisturbed and disturbed A, B, C and D characters with coefficient between 0.1 and 0.99.

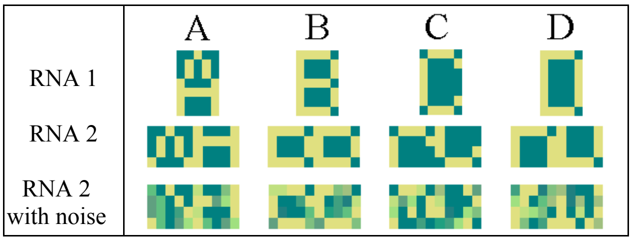


Figure 5. A, B, C and D characters undisturbed and disturbed by noise

In Table I, there are presented the disturbance effects over bits of "0" and of "1", and the response of the network.

TABLE I. DISTURBANCE LEVEL OF A, B, C AND D CHARACTERS

	A		**B**		**C**		**D**	
No. of "1" bits	16		20		13		18	
No. of "0" bits		19		15		22		17
	The bits' degradation value (in %)							
	"1"	**"0"**	**"1"**	**"0"**	**"1"**	**"0"**	**"1"**	**"0"**
	70	41	100	47	80	100	100	99
	59	38	67	39	47	100	100	79
	59	36	59	29	38	95	91	75
	56	33	21	26	35	82	80	67
	36	32	21	17	33	74	43	59
	24	9	16	12	27	68	41	57
	14		13	3	20	66	40	48
	11		9		20	34	36	46
			9			16	16	42
						9	9	9
						2	5	8
							2	
No. of "1" affected bits	8		9		8		12	
No. of "0" affected bits		6		7		11		11
Total disturbed bits	14		16		19		23	
Identification	**Match**		**Match**		**Mismatch** Identicated as "D"		**Mismatch** Identicated as "X"	

By analysing the value of the four characters presented as example, we can conclude that the 14 disturbed bits in the case of letter A, the values of 4 bits are modified by 50%

[Downloaded from www.aece.ro on Thursday, March 28, 2024 at 17:24:03 (UTC) by 34.238.138.162. Redistribution subject to AECE license or copyright.]

and for the letter B, 3 bits from the total of 16 bits are modified by 50%. In the case of letters C and D, a number of 8 bits and 10 bits respectively, are modified within the area of 40%-100%. One may observe that for a number of disturbed bits smaller than half of the total number of “1” bits, the network correctly identifies the character. When the information is strongly disturbed and this rate is exceeded the network generates errors. Tests have been performed on 10 batches, each with 100 sets of different data. Each set contains all the disturbed characters of the alphabet. The final values represent the average of those 10 batches.

Because the networks react slightly different for the same primary information, two modules of assessing the output values of the neural networks have been designed.

The M1 module assesses the values of those 52 outputs and selects the maximum value for each output. The M2 module assesses the sums obtained from the two networks and chooses the character with the best set of values as the correctly identified character. The RNA1 and RNA2 networks have been trained with the same type of noise. The results at their output are materialized in mean values in correctly identification percentages of the tested characters, of 86.4% for RNA1 and of 87.5% for RNA2. The variation of the results vis-à-vis the mean value is plus/minus ±0.5%.

For the M1 module a character recognition coefficient of 89.1% was obtained, and for the M2 module, the system’s performance reached the value of 89.8%. All the above mentioned four values are assessed for a mean noise of 50% applied to the input vector.

In Table II and Table III, there are presented as partial examples the values obtained at the output of the two networks.

TABLE II. VALUES PROVIDED AT THE RNA1 OUTPUT

	Q	**R**	**S**	**T**	**U**	**V**	**W**	**X**	**Y**	**Z**
Q	0.92									
R		0.99								
S			1.00							0.01
T				0.47						
U					0.98	**0.99**				
V	**0.99**					0.57				
W							1.00			
X								0.98	0.22	0.10
Y						0.03		**0.99**	0.97	
Z										0.98

TABLE III. VALUES PROVIDED AT THE RNA2 OUTPUT

	Q	**R**	**S**	**T**	**U**	**V**	**W**	**X**	**Y**	**Z**
Q	0.99									
R		1.00								
S			0.99							
T				0.31						
U					0.90	**0.96**				
V						0.01				
W							1.00			
X								0.99	0.07	
Y									0.97	
Z										0.17

The values from the hachured cells represent erroneous character identification. Horizontally characters entered into the network are presented and vertically we have the value of the outputs.

Table IV presents intermediate values and Table V illustrates the values obtained at the output of the M2 analysis block.

TABLE IV. VALUES PROVIDED AT THE OUTPUT OF THE M2 PARTIAL ANALYSIS BLOCK

	Q	**R**	**S**	**T**	**U**	**V**	**W**	**X**	**Y**	**Z**
...										
Q	**1.91**									
R		1.99								
S			1.99							0.01
T				0.78						
U					1.88	**1.95**				
V	0.99					0.58				
W							2.00			
X								**1.97**	0.29	0.10
Y						0.03		0.99	1.94	
Z										1.15

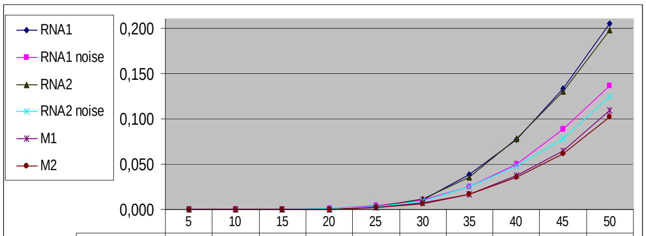
TABLE V. VALUES PROVIDED AT THE OUTPUT OF THE M2 ANALYSIS BLOCK AFTER ANALYSIS

	Q	**R**	**S**	**T**	**U**	**V**	**W**	**X**	**Y**	**Z**
...										
Q	**1.00**									
R		1.00								
S			1.00							0.01
T				0.78						
U					1.00	**1.00**				
V	0.99					0.58				
W							1.00			
X								**1.00**	0.29	0.10
Y						0.03		0.99	1.00	
Z										1.00

Assessing Table II (RNA1) we may conclude that for the X and Y outputs, the neural network reacts with values closer to 0.98 for X and 0.99 for Y. In this case, the probability of identifying the character correctly is of 50%. RNA2 from table no. 3 ensures identification of 100% for the X character and of 0% for the Y character, the final probability being of 75% for the X character and of 25% for the Y character. Based on the information provided by the second network, the data are assessed in correlation with the response from the second network (Table IV). Because RNA2 provides a value for the X output closer to 1.00 (0.99) and the value for the Y output is of 0.07, the analysis algorithm shall mark the X character as correctly identified (Table V). In the case of the disturbed V character entered into the network, the block shall identify it as the U character, thus representing an erroneous recognition.

In the diagram and table from Fig. 6 there are presented the final results of the suggested model. Horizontally, the disturbance level in percentages is presented (from 0 to 50%) and vertically it is presented the number of all the errors divided by 100.

[Downloaded from www.aece.ro on Thursday, March 28, 2024 at 17:24:03 (UTC) by 34.238.138.162. Redistribution subject to AECE license or copyright.]



	5	10	15	20	25	30	35	40	45	50
RNA1	0,000	0,000	0,000	0,000	0,003	0,011	0,038	0,077	0,133	0,205
RNA1 noise	0,000	0,000	0,000	0,001	0,004	0,011	0,025	0,050	0,088	0,136
RNA2	0,000	0,000	0,000	0,000	0,003	0,012	0,036	0,078	0,130	0,197
RNA2 noise	0,000	0,000	0,000	0,001	0,004	0,009	0,025	0,048	0,078	0,125
M1	0,000	0,000	0,000	0,000	0,002	0,007	0,016	0,037	0,065	0,109
M2	0,000	0,000	0,000	0,000	0,002	0,006	0,016	0,035	0,061	0,102

Figure 6. Diagram and values experimentally determined (noise up to 50%)

The hidden layer was reconfigured for RNA1 and RNA2 networks by increasing the number of neurons at 45, 60, 75 and 90. The results obtained at the output of M1 and M2 analysis modules are presented in Fig. 7 and Fig. 8.

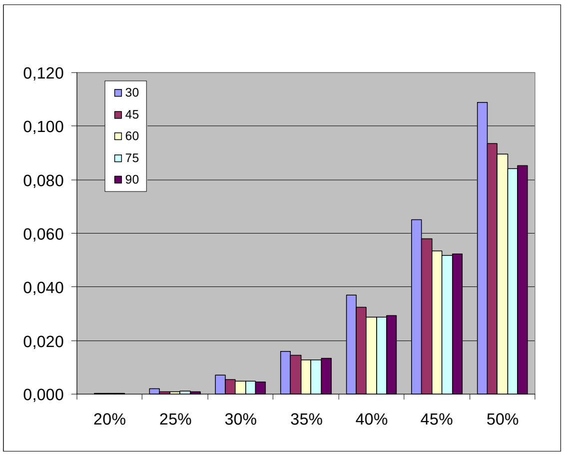


Figure 7. The number of errors (M1 module) for input noise up to 50%

Horizontally, the disturbance level is presented and vertically, the number of character identification errors.

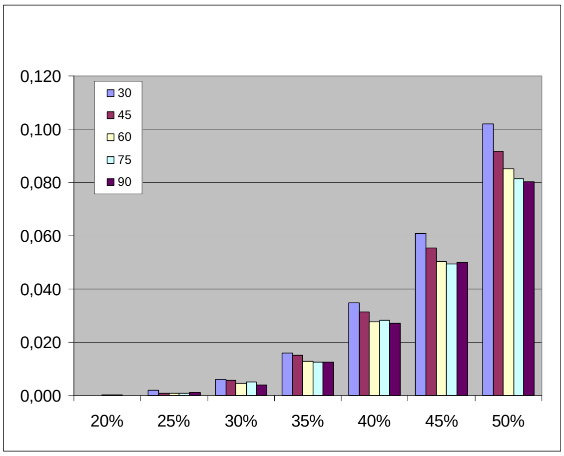


Figure 8. The number of errors (M2 module) for input noise up to 50%

For M1 module we may observe a decrease in the number of errors for 45, 60 and 75 neurons, that is 9.3%, 9.0% and 8.4% and it is found that for the configuration with 90 neurons, the number of errors increases a little (8.5%).

For the M2 module a decrease in the number of errors, from 10.2% for the configuration of 30 neurons to 9.2% for 45, 8,5% for 60, 8.1% for 75 and to 8.0% for the configuration of 90 neurons is observed. Accordingly, the system's recognition percentage increased from 89.8% (n=30) to 92% (n=90 neurons). It is found that as the number of neurons grows, it appears the phenomenon of limiting the system's performance towards the value of 91.9% for 75 neurons and of 92% for 90 neurons. The network's complexity and necessary volume calculation in the training phase are not justifying the percentage of 2.2% obtained additionally, unless the architecture is used in sensitive applications or it requires errors as small as possible.

V. CONCLUSION

For the suggested model, using the primary information reorganized in a different format, the conclusion determined based on the assessment of the values obtained at the outputs of the two modules emphasize the increase of the character recognition level up 89.8% for the M2 module, the number of errors decreasing vis-à-vis the response of the RNA2 network from 12.5% to 10.2% for a mean noise of 50% applied to the input vector. The application requires greater accuracy, and better results are obtained by increasing the number of neurons from the hidden layer of each network from 30 to 45, 60, 75 and 90 neurons. For a configuration of 90 neurons a character recognition level of 92% was obtained at the output of M2 analysis module. The suggested architecture shall be developed by using more neural networks and the analysis algorithms shall be optimized in order to improve the system's performance.

REFERENCES

[1] G. L. Martin, J. A. Pittman, "Recognizing hand-printed letters and digits using backpropagation learning", Neural Computation, vol. 3, no. 2, pp. 258-267, Summer 1991.

[2] M. Fukumi, S. Omatu, F. Takeda, T. Kosaka, "Rotation-invariant neural pattern recognition system with application to coin recognition", IEEE Trans. Neural Networks, vol. 3, no. 2, 1992.

[3] Z. Saidane, C. Garcia, "Automatic scene text recognition using a convolutional neural network", In Workshop on Camera-Based Document Analysis and Recognition, 2007.

[4] Yaoqun Xu, "Effect of white noise on chaotic neural network", Control and Decision Conference, CCDC'09, pp.3229-3234, 2009.

[5] R.M. Zur, Yulei Jiang, L. L. Pesce, K. Drukker, "Noise injection for training artificial neural networks: A comparation with weight decay and early stopping", Medical Physics, vol.36(10), pp.4810-4818,2009.

[6] F. Mamedov, Jamal Fathi Abu Hasna, "Character Recognition using Neural Networks", The 2006 World Congress in Computer Science, Computer Engineering and Applied Computing, ICAI06, 2006.

[7] G. Montavon, G. B. Orr, K. R. Muller, Neural Networks Tricks of the Trade, Springer-Verlag, LNCS7700, ISBN:978-3-642-35288-1, 2012.

[8] Yingqiao Shi, Wenbing Fan, Guodong Shi, "The research of printed character recognition based on neural network", Fourth International Symposium on Parallel Architecture, Algoritms and Programming, pp.119-122, 2011.

[9] Li Fuliang, Gao Shuangxi, "Character recognition system board on backpropagation neural network", International Conference on Machine Vision and Human-machine Interface, pp.393-396, 2010.

[10] S. Geman, E. Bienenstock, R. Doursat, "Neural networks and the bias/variance dilemma", Neural Computation 4, pp.1-58, 1992.

[11] M. I. Jordan, C. M. Bishop, "Neural Networks", ACM Computing Surveys, ISSN:0360-0300, doi:10.1145/234313.234348, 1996.

[12] A. Coates, H. Lee, A. Y. Ng, "An analysis of single layer networks in unsupervised feature learning", In AIS-TATS, 2011.

[13] A. I. Galushkin, Neural networks theory, ISBN: 978-3-540-48124-9, Springer-Verlag Berlin Heidelberg, 2007.

[14] S. Gheorghita, R. Munteanu, M. Enache, "Study of Neural Networks to Improve Performance for Character Recognition", Automation Quality and Testing Robotics (AQTR), IEEE International Conference, p. 323-326, 2012.

[15] M. Hogan, H. Demuth, M. Beale, Neural network toolbox 6 user's guide, 2008.