

UDC 621.3

N.O.Matveeva, Yu.V.Lazorenko

RECOGNITION VISUAL IMAGES OF SIGNALS USING NEURAL NETWORKS

Abstract. The using of neural networks based on multi-layer perceptron as a method for the visual images recognition is presented. Network performance is demonstrated by the recognition example and electromagnetic signals classification. Modeling was performed in Matlab environment.

Keyword: composite materials, neural networks, multilayered perceptron, defect, the visual image of signal, the function of activity.

Introduction

In industry, one of the most important issues is quality control of production. There are many methods of control, but the most used are nondestructive control methods that do not cause damage to the controlled product.

One of the methods nondestructive control of composite materials is eddy current. It can be carried out without contact of transducer and the object and obtain acceptable results of control even at high speeds displacement transducer. Eddy current method based on registration of changes in the eddy current density, so the received signal can influence the outer eddies currents. Is necessary to consider the surface roughness of composite materials. There is a task - at the analysis of processed signals get information on the availability and size of defects.

Developed many methods of pattern recognition using neural networks with different structures and characteristics so that opens a wide scope for research. Modern technologies allow to create computer systems with involving neural networks for whom the as input parameters can be used characteristics of electromagnetic signals.

Problem definition

The aim of this work is to create graphic images of electromagnetic signals obtained by scanning the composite material, and their research using created neural network to solve the problem of classification signals flaw detection.

Main part

Each artificial neural network is a set of simple elements - neurons that are connected in some way. The particular form of executable network data conversion due not only characteristics of neurons that make up its structure but also its

architectural features such as topology interneuron links directions and methods of information transfer between neurons and learning tools.

At the analysis of input and output data values of the scales and shifts the neural network automatically adjusted to minimize the difference between the desired signal and the output obtained as a result of the simulation. This difference is called the error of training and for the specific configuration of the neural network is determined by passing through a network all observations and compares the output values with desired target value. That error function is formed (criterion of quality education).

Multilayer neural networks of direct distribution are nonlinear systems that enable better qualified than conventional statistical methods. Multilayer perceptron (MLP) has a plurality of input nodes that provide input layer one or more hidden layers of neurons with sigmoid activation functions and output layer neurons with linear activation functions. At training MLP uses back propagation algorithm (back-propagation learning) [3]. Each neuron of MLP which learns based on back propagation algorithm has nonlinear smooth activation function often use nonlinear logistic sigmoid function type or hyperbolic tangent [3, 4].

Algorithms optimize learning are strategies based on the realization of the idea of iterative descent that ensure minimization of functional training. While working algorithms typically the challenge is one-dimensional search along the specified minimum direction [4].

The network learning process includes setting values weights and landslides of network to optimize network performance. Setting performance for networks with direct propagation is determined by the mean squared function (mse) between the outputs of the network (a) and targeted outputs (t) and defined by the formula [3]

$$F = mse = \frac{1}{N} \sum_{i=1}^N (t_i - a_i)^2 \quad (1)$$

Greatest interest is such indicators as value of gradient productivity and the number of inspections reliability which are used for network learning. At the slightest gradient value training meets the minimum performance. The value of the gradient can be determined yourself.

After training and testing neural network the network object can be used to calculate the answer for any input value.

The basic algorithm of back propagation errors the parameters that are adjusted in the direction as soon as possible reduce functional errors. The methods of the

second order require knowledge of the second derivative this functional errors. These methods include Newton's method, the basic step which is given by:

$$x_{k+1} = x_k - H_k^{-1} g_k, \quad (2)$$

where x_k – vector parameters for the k-th iteration; H – matrix of second partial derivatives of the objective function or Hessian matrix; g_k – gradient vector k-th iteration. Newton's method converges in many cases rather than conjugate gradient methods, but requires large costs by calculating Hessian. To avoid Hesse matrix calculations there are various means of substitution approximate expressions it creates so-called quasi-newton algorithms (eg cutting plane algorithm method or Levenberg-Marquardt) [3, 4].

Levenberg-Marquardt algorithm implements the following strategy for the estimation matrix Hesse. Assuming the functionality is defined as the sum of squared errors, which is typical in learning neural networks with direct propagation, Hessian can be approximately calculated as

$$H \cong J^T J, \quad (3)$$

gradient calculated by the formula

$$g = J^T e, \quad (4)$$

where $J = \frac{\partial J}{\partial w}$ - Jacobi matrix derivative functional errors for parameters that are set; e - error vector of the network.

Jacobi matrix can be calculated based on standard method of back propagation; it is much simpler matrix calculation Hesse. Algorithm Levenberg - Marquardt uses approximation of the Hessian of the following form:

$$x_{k+1} = x_k - (J^T J + \mu I)^{-1} J^T e_k. \quad (5)$$

When the coefficient μ is 0, getting Newton's method of approximation to the Hessian; when the value of μ is large, get the method of gradient descent with a small step. Because Newton's method has a high accuracy and convergence rate near the minimum objective is to minimize the process as soon as possible go to the Newton method. For this purpose the parameter μ decrease after each successful iteration and increase only if the test step shows that functional errors increases. This strategy after each iteration would reduce the error of the algorithm.

This algorithm is very effective implementation in the system MATLAB, which is the interpreter vector machine where the scalar product operation is realized with high precision and speed in math coprocessor computer. The main drawback of the

algorithm Levenberg - Marquardt is that it requires a lot of memory to store large matrices.

Training of the neural network made stops at one of the conditions: the function quality of training was less than the limit; gradient quality criteria became less; achieved by limiting the number of cycles; exceeded the maximum time allocated for training.

Experimental part

When done scanning composite materials using eddy current transducer signals received three forms unimodal, bimodal and gently sloping unimodal. Unimodal signal with maximum amplitude characterizes defects that exceed the size of the converter and the biggest failure bimodal peaks belong to point defects.

The paper presents to explore model signals that are obtained by scanning the surface of the composite [5] and are described by the formula:

$$y(x) = \exp(-1,5x^2) - k * \exp(-3x^2), \quad (6)$$

where k varies from 0 to 1, with $k = 0 \div 0.35$ obtain narrow unimodal signal which characterizes long cracks whose length exceeds the control zone. When changing $k = 0.35 \div 0.55$ getting gently sloping unimodal signal characteristic cracks of smaller dimension. Taking $k = 0.6 \div 1$ we obtain bimodal signal which have small cracks (with $k = 1$ - point defect).

At present, for modeling neural networks, there are many software. The article proposed to use toolkit Neural Networks Toolbox application package of MATLAB R2010b, which provided a wide range of commands and functions for the design and research of both static and dynamic neural networks.

The aim – to create a neural network, which is able to recognize the visual image signals.

In the work by using developed function *create_pic_of_sign.m* and formula (6) created images of signals grayscale size 21x13. Then convert signal is performed from one dimensional into two-dimensional matrix coding is performed as follows. For each cell in width enshrined coordinates converter on the axis x and height - the value of the signal at these points so that the lowest cell took values from 0 to 0.077, while the highest - from 0.924 to 1.001. After the conversion of two-dimensional signal we get a matrix in which each point corresponds to zero signal in the appropriate cell. With the built-in environment Matlab function *imwrite(A, filename)* creates a grayscale image.

To view the image signals were created function *plotsign(c)* with which you can see unimodal signals (Fig. 1) and bimodal (Fig. 2) form.

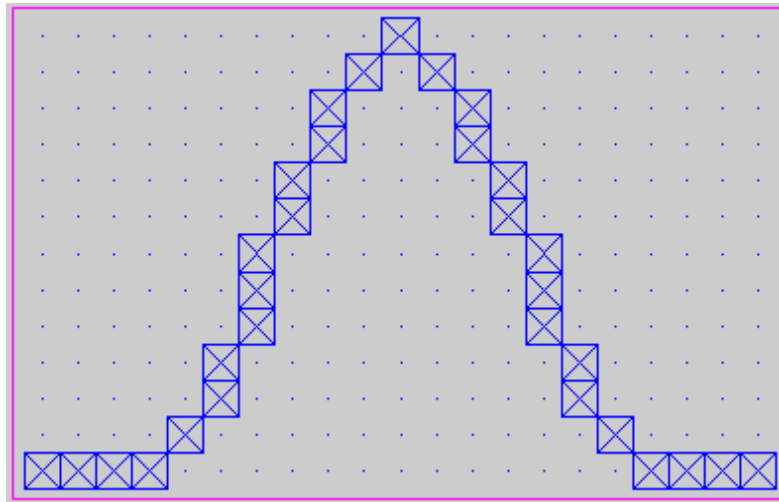


Figure 1 - Unimodal signal

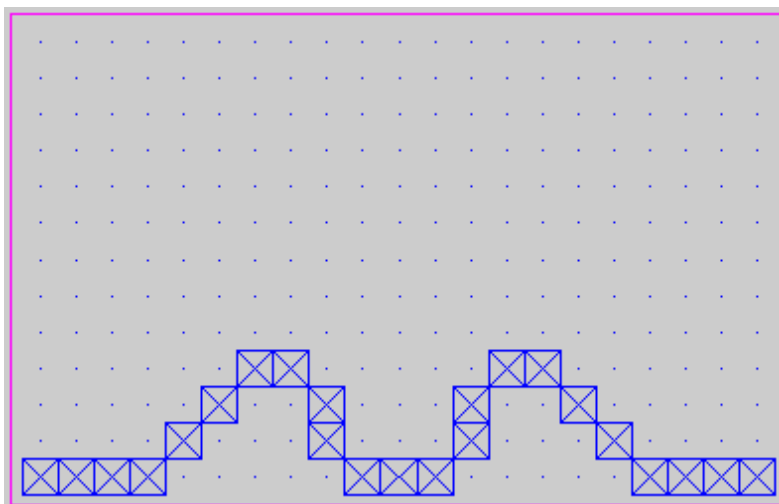


Figure 2 - Bimodal signal

To use the images created in Matlab environment it is necessary to import and process these images. For this were created function *Imgread_m(x)* which after importing figure processes it follows. Since the images saved in grayscale each pixel value that lies in the range $0 \div 255$ and used for experiments encoding format logical "1" or "0". It is necessary to replace the value 0 to 1 and 255 to 0. After processing two-dimensional matrix converted vector of 273 elements, and function *Imgread_m(x)* returns in this environment vector.

In research process for created signals have been added noise value. For this feature were developed function *noise_img(img, n_p)*, which allows you to add noise-points on the image signals. For example, add 25 points in the sound-image unimodal signal (Fig. 3).

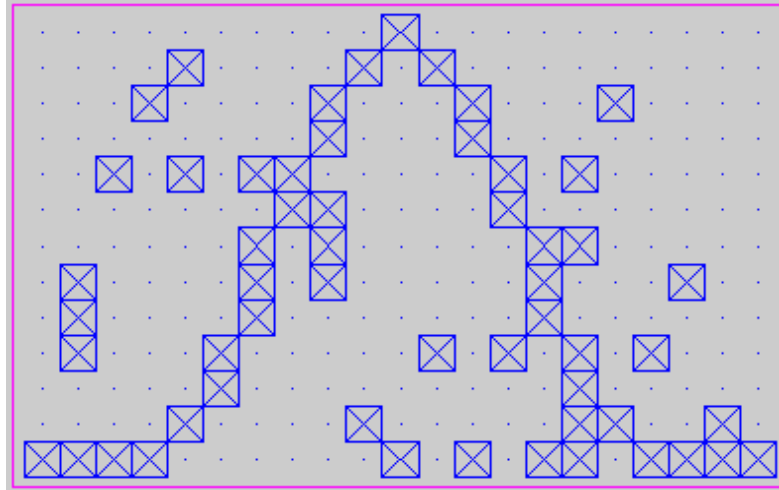


Figure 3 - Unimodal signal with 25 noise-points

To recognize the signals used double layer neural network – multilayer perceptron of 273 neurons in the input layer (number of components of the input vector), 30 neurons in the hidden layer and 2 neurons in output layer (the number of component output vector).

To create a neural network of direct distribution function used *feedforwardnet(hiddenSizes, trainFcn)*, where *hiddenSizes* - row vector of one or more hidden layer size; *trainFcn* - function studies.

To test the effectiveness of created neural network test conducted on three images signals: unimodal form gently sloping unimodal form and bimodal signal form. Initially, training was conducted on model signals without noise. Then on signals were added 30 noise-points. Created two structures of neural networks that have a hidden layer of 30 neurons and used logistic sigmoid function (*logsig*). The output layer consisted of 3 neurons (the number of waveform) and applied linear function (*purelin*).

Testing was conducted as follows: to the created neural networks submitted image signals to which gradually have been added the number of noise-point from 0 to 100. For each noise-points have been formed 100 signals with noise and output was estimated by the net. The output signal was processed M-function *compet* with the aim to select one of three signals. Then estimated the number of false classifications and estimated percentage of errors. The corresponding graph is shown in Fig. 4.

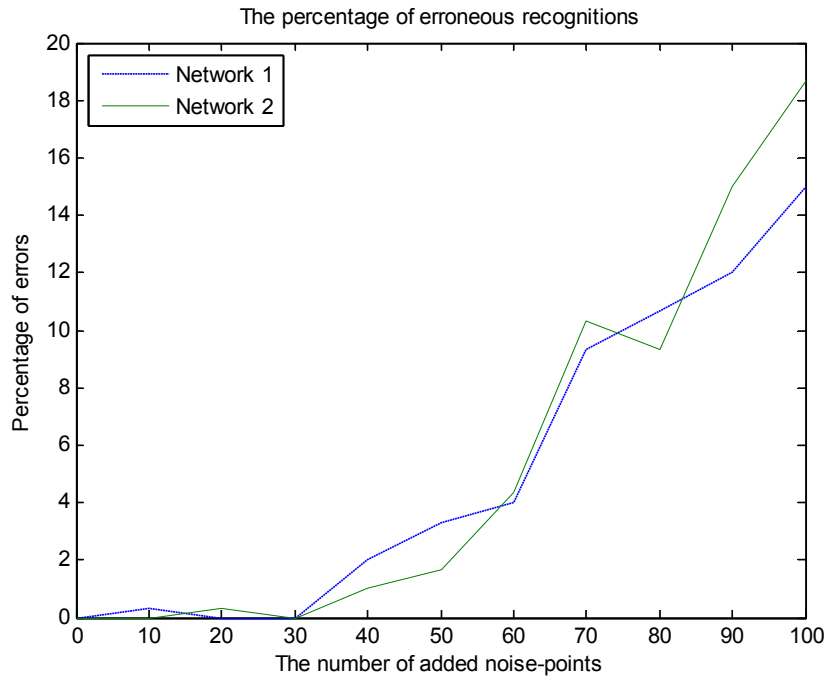


Figure 4 – Errors of the network

During further experiments created neural network used in this way. Study was conducted on the images of two signals unimodal (with $k = 0.1$) and bimodal (with $k=0.9$). The two-layer neural network is used. Only in the hidden layer were applied 30 neurons in the output layer 2 neurons used as working alternately with two waveforms. For each type of signals are used 1000 signals examples that was conducted on the model and noised signals. Testing was carried out for different values of k formula (6). They took 500 examples of each type of signal. The results are shown in Tables 1 and 2.

Table 1

Evaluation of errors in training the neural network to model signals

The signal for testing the network	0	10	20	30	40	45	50	55	60	65	70
k=0	100	97.8	90.8	88.4	84.8	83.6	82.6	79.6	79.4	80.6	76.6
k=0.2	100	98	95	92.4	89.2	89.4	88.8	86	83.8	86.2	88.2
k=0.3	100	82.6	72.8	73	76.4	71	71.4	67.4	71.4	73	70.4
k=0.6	100	99.6	99	95.6	92.2	91.4	88	87.4	82.4	82.4	78.2
k=0.8	100	99.6	98.2	95.6	93.4	90.8	89.4	87	85.4	84.4	80.4
k =1	100	99.8	98.4	95.4	92.4	92.6	89.6	87	84.8	81.2	81.6

Table 2

Evaluation of errors in training the neural network on signals with noise

The signal for testing the network	0	10	20	25	30	35	40	45	50	55	60	65	70
k=0	100	99	99.4	99.4	99	99.4	99.2	99.4	99.2	99	98.8	99	98.4
k=0.2	100	100	100	100	100	100	100	100	99.8	99.6	99.8	99.8	99.4
k=0.3	100	99.8	99	98.8	98.2	98.4	98.6	98.2	97.6	97.6	97.2	95.8	96.6
k=0.6	100	99.6	98.8	98.4	96.6	95	96.2	95.6	92.2	93.2	89.2	86.4	87.2
k=0.8	100	96.2	90.2	89.8	90	91.6	89.6	87	89.6	87.2	87.4	85.2	84
k=1	100	100	100	100	100	100	100	100	100	99.8	99.8	99.8	99.6

Conclusions. Studies have shown in principle the possibility of using neural networks to detect signals that are visual images. Training network on different sets of signals with noise allowed to teach her to work with distorted information, which is typical when conducted non-destructive testing in real conditions.

Established neural network demonstrates the quality of recognition of approximately 90% if the ideal signal is added up to 30% noise-points of the total number of points on the image.

Studies have shown that during training the neural network on with noise signals, it shows better image recognition signals.

REFERENCES

1. Хайкин Саймон. Нейронные сети: полный курс, 2-е издание.: Пер. с англ. – М.: Издательский дом «Вильямс», 2006
2. Аксенов С.В. Организация и использование нейронных сетей (методы и технологии) / С.В. Аксенов, В.Б. Новосельцев – Томск: Изд-во НТЛ, 2006. – 128 с.
3. Медведев В.С. Нейронные сети. MATLAB 6 / В.С. Медведев, В.Г. Потемкин – М.: ДИАЛОГ-МИФИ, 2002. – 496 с.
4. Дьяконов В., Круглов В. Математические пакеты расширения MATLAB. Специальный справочник. – СПб.: Питер, 2001. – 480 с.
5. Ханецкий В.С. Спектральная идентификация сигналов в дефектоскопии композитов с использованием теории статистических испытаний / Ханецкий В.С., Герасимов В.В. //Вісник ДНУ: Фізика. Радіоелектроніка. – Дніпропетровськ: – 2003. № 10. – С. 128 – 132

6. Матвеева Н.А. Моделирование нейросети для решения задачи классификации в дефектоскопии // Системні технології. Регіон. міжвуз. зб. наук. праць. - Дніпропетровськ: ДНВП «Системні технології», 2011. -Вип. 1(72). - С. 37-44.

Received 10.02.2017