***Запропоновано метод нейромережевої обробки зашумлених цифрових зображень, які можуть містити викривлені фрагменти. Метод заснований на використанні шумопригнічуючих автоенкодерів (ШАЕ). Для настроювання параметрів ШАЕ та вибору його структури застосовано нейроеволюційний підхід. Запропонована нейроеволюційна модель ШАЕ характеризується поліпшеними апроксимуючими властивостями. Результати моделювання свідчать про можливість практичного використання запропонованого методу (зокрема, для обробки даних в геоінформаційних системах)***

***Ключові слова: обробка цифрових зображень, фільтрація шумів, еволюція, популяція, штучна нейронна мережа, генетичний алгоритм, автоенкодер***

***Предложен метод нейросетевой обработки зашумленных цифровых изображений, которые могут содержать искаженные фрагменты. Метод основан на применении шумоподавляющих автоэнкодеров (ШАЭ). Для настройки параметров ШАЭ и выбора его структуры использован нейроэволюционный подход. Предложенная нейроэволюционная модель ИНС характеризуется улучшенными аппроксимирующими свойствами. Результаты моделирования свидетельствуют о возможности практического применения предложенного метода (в частности, для обработки пространственных данных в геоинформационных системах)***

***Ключевые слова: обработка цифровых изображений, фильтрация шумов, эволюция, популяция, искусственная нейронная сеть, генетический алгоритм, автоэнкодер***

UDC 004.852
DOI: 10.15587/1729-4061.2017.116134

# PROCESSING OF NOISY DIGITAL IMAGES WITH USE OF EVOLVING AUTOENCODERS

**O. Bezsonov**
Doctor of Technical Sciences, Professor*
E-mail: oleksandr.bezsonov@nure.ua

**O. Rudenko**
Doctor of Technical Sciences, Professor, Head of Department*
E-mail: oleg.rudenko@ hneu.net

**S. Udovenko**
Doctor of Technical Sciences, Professor, Head of Department**
E-mail: serhiy.udovenko@hneu.net

**O. Dudinova**
Postgraduate student
Department of Electronic Computers
Kharkiv National University of Radio Electronics
Nauky ave., 14, Kharkiv, Ukraine, 61166
E-mail: olga.dudinova@nure.ua

*Department of information systems**
**Department of informatics and computer technique***
***Simon Kuznets Kharkiv National University of Economics
Nauky ave., 9-A, Kharkiv, Ukraine, 61166

## 1. Introduction

In recent years, considerable attention has been paid to the problem of computer analysis of digital images in information systems of various functional purposes, in particular, in geographic information systems (GIS). In general, computer processing of realistic cartographic images in GIS involves preliminary processing, segmentation, recognition and interpretation. The problems of object recognition in satellite images and aerial photographs are of particular practical interest. The main difficulties of recognition include changes in the visibility of objects caused by various internal and external factors (lighting, orientation, position, presence of distorted image fragments, etc.) [1].

Analysis and interpretation of aerial photographs (or aerospace images) are an important part of the implementation of many GIS applications. These include, in particular, topographic mapping, cadastral mapping, localization of contaminated areas, monitoring of changes in the edges of parts of images.

The problem of spatial data processing in GIS for environmental monitoring can be either image improvement (restoration) by some criterion, or a special conversion that purposefully changes the image. In the latter case, image processing can be an intermediate step for further image recognition (for example, edge detection). Image processing methods can vary significantly depending on the way the image was obtained (synthesized by the computer graphics system, or by digitizing a black and white or color photograph or video frame). If images are obtained by digitizing, they are usually distorted by noise. There may be different sources of noise: camera or scanner errors, poor shooting conditions, noise in transmission over analog channels, etc. [2]. Denoising in the problems of spatial data processing in GIS serves to enhance visual perception of the analyzed images. This raises the need to improve sharpness during the edge detection, preprocessing and subsequent recognition of image fragments, etc. Promising options for the processing of noisy spatial data include the application of neural network methods that allow using parallel flowcharts of image processing by means of ANN [3].

One of the approaches widely used in the field of neural network image analysis is the use of conventional classification models that undergo supervised learning of a data array containing learning and test samples. Among the many methods that implement this approach, the support vector machine (SVM), multilayer perceptron (MLP) and convolutional neural networks (CNN), being a variation of the MLP architecture, are the most common. Various modifications of MLP are widely applied in recognition of images of certain categories, such as natural language symbols, handwritten digits, handwriting, static realistic images. However, it is often difficult to determine the MLP structure and parameters. In many applications that use direct supervised learning for image recognition, the SVM that requires less computing resources at the learning stage is more effective. At the same time, the learned MLP is more preferable for recognition, since the SVM makes predictions much more slowly in cases where the number of vectors is large compared to the sample size [4]. The CNN, being very effective in recognition of static images, are poor at processing small objects and unable to deal with distortions, such as blur or strong noise.

A promising class of neural network models applied recently to solve the problems of recognition of noisy images are autoassociative memory models or autoencoders (AE) that have certain advantages over SVM, MLP and CNN. Thus, the development of a neural network method for image filtering and recognition in the presence of noise and distortions using the modified AE model is relevant.

## 2. Literature review and problem statement

Due to the significant progress in the field of deep learning of artificial neural networks (ANN), new scopes of the AE neural network autoassociative model have been recently investigated. The AE basic architecture has initially been proposed for solving problems of classification and allocation of features of the input data using unsupervised learning. In particular, the authors of [5] first have considered the denoising AE (DAE) – a robust modification of AE used for filtering noisy images. In [6], the possibility of using the DAE to restore partially damaged images has been considered. In [7], the AE has been proposed to use for pre-learning of deep neural networks. In [8], the AE model, capable of generating fragments of images based on their spatial orientation has been presented. In [9], the AE modification that allows using the experience of previous observations to solve image reconstruction problems has been considered. In [10], the AE has been proposed to use for restoring texture information on missing image fragments. In [11], the possibility of using the convolutional AE for image reconstruction in invisible areas has been considered.

Note that the use of AE poses difficulties in choosing the network structure (the number of hidden layers and the number of neurons in each hidden layer), the parameters and the type of activation functions. Furthermore, in the known studies for the AE learning, gradient and probabilistic methods are usually applied, which greatly increases the network learning time. In this case, it is difficult to determine the optimum parameters of the algorithms providing their maximum rate of convergence, accuracy, robustness, etc. These disadvantages can be eliminated by using, in addition to traditional learning, another fundamental form of adaptation – evolution realized by applying evolutionary computation [12]. Studies on the use of the evolution principles for the DAE synthesis in noisy image processing systems in the presence of local distortions have not yet been properly developed in existing scientific publications. First of all, this is due to certain learning difficulties of such DAE, caused by the need for a large initial structured data set with different types of noise and distortion. The imbalance of data in such a set decreases the efficiency of the entire image processing flowchart using the evolution mechanisms.

To solve the problem of processing noisy and distorted digital images with the help of DAE, it is proposed to use a promising neuroevolutionary approach, allowing not only to adjust the neural network parameters, but also to determine the structure.

## 3. The aim and objectives of the study

The aim of the research is to develop a method for processing noisy and distorted digital images using the neural network evolving autoencoder. This will allow implementing an approach to noise filtering and restoration of irregular local distortions during image processing in environmental monitoring systems, which provides for a periodic scanning of landscape areas, allowing to use a sequence of similar frames for assessing current changes in the monitored objects using autoencoders.

To achieve this aim, it is necessary to accomplish the following objectives:

– to synthesize the structure of the evolving denoising autoencoder (EDAE), designed to suppress noise and restore distorted fragments in digital images;

– to develop a neuroevolutionary algorithm for the EDAE setting;

– to carry out a test simulation to assess the effectiveness of the proposed method for processing noisy and distorted images.

## 4. Architecture and learning algorithm of the evolving denoising autoencoder

An autoencoder is a type of a multilayer neural network that attempts to perform an autoassociative approximation of a function so that its output signal corresponds the input one as nearly as possible. This identical correspondence is not trivial if we impose some specific restrictions on the autoencoder neural network. These can be, first of all, restrictions on the number of neurons on the hidden layer and imposition of sparsity criteria on the activation of these neurons.

It should be noted that the AE is generally characterized by the following features:

– has a symmetric structure;

– contains an odd number of layers;

– consists of an encoder and a decoder;

– contains a so-called “bottleneck” – the encoder output layer (the decoder input layer), in which linear activation functions are used;

– the number of neurons in the “bottleneck” should not exceed the dimension of the input data, or only partial activation of these neurons (sparse activation) should occur;

– Gaussian activation functions are usually used in the AE input and output layers.

The architecture symmetry greatly facilitates adjustment of the AE parameters, since it is necessary to determine only half of all network parameters.

Let us consider a classical AE with one hidden layer. We denote the network input signals as $x(1), x(2),...,x(n)$.

First, the AE converts (encodes) the input signal $x \in [0,1]^d$ into some internal representation $y \in [0,1]^d$ using a conversion of the form

$$y = s(Wx + b), \quad (1)$$

where $W$ and $b$ are general weight and shift matrices of the network; $s$ is the nonlinear conversion function (e. g. hyperbolic tangent).

Then, the internal representation (code) of the input signal $y$ is converted back (decoded) into the signal $z$, which is the reconstruction of the input signal and has the same dimension. This conversion can be written as follows:

$$z = s(W'y + b'), \quad (2)$$

where $W'$ and $b'$ are the adjusted weight and shift matrices of the decoding conversion network.

The model parameters ($W$, $W'$, $b$, $b'$) are adjusted so as to minimize the input signal reconstruction error, which can be realized using various loss functions, e.g. quadratic function or cross-entropy.

The AE classical architecture is shown in Fig. 1 [13].

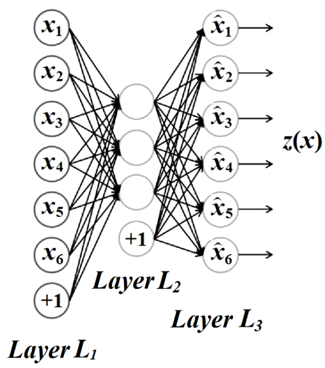



Fig. 1. AE classical architecture

Here, the L1 layer is an input layer that is encoded in the L2 layer using the hidden representation, and restoration of the input signal occurs in the L3 layer. Since the number of hidden neurons is lower than that of inputs, this forces the AE to perform data compression. It should be noted that the data conversion procedure in the AE resembles the similar procedure in the principal component analysis (PCA). If the number of hidden neurons exceeds the number of inputs, the AE can still find some useful information in the input data, establishing certain sparsity restrictions.

The denoising autoencoder (DAE) is an AE version, in which inputs are combined with some distortions (such as additive noise or masking) and the system is learned to restore data without noise [6]. Thus, the DAE can be considered as a stochastic extension of the classical autoencoder, which forces the model to learn restoration of the input signal when its noisy version is fed to the input. The stochastic data corruption process randomly sets some of the input data to zero, forcing the DAE to predict missing (corrupted) values for random subsets of missing patterns. The DAE learning flowchart can be used in denoising applications and is a popular way the autoencoder learns more meaningful hidden data representations [13].

The DAE architecture is shown in Fig. 2.

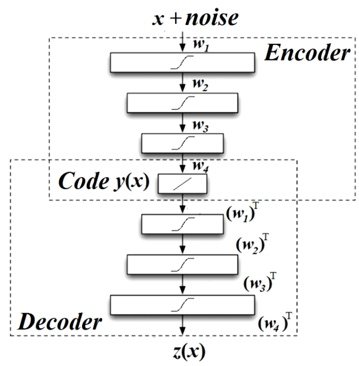



Fig. 2. Denoising AE architecture

Fig. 3 illustrates the principle of restoration of partially damaged or noisy landscape images with the help of DAE. This principle is that a network is given a number of noiseless (clean) images for learning. During the learning, all the images presented are compressed and stored in the autoencoder memory (in the form of its parameters). After the network is learned, it can be used in actual practice. It should be noted that when processing spatial data in GIS, actual images sometimes turn out to be noisy or partially damaged, as shown in Fig. 3. After such an image is presented to the previously learned DAE, the closest image will be extracted from its memory and partial loss of information will be eliminated. Obviously, the DAE can only use images from a set of memorized samples, which poses the problem of generating a learning sample.

Extensive practical application of DAE related to image processing is often hindered by the lack of a learning sequence of really clean (noiseless) data necessary for tuning encoders. In addition, the DAE parameters are usually set using gradient algorithms, the implementation of which involves cumbersome matrix inversion operations [14].

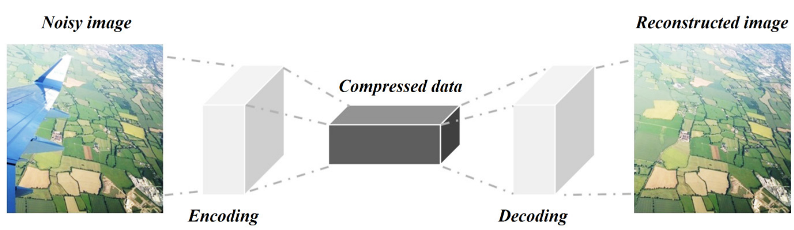



Fig. 3. Principle of processing noisy images with the help of DAE

In this paper, a DAE modification, which uses the neuroevolutionary approach, allowing not only to adjust the network parameters, but also determine its structure is proposed. Further, this modification will be referred to as the EDAE (evolving denoising autoencoder).

One of the most common types of evolutionary algorithms are genetic algorithms (GA), proposed by J. Holland. The main stages of the GA are population initialization, population estimation (fitness function computation), selection, crossover, mutation. The criterion of the GA stopping is usually either the fulfillment of the maximum permissible number of iterations, or the achievement of some predetermined threshold by the fitness function of an individual (solution).

The main difference between the neuroevolving algorithm (NA) and the gradient algorithms implemented with the help of ANN is that all operations are performed not with one network, but with a number of them (population). This approach is associated with the increased amount of memory for storing information about all networks in the population. At the same time, this allows abandoning complex operations such as matrix inversion in the network learning.

This eliminates the need to define the network structure (the number of hidden layers), the number of neurons of the hidden layer, and the type of basis functions. In addition, the problem of determining the coefficients included in the learning algorithm and affecting the EDAE learning process duration is automatically solved.

At the initial stage of the NA, the population $P_0$ with $N$ individuals (AE) is randomly initialized:

$$P_0=\{H_1,H_2,...,H_N\}.$$

Each individual in the population gets a unique description encoded in the chromosome

$$H_j=\{h_{1j},h_{2j},...h_{Lj}\},$$

consisting of L genes, where $h_{ij}\in[w_{\min}w_{\max}]$ is the value of the $i$-th gene of the $j$-th chromosome ( $w_{\min}$ – minimum, and $w_{\max}$ – maximum allowable values, respectively). The chromosome format and correspondence between the AE genes and parameters are shown in Fig. 4. It should be noted that the chromosome length is limited to the maximum permissible number of neurons.

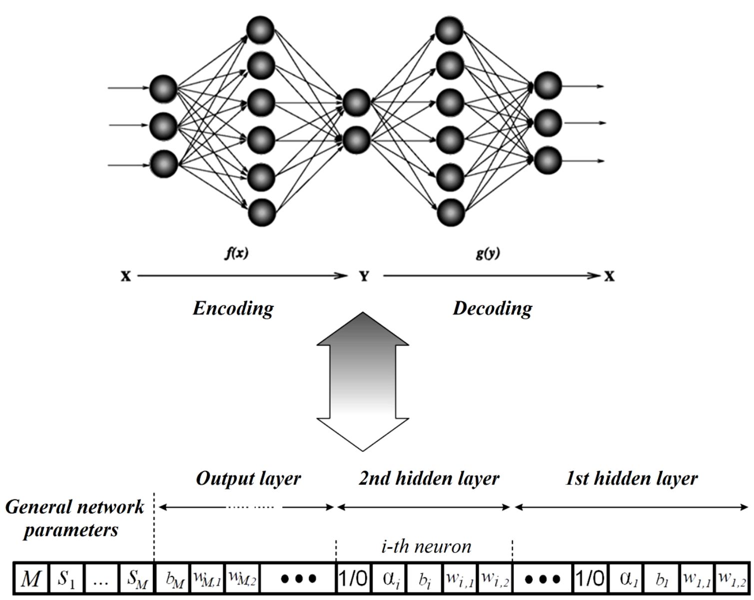



Fig. 4. EDAE chromosome formats

As can be seen from Fig. 4, each chromosome consists of genes, which store information on the relevant network parameters. The high-order bits of the chromosome contain the genes with information about the general network parameters (the maximum number of hidden layers $M$, the maximum number of neurons in each layer $S_1$,..., $S_M$). The following genes encode information on the shift of neurons of the AE output layer. Then there are blocks of genes that encode the parameters of the corresponding neurons of the hidden layer. The first gene of each block (1/0) determines whether the corresponding neuron is present in the network structure, i.e. involved or not in the calculation of the network output response to the incoming input signal. It should be noted that due to the special autoencoder architecture, the structure of only its encoding part is determined, and the structure of the decoding part is its mirror reflection. As activation functions (AF) in the AE, we use sigmoidal AF of the form

$$f(x)=(1+e^{-\alpha x})^{-1}, \tag{3}$$

so the next gene in the chromosome (α) determines the form of the AF of the corresponding neuron.

It should be noted that the mutation mechanism in GA is the main way of introducing new information into the individual's chromosomes. This can lead to entirely new gene values, which can later be added to the population gene pool. With the help of these new gene values, the GA is able to find better solutions. Mutation is an important part of genetic search and helps to prevent the population from entering local minima. As a rule, it occurs in the individual's chromosomes obtained during the crossover with some usually predetermined probability. The mutation probability can also be tied to the value of the fitness function, i. e. the lower the fitness function value, the higher the mutation probability.

In the proposed synthesis of the EDEA evolution procedure, two standard methods for gene mutation can be used. In the first case, the value of the random offspring chromosome gene is replaced with a new value in the range of admissible values

$$h_{ij}=\text{rand}[h_{\min},h_{\max}], \tag{4}$$

where rand $[x, y]$ is a random number in the $[x, y]$ interval distributed under the uniform law.

In the second case, random shift is added to the existing gene value

$$h'_{ij}=h_{ij}+\Delta h_{ij}, \tag{5}$$

where $h_{ij}$ is the gene before the mutation; $h'_{ij}$ is the gene after mutation.

However, when setting the AE used to process noisy images, the learning process is prolonged, since the adjusted parameters are diverse and their values lie in different ranges. In this regard, it becomes necessary to use adaptive mutations, which allow determining mutation parameters for each gene separately. The mutation parameters are stored in the additional vector $v$.

In the developed method, the possibility of using three types of adaptive mutations was investigated.

In the first case, the adaptive change in the mutation step was carried out as follows:

$$\nu_j(k)=\gamma\nu_j(k-1),$$

$$\gamma\in[0,1]. \tag{6}$$

Herewith, the change in the gene value occurred under the following rule:

$$h'_{ij} = h_{ij} + \nu_j(k)C(k,\tau), \tag{7}$$

where $C(k, t)$ is a random variable distributed under the Cauchy law with the parameter $\tau$. The probability density of the random variable is described by the following function:

$$f(x) = \frac{1}{\pi}\left(\frac{\tau}{x^2+\tau^2}\right). \tag{8}$$

In the second case, the mutation step was first corrected, and then the value of the gene mutating in the chromosome was changed. This procedure can be described as follows:

$$\nu_j(k) = \nu_j(k-1)\exp\left[lN(0,1)+l'N(0,1)\right]; \tag{9}$$

$$h'_{ij} = h_{ij} + \nu_j(k)N(0,1), \tag{10}$$

where $N(0, 1)$ is a random variable distributed under the normal law with zero population mean and unit variance; $l$ and $l'$ are some parameters, the following optimal values of which were obtained in the work:

$$l = (2N)^{-0.5}$$

and

$$l' = \left(2\sqrt{N}\right)^{-0.5},$$

respectively.

In the second case, the mutation step was adjusted according to the rule (9), but the gene mutation was performed as follows:

$$h'_{ij} = h_{ij} + \nu_j(k)L(\alpha), \tag{11}$$

where $L$ is a random variable distributed under the Laplace's law with the parameter $\alpha$, the probability density of which has the form:

$$f(x) = \begin{cases} \frac{1}{2}e^{\alpha x}, & \text{if } x \le 0; \\ 1-\frac{1}{2}e^{-\alpha x}, & \text{if } x > 0. \end{cases} \tag{12}$$

After the initial population is formed, the fitness of each entering individual is estimated using the fitness function. Often, when solving optimization problems, an estimate of the possible final result of the evolution algorithm is unknown. However, when solving image identification problems, the desired network response $y^*(k)$ and actual output signal $\hat{y}(k)$ are known. It is obvious that the network for which the difference $y^*(k)-\hat{y}(k)$ is minimum will be considered the fittest for solving object identification problems. In the case of off-line learning with the full sample of input-output object signals, the fitness function of the $i$-th individual can be written as follows:

$$f_i(x_j) = \frac{1}{M}\sum_{j=1}^{M}\left|y_j^*(x_j)-\hat{y}_j(x_j)\right|, \tag{13}$$

where $M$ is the sample size.

To simplify further population sorting operations, the fitness function is usually normalized

$$f_i^N(x_j) = \frac{f_i(x_j)}{\sum_{j=1}^{N} f_j(x_j)}. \tag{14}$$

Thus, to determine the network fitness, it is simulated on the entire sample, and then the network response is compared to the actual output signal of the object. The average error is the desired value of the fitness function used.

## 5. Discussion of the results of the experimental research of the proposed method

A number of experiments were carried out during the research of the proposed method of image processing with the help of EDEA. The aim was to show the possibility of using the neuroevolving encoder for processing noisy or partially damaged images.

*Experiment 1*. In this experiment, the EDEA denoising properties were investigated on the example of restoring the following noisy function of 6 variables:

$$\begin{aligned} F(x_1,x_2,x_3,x_4,x_5,x_6) &= \sin(\pi x_1)\cdot\sin(\pi x_2)\cdot\sin(\pi x_3)+ \\ &+\sin(\pi x_4)\cdot\sin(\pi x_5)\cdot\sin(\pi x_6)+ \\ &+\exp\left(-\left(x_1^2+x_2^2\right)\right)+\exp\left(-\left(x_3^2+6x_3x_4+10x_4^2\right)\right)+\xi. \end{aligned} \tag{15}$$

To do this, the EDEA population consisting of 150 individuals was used. Each autoencoder had 7 input and output neurons ($x_1$,..., $x_6$, $F$), respectively. The network structure (the number of layers and neurons in them) was determined by means of the evolutionary process, considering the EDEA structural peculiarities discussed above. The input signal corresponding to the value of the $F$ function was noisy with different distributions: Rayleigh, Laplace, and Gaussian. The results of the experiment are shown in Fig. 5–7, which show the noise histograms and the corresponding graphs of the learning error. As can be seen from the simulation results, the autoencoder shows satisfactory results with any type of noise.

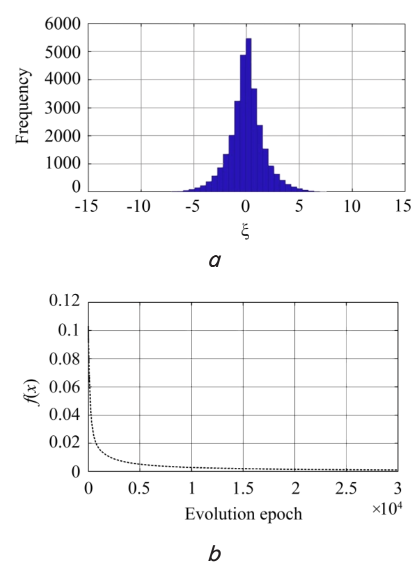
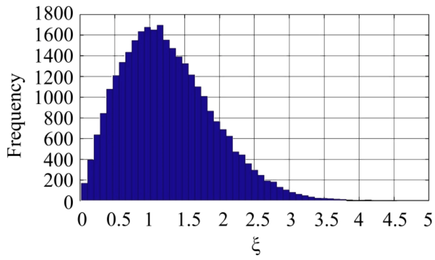
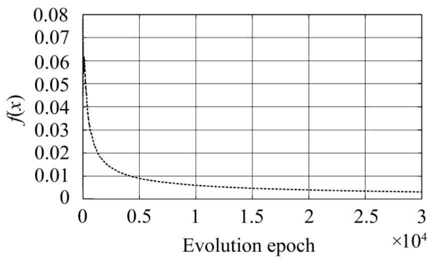



Fig. 5. Results of the experiment for the Laplace distribution: *a* — noise histogram; *b* — learning error

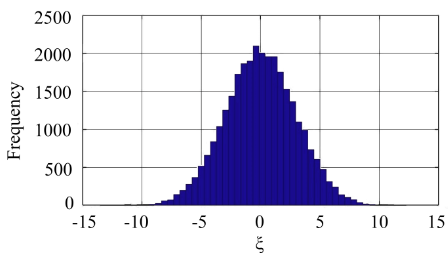



*a*

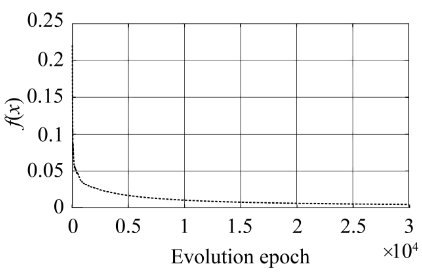



*b*

Fig. 6. Results of the experiment for the Rayleigh distribution: *a* – noise histogram; *b* – learning error




*a*




*b*

Fig. 7. Results of the experiment for the Gaussian distribution: *a* – noise histogram; *b* – learning error

*Experiment 2*. The study of the autoencoder restoring properties on partially damaged landscape images was conducted. For this purpose, the population of autoencoders consisting of 300 individuals was used. Learning was carried out with a learning sample of 1,000 different landscape images, whose size was previously reduced. The size of the images presented to the autoencoder was 80×80 pixels. Learning was conducted over 30,000 epochs. The evolutionary algorithm yielded an optimum seven-layer network, allowing to compress original images by an average of 32 %. After learning, some foreign objects, which partially obscured original information were superimposed on some images. The results of restoration of damaged images by the autoencoder are shown in Fig. 8, 9. The left parts in Fig. 8, 9 show distorted images, and the right – restored from memory of the best autoencoder in the population. As can be seen from the simulation results, the evolving autoencoder is a fairly effective neural network model for solving problems of recovering partially damaged landscape images.

*a* *b*

Fig. 8. Image restoration example 1: *a* – distorted image; *b* – restored image

*a* *b*

Fig. 9. Image restoration example 2: *a* – distorted image; *b* – restored image

During the experiments, it was found that the process learning the proposed evolving autoencoder shows satisfactory results with different types of noise typical of the process of recognition of noisy images. In addition, the possibility of recovering distorted areas of processed images with the help of EDAE was experimentally confirmed.

The advantages of the proposed approach include:

– the ability to automatically determine the structure and parameters of the ANN used for pre-processing of noisy images (for different types of noise) and recognition of their individual fragments;

– reduced computing complexity of image processing with the help of EDAE due to the use of an intermediate procedure for data compression and elimination of matrix inversion operations.

The disadvantage of the proposed approach is the lack of sufficient theoretical justification, which does not allow concluding about the EDAE effectiveness in various data processing problems.

At the same time, the results confirm the EDAE effectiveness in the problems of processing noisy and distorted images in the presence of various types of noise. Such problems include, in particular, processing of landscape images in systems of environmental monitoring of natural resources. This provides for a periodic scanning of natural and industrial areas, allowing to use a sequence of similar frames for assessing current changes in the monitored objects with the help of EDAE.

It should be noted that the research can be considered a continuation of scientific papers [2–4] in the field of creating neural network data processing systems. Further, it seems promising to carry out additional studies on the functionality expansion and theoretical justification of the proposed approach.

## 7. Conclusions

1. The structure of the evolving denoising autoencoder (EDAE) designed to suppress noise and restore distorted fragments in digital images is proposed. This neural network structure is based on the use of intermediate data compression and the evolutionary approach to setting the network parameters and determining the structure.

2. The neuroevolutionary algorithm for the EDAE tuning, allowing to synthesize a network with improved approximating and extrapolating properties, which makes it possible to efficiently process noisy images due to the iterative learning procedure is developed. The algorithm uses the fact that the structure of only the encoding part is determined in the network synthesis, and the structure of the decoding part is its mirror reflection. The flowchart of the evolutionary learning algorithm with the help of GA is given, and the EDAE chromosome formats considering the features of the problem being solved are proposed. The EDAE setting provides for the application of adaptive mutations, the parameters of which are determined for each gene separately and stored in an additional vector.

3. The test simulation confirming the possibility of using the proposed method for processing noisy and distorted images in the presence of various types of noise is performed. This confirms, in particular, promising applications of EDAE for processing landscape images, typical of problems of environmental monitoring of natural resources. Such problems provide for a periodic scanning of natural and industrial areas, allowing to use a sequence of similar frames for assessing current changes in the monitored objects with the help of EDAE.

The results of the experiments indicate the reduced complexity of processing noisy images (with various types of noise). In particular, this is achieved through the use of an intermediate procedure of data compression and elimination of matrix inversion operations in the automatic determination of the structure and parameters of the adjusted network.

References


1. Udovenko, S. G. Metody obrabotki kartograficheskih izobrazheniy [Text] / S. G. Udovenko, A. A. Shamraev, E. O. Shamraeva // 23-ya Mezhdunarodnaya Krymskaya konferentsiya «SVCh-tekhnika i telekommunikatsionnye tekhnologii». – Sevastopol': Veber, 2013. – P. 394–395.
2. Dudinova, O. B. Metod neyrosetevoy obrabotki zashumlennyh prostranstvennyh dannyh [Text] / O. B. Dudinova, S. G. Udovenko // Avtomatizirovannye sistemy upravleniya i pribory avtomatiki. – 2015. – Issue 173. – P. 14–23.
3. Bessonov, A. A. Apparatnaya realizatsiya nechetkoy seti SMAS i ee primenenie dlya zadach szhatiya izobrazheniy [Text] / A. A. Bessonov, V. A. Bobuh, O. G. Rudenko // Avtomatika. Avtomatizatsiya. Elektrotekhnicheskie kompleksy i sistemy. – 2005. – Issue 2 (16). – P. 47–52.
4. Rudenko, O. Function Approximation Using Robust Radial Basis Function Networks [Text] / O. Rudenko, O. Bezsonov // Journal of Intelligent Learning Systems and Applications. – 2011. – Vol. 03, Issue 01. – P. 17–25. doi: 10.4236/jilsa.2011.31003
5. Vincent, P. Extracting and composing robust features with denoising autoencoders [Text] / P. Vincent, H. Larochelle, Y. Bengio, P.-A. Manzagol // Proceedings of the 25th international conference on Machine learning – ICML '08. – 2008. doi: 10.1145/1390156.1390294
6. Vincent, P. Stacked denoising autoencoders: Learning useful representations in a deep network with a local denoising criterion [Text] / P. Vincent, H. Larochelle, I. Lajoie, Y. Bengio, P. A. Manzagol // The Journal of Machine Learning Research. – 2010. – Vol. 11. – P. 3371–3408.
7. Erhan, D. Why does unsupervised pre-training help deep learning [Text] / D. Erhan, Y. Bengio, A. Courville, P. A. Manzagol, P. Vincent, S. Bengio // JMLR. – 2010. – Vol. 11. – P. 625–660.
8. Hurshudov, A. A. Obuchenie mnogosloynogo razrezhennogo avtoenkodera na izobrazheniyah bol'shogo masshtaba [Text] / A. A. Hurshudov // Vestnik komp'yuternyh i informatsionnyh tekhnologiy. – 2014. – Issue 2 (116). – P. 27–31.
9. Potapov, A. S. Principle of representational minimum description length in image analysis and pattern recognition [Text] / A. S. Potapov // Pattern Recognition and Image Analysis. – 2012. – Vol. 22, Issue 1. – P. 82–91. doi: 10.1134/s1054661812010294
10. Potapov, A. S. Metod predskazaniya na osnove algoritmicheskoy veroyatnosti v zadache vosstanovleniya izobrazheniy v uteryannyh oblastyah [Text] / A. S. Potapov, O. V. Shcherbakov, I. N. Zhdanov // Opticheskiy zhurnal. – 2013. – Vol. 80, Issue 11. – P. 48–53.
11. Masci, J. Stacked Convolutional Auto-Encoders for Hierarchical Feature Extraction [Text] / J. Masci, U. Meier, D. Cireşan, J. Schmidhuber // Lecture Notes in Computer Science. – 2011. – P. 52–59. doi: 10.1007/978-3-642-21735-7_7
12. Floreano, D. Neuroevolution: from architectures to learning [Text] / D. Floreano, P. Dürr, C. Mattiussi // Evolutionary Intelligence. – 2008. – Vol. 1, Issue 1. – P. 47–62. doi: 10.1007/s12065-007-0002-4
13. Auger, A. Theory of Evolution Strategies: A New Perspective [Text] / A. Auger, N. Hansen // Series on Theoretical Computer Science. – 2011. – P. 289–325. doi: 10.1142/9789814282673_0010
14. Autoencoders [Electronic resource]. – Available at: http://ufldl.stanford.edu/tutorial/unsupervised/Autoencoders/