Висновки

Із використанням алгоритму оптимізації орієнтації елементів покращалося значення критерію для конструктивів в межах до 5 %. Цей алгоритм може ефективно використовуватись для оптимізації розміщення різногабаритних елементів.

Ефективним також є використання цього алгоритму в комбінованому поєднанні з іншими алгоритмами оптимізації. В цьому випадку застосування цього алгоритму на деякому проміжному етапі дає можливість сформувати краще розміщення, яке на наступному етапі буде базовим для інших оптимізаційних алгоритмів (наприклад, для алгоритму сканувальної області).

1. Базилевич Р.П., Щерб'юк І.Ф. Початкове розміщення елементів нарощуванням // Вісн. Держ. ун-ту "Львівська політехніка". – 2000. – № 392. – С. 54–57. 2. Базилевич Р.П., Щерб'юк І.Ф. Початкове розміщення елементів методом групування та дворівневого макромоделювання // Вісн. Нац. ун-ту "Львівська політехніка". – 2001. – № 433. – С. 48–50. 3. Базилевич Р.П., Щерб'юк І.Ф. Оптимізація розміщення елементів методом точкового сканування // Комп'ютерні технології друкарства: Зб. наук. пр. – 2000. – № 4. – С. 179–182. 4. Базилевич Р.П., Щерб'юк І.Ф. Низхідне багаторівневе розміщення різногабаритних елементів // Комп'ютерні технології друкарства: Зб. наук. праць. – 2002. – № 9. – С. 90–95. 5. Базилевич Р.П., Щерб'юк І.Ф. Алгоритми та програмні засоби для оптимізації розміщення різногабаритних елементів в локальних областях. – 2001.

UDC 004.032.26

A. Dolotov, Ye. Bodyanskiy, Ye. Viktorov
Control Systems Research Laboratory, Kharkiv National University of Radio Electronics

HETEROGENEOUS SPIKING NEURAL NETWORK IN CLUSTERING PROBLEM

Розглянуто гетерогенну спайк-нейронну мережу з рецепторними та радіально-базисними нейронами в задачі кластеризації. Запропоновано адаптивний алгоритм навчання на основі правила Гебба. Наведені результати експерименту, що були отримані під час обробки супутникового зображення.

Heterogeneous spiking neural network with receptive and radial-basis function neurons in clustering problem is considered. Adaptive learning algorithm based on the Hebbian rule is proposed. Results achieved in experiment on satellite image processing are presented.

Introduction

Nowadays clustering, or unsupervised classification task, is the one of the challenging problems of Data Mining and Exploratory Data Analysis. There exists a variety of approaches to solve it. One of the most advanced and well-known of them is applying of Computational Intelligence methods [1] such as artificial neural networks (self-organizing maps, ART networks, BSB models) [2] and fuzzy clustering methods [3, 4].

Last ten years new, the third generation of artificial neural networks [5–7], commonly known as spiking neural networks, has appeared and has been evolving. It was shown that spiking neural networks (SNN) operate in a biologically plausible way. They are computationally more powerful over conventional artificial neural networks (ANN) when solving common Data Mining and Exploratory Analysis tasks. Compared to ANNs with sigmoidal transfer functions, SNNs also require less number of neurons to process data effectively [6, 8].

The main difference of SNN against ANN is input data encoding method. If one compares generations of neural networks in scope of automatic control theory [9], ANN can be treated as nonlinear pulse-amplitude systems, while SNN can be treated as time-pulse systems that transmit and transform data in successive impulses, or spikes.

Spiking neural networks can be applied to solve clustering problem [10-12]. They are capable of performing rapid and effective processing of complex and high dimensional data, for example sufficiently complex images. But it should be noted that since SNNs are on the initial stage of their development, the number of known architectures and learning (including unsupervised learning) algorithms is not large enough. In this paper architecture of multilayered heterogeneous spiking neural network in clustering problem with arbitrary complex shapes of classes is proposed.

Heterogeneous spiking neural network architecture

Architecture of the self-learning multilayered spiking neural network for clustering is shown on fig. 1. Evidently it is heterogeneous feed-forward neural network with lateral connections in the output layer.

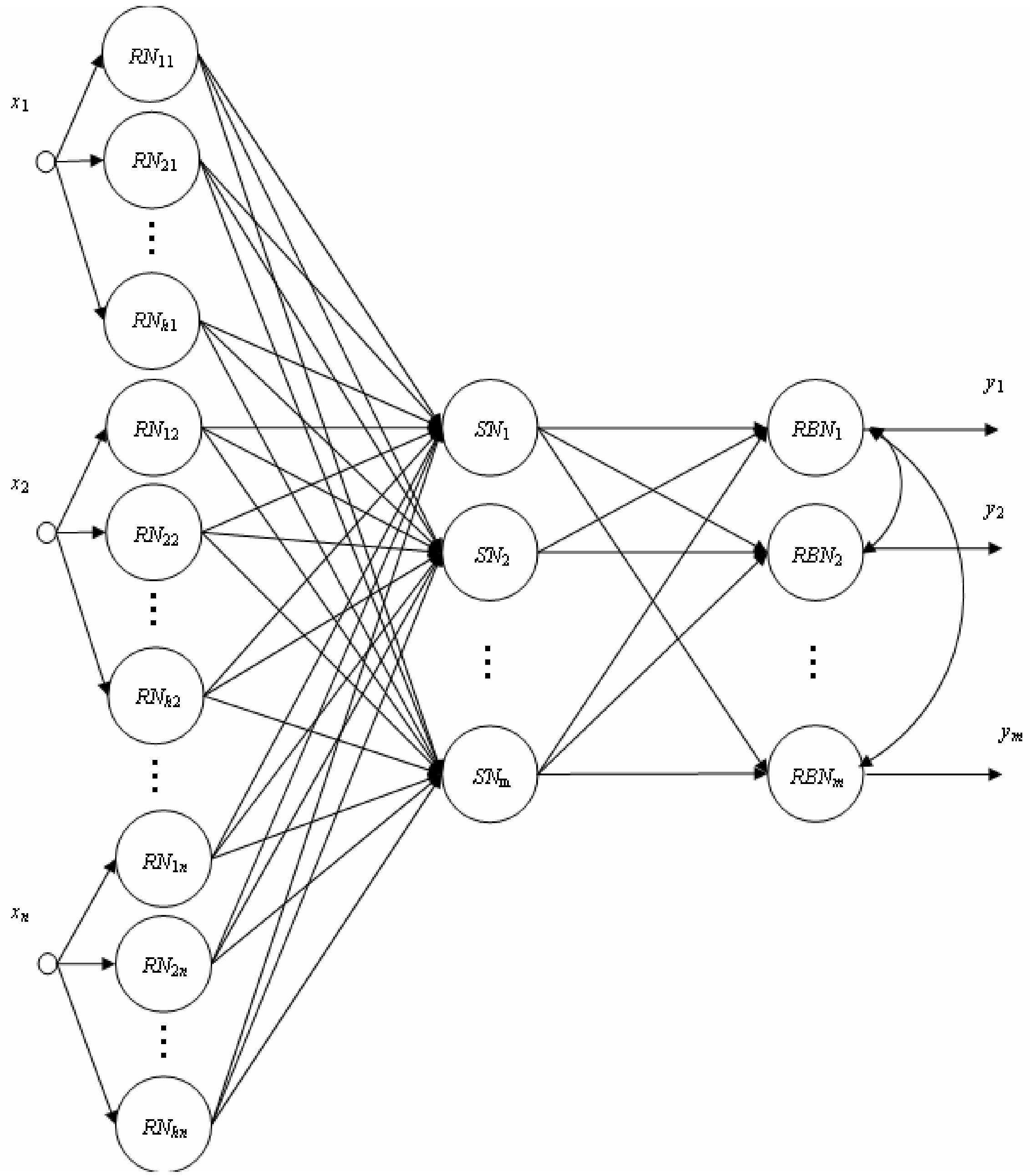

Fig. 1. Heterogeneous spiking neural network

The first hidden layer is designed for encoding input analog signal x_i, $i = 1, 2, \ldots, n$, previously converted so that $0 \le x_i \le 1$, to spike (pulse) trains where two any spikes are distinguished by their firing rate. It is the layer where pulse-amplitude signal are transformed into time-pulsed form. A population coding [11, 13-15] is implemented in the proposed SNN architecture to increase accuracy of clustering. Population encoding scheme implies that a signal is processed by a set of receptive neurons at the same time. The set consists of h receptive neurons RN_{li}, $l = 1, 2, \ldots, h$. As a rule their activation functions are shifted, overlapped, and bell-shaped (Gaussians usually).

Firing time t_{li} of spike produced by receptive neuron RN_{li} depends on distance between center of the neuron and the i-th component of input signal: the nearer it is to center, the early the neuron emits spike.

The second hidden layer consists of m compartmental spiking neurons SN_j [10, 16] (fig. 2) with complex synapses. Each complex synapse includes parallel subsynapses with various axonal time delays d^p, $p = 1, 2, \ldots, q$, d^p-d^{p-1}>0.

A spiking neuron has hn inputs, each of them corresponds to complex synapse MS_{li} formed by q subsynapses. A subsynapse consists of parallel connected post-synaptic potential shapers $\varepsilon^p_{jli}(t)$ and weights w^p_{jli}. Weights are adjustable parameters in the architecture of SNN. Having a spike with firing time t_{li} on the li-th input, subsynapse generates post-synaptic potential

$$\varepsilon^p_{jli}(t) = \varepsilon(t - t_{li} - d^p), \tag{1}$$

Where

$$\varepsilon(t) = \begin{cases} \frac{t}{\tau}\exp(1-\frac{t}{\tau}), t \ge 0, \\ 0, t < -0, \end{cases} \tag{2}$$

τ is a time constant, its value can be obtained empirically.

The complex subsynapse output is described by following expression

$$\mathrm{u}_{jli}(t) = \sum_{p=1}^{q} w^p_{jli}\, \varepsilon(t - t_{li} - d^p), \tag{3}$$

and the output of the whole compartmental spiking neuron is

$$\mathrm{u}_j(t) = \sum_{i=1}^{n}\sum_{l=1}^{h}\sum_{p=1}^{q} w^p_{jli}\, \varepsilon(t - t_{li} - d^p). \tag{4}$$

It is notable that the compartmental spiking neuron architecture is the same as the neo-fuzzy-neuron one [17–12], and the complex synapse is the same as the nonlinear synapse while they are designed for vastly different purposes.

The output layer of SNN includes m conventional radial-basis function neurons RBN_j, $j = 1, 2, \ldots, m$, and the center of each radial-basis multidimensional function $c_j = (c_{j1}, c_{j2}, \ldots, c_{jn})^T$ matches to the vector-prototype of corresponding cluster. Radial-basis function neurons of the output layer are connected by lateral connection implementing the “winner-takes-all” mechanism. Such scheme is meant to determine membership of an input pattern $x(k)$ to a certain cluster identically to self-organizing map clustering scheme.

Mean delay from input to output is considered [10] to determine coordinates of radial-basis function neurons centers in the output layer. It is computed according to the expression

$$d_{ji} = \frac{\sum_{l=1}^{h}\sum_{p=1}^{q} w^p_{jli} d^p}{\sum_{l=1}^{h}\sum_{p=1}^{q} w^p_{jli}}, j = 1,2,\ldots,m,\ i = 1,2,\ldots,n. \tag{5}$$

On the next step centers are calculated on the base of expression (5)

$$c_{ji} = d_{ji} - \min_i\{d_{ji} \,|\, 1 \le i \le n\}, \tag{6}$$

and they are assign to bell-shaped activation functions (multidimensional Gaussians typically) $\varphi_j(\| u - c_j \|^2, \sigma^2)$ of neurons in the output layer

$$\varphi_j(\| u - c_j \|^2, \sigma^2) = \exp(-\frac{\| u - c_j \|^2}{\sigma^2}) \tag{7}$$

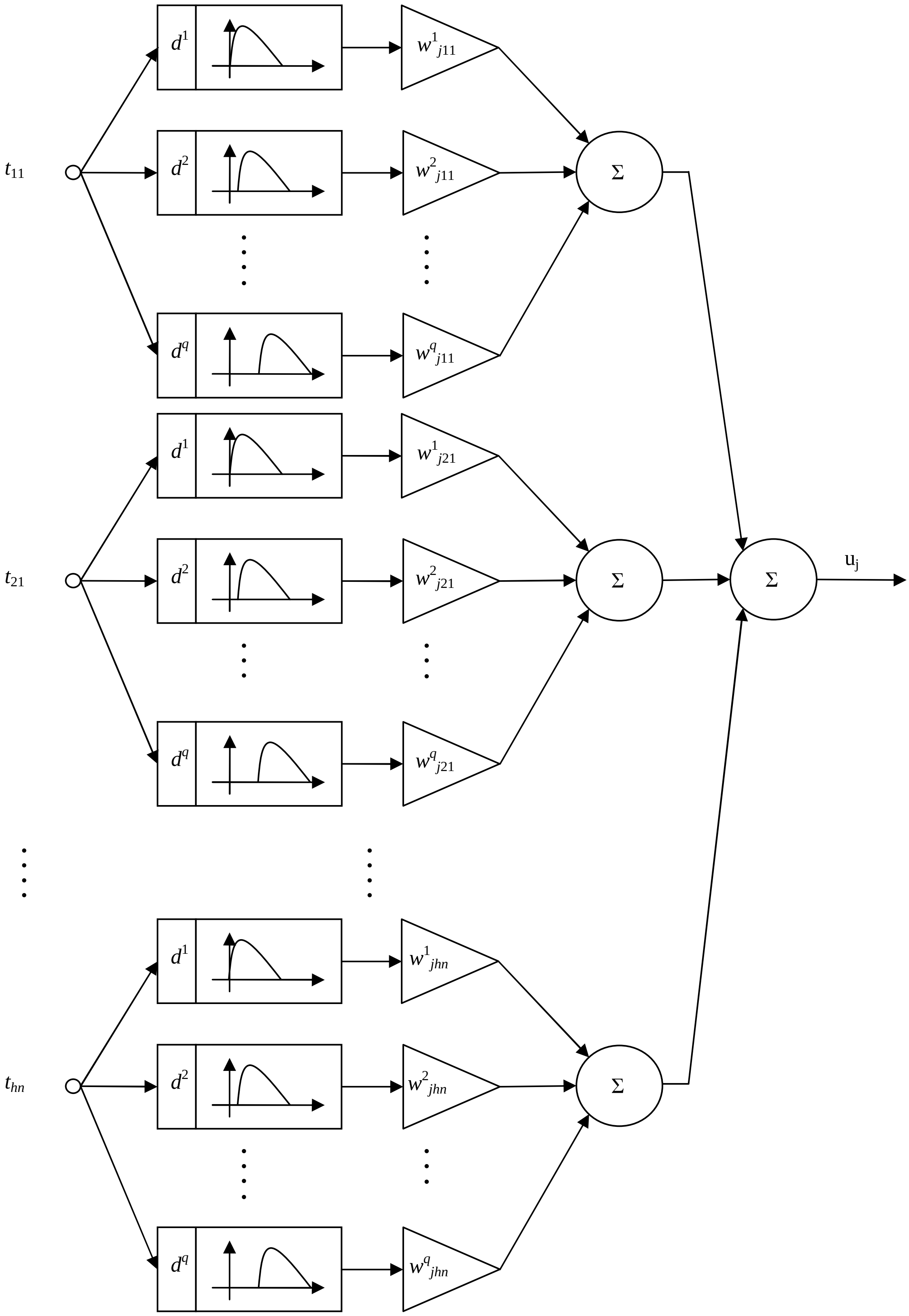

Fig. 2. Compartmental spiking neuron

When clustering the neuron that has the largest firing rate is considered a “winner” and only it emits output signal y_j indicating membership of the input pattern to the j-th cluster.

Spiking neural network learning

Unsupervised learning of spiking neural network is usually based on the Hebb rule that is following

$$w_{jli}^{p}(k+1)=w_{jli}^{p}(k)+\eta_{w}(k)\,\mathrm{L}(\Delta t_{jli})=w_{jli}^{p}(k)+\eta_{w}(k)((1+\beta)\exp\frac{(\Delta t_{jli}-\alpha)^{2}}{-\nu^{2}\Big/2\ln\frac{\beta}{1+\beta}}-\beta), \tag{8}$$

where k is the current epoch, η_w is a learning rate, α, β, ν – shape parameters of partial derivative of the learning criterion with respect to the adjustable variable (offset, shift, and width respectively), Δt_{jli} is a firing time delay of the j-th spiking neuron when it gets a spike from the li-th receptive neuron with corresponding delay of the p-th subsynapse:

$$\Delta t_{jli}=t_{li}+d^{p}-t_{j} \tag{9}$$

t_{li} is a firing time of the l-th receptive neuron for the i-th component of input signal, d^p is a time delay of the p-th subsynapse, t_j is fiting time of the postsynaptic neuron.

Function L(·) is set to provide the following rule: subsynapses which contributed to the neuron-winner’s firing are strengthened and synapses which did not contribute are weakened during learning. Thus, weights are adjusted the way to move the center of the corresponding radial-basis function neuron nearer to input pattern.

After synaptic weights w^{p}_{jli} have been adjusted, centers of radial-basis functions in the output layer can be calculated according to expression (6). In case when data processing is performed in real time, values of centers are calculated on the base of values of current weights $w^{p}_{jli}(k)$, obtained according to (8):

$$d_{ji}(k)=\frac{\sum_{l=1}^{h}\sum_{p=1}^{q}w_{jli}^{p}(k)d^{p}}{\sum_{l=1}^{h}\sum_{p=1}^{q}w_{jli}^{p}(p)},\ j=1,2,...,m,\ i=1,2,...,n, \tag{10}$$

$$c_{ji}(k)=d_{ji}-\min\{d_{ji}(k)\,|\,1\le i\le n\}, \tag{11}$$

and then they must be smoothed according to the Kohonen self-learning rule

$$\bar{c}_{j}(k+1)=\bar{c}_{j}(k)+\eta_{c}(k)(c_{j}(k)-\bar{c}_{j}(k)), \tag{12}$$

where $\bar{c}_j(k+1)$ is a smoothed value of radial-basis function of the j-th neuron on the k-th epoch of processing, η_c is a learning rate, it is usually set to satisfy the conditions of Dvoretsky.

Example of Image Clustering

Spiking neural network was tested on the color satellite image of 841x542 size (fig. 3).

Trained set contained a half of all pixels that had been chosen randomly. RGB-entry of each pixel was used as input. The network had 3 receptive fields for each input coordinate and 5 radial-basis neurons in the output layer. Self-organizing map was run over the same train set to compare its performance with SNN’s one. Each cluster was assigned to shades of gray to visualize obtained results (fig. 4). The results were obtained on the 4th epoch by SNN and on the 50th epoch by self-organizing map.

Thus, proposed SNN requires at least by an order of smaller number of epochs than self-organizing map requires.

Fig 3. Satellite image of Svoboda Sq., Kharkiv City

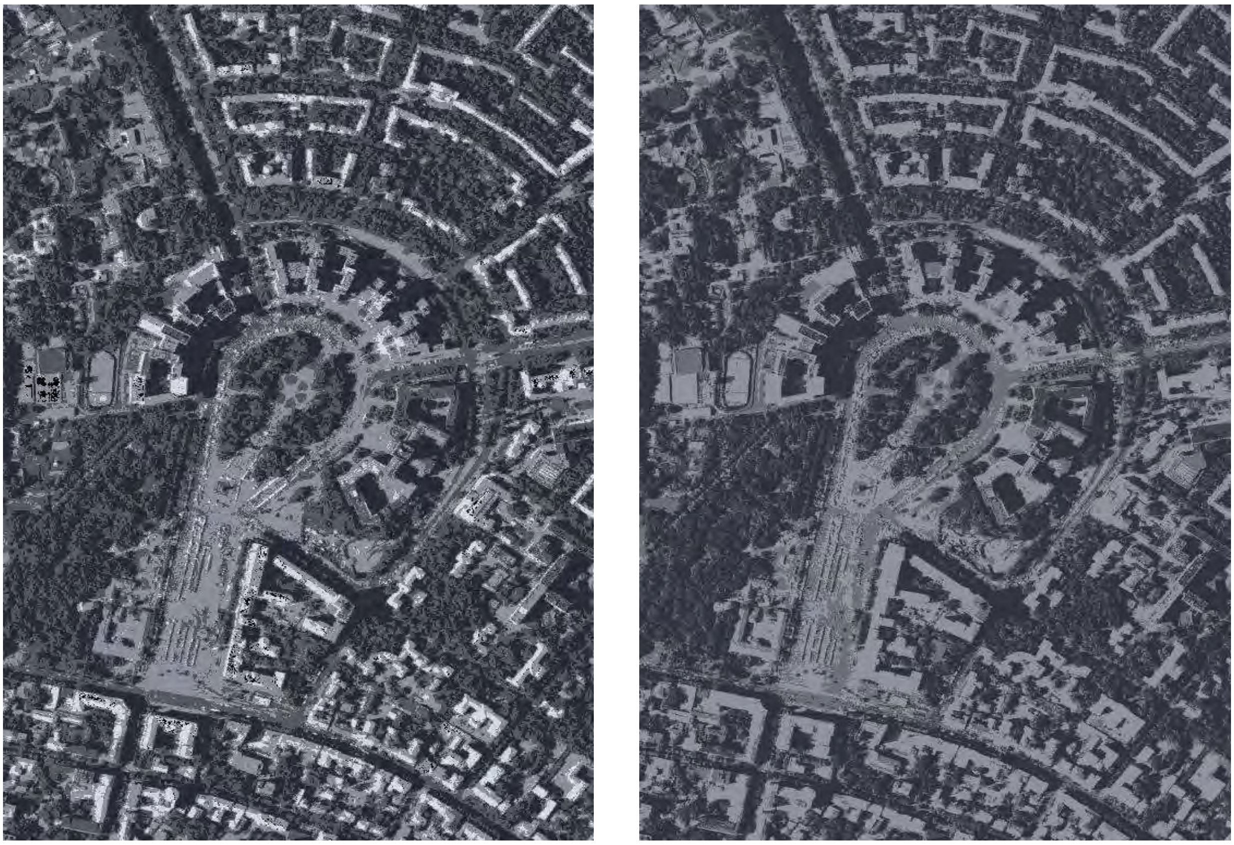

Fig 4. Left-to-right: the satellite image clustered by SNN and by self-organizing map

Conclusions

Architecture and unsupervised learning algorithm of SNN that provides sufficiently rapid data processing of high quality is considered. The spiking neural network is produced in a biologically plausible way and thus it differs from the data processing scheme of conventional artificial neural networks.

1. Jang J.-Sh. R., Sun Ch.-T., Mizutani E. Neuro-Fuzzy and Soft Computing: a Computational Approach to Learning and Machine Intelligence. – Upper Saddle River, N.J.: Prentice Hall, Inc., 1997. – 614 p. 2. Haykin S. Neural Networks. A Comprehensive Foundation. – Upper Saddle River, N.J.: Prentice Hall, Inc., 1999. – 842 p. 3. Höppner F., Klawonn F., Kruse R. Fuzzy-Clusteranalyse. Verfahren für die Bilderkennung, Klassifikation und Datenanalyse. – Braunschweig: Vieweg, 1996. – 280 s. 4. Sato-Ilic M., Jain L.C. Innovations in Fuzzy Clustering. – Berlin Heidelberg New York: Springer, 2006. – 152 p. 5. Maas W., Bishop M. Pulsed Neural Networks. – Cambridge, MA: The MIT Press, 1998. – 408 p. 6. Maass W. Networks of spiking neurons: the third generation of neural network models // Neural Networks. – 1997. – 10. – P. 1659–1671. 7. Natschläger T. Networks of spiking neurons: a new generation of neural networks models. – 1998. – http://www.igi.tugraz.at/tnatschl/online/3rd_gen_eng/3rd_gen_eng.html. 8. Maass W. On the computational complexity of networks of spiking neurons. / Eds. by G. Tesauro, D.S. Touretzky, T.K. Leen "Advances in Neural Information Processing Systems 7". – Cambridge, MA: The MIT Press, 1995. – P. 183–190. 9. Цыпкин Я.З. Теория линейных импульсных систем. – М.: Физматгиз, 1963. – 968 с. 10. Natschläger T., Ruf B. Spatial and temporal pattern analysis via spiking neurons // Network: Computations in Neural Systems. – 1998. – 9. – P. 319–332. 11. Bohte S.M., Kok J.S., La Poutre H. Unsupervised clustering with spiking neurons by sparse temporal coding and multi-layer RBF networks // IEEE Trans. on Neural Networks. – 2002. – 13. – P. 426–435. 12. Долотов А.И. Обучение без учителя на основе сети спайк-нейронов // Системный анализ и информационные технологии: Материалы IX Междунар. науч.-техн. конф. (15–19 мая 2007 г., Киев). – К.: НТУУ “КПИ”, 2007. – С. 104. 13. Snippe H.P. Parameter extraction from population codes: a critical assesment // Neural Computation. – 1996. – 8(4). – P. 511–529. 14. Zhang K., Sejnowski T.J. Neuronal tuning: to sharpen or broaden? //Neural Computation. – 1999. – 11. – P. 75–84. 15. Долотов А.И., Викторов Е.А. Применение рецепторных полей в спайк-нейронных сетях при решении задачи кластеризации // 11-й Междунар. молодежный форум “Радиоэлектроника и молодежь в XXI веке”: Сб. материалов форума. – Харьков: ХНУРЭ, 2007. – Ч. 1. – С. 351. 16. Bohte S.M., Kok J.S., La Poutre H. Error back-propagation in temporally encoded networks of spiking neurons // Neurocomputing. – 2002. – 48. – P. 17–37. 17. Yamakawa T., Uchino E., Miki T., Kusanagi H. A neo fuzzy neuron and its applications to system identification and prediction of the system behavior // Proc. 2nd Int. Conf. on Fuzzy Logic and Neural Networks “IIZUKA-92”. – Iizuka, Japan, 1992. – P. 477–483. 18. Miki T., Yamakawa T, Analog implementation of neo-fuzzy neuron and its on-board learning / Ed. N. E. Mastorakis “Computational Intelligence and Applications”. – Piraeus: WSES Press, 1999. – P. 144–149. 19. Bodyanskiy Ye., Kokshenev I., Kolodyazhniy V. An adaptive learning algorithm for a neo fuzzy neuron // Proc. 3rd Int. Conf. of European Union Society for Fuzzy Logic and Technology “EUSFLAT 2003”. – Zittau, Germany, 2003. – P. 375–379. 20. Kolodyazhniy V., Bodyanskiy Ye., Otto P. Universal approximator employing neo-fuzzy neuron / Ed. B. Reusch. – Berlin-Heidelberg: Springer, 2005. – P. 631–640.