NEURAL SELF-ORGANIZATION IN PROCESSING HIGH-RESOLUTION IMAGE DATA

A. Barsi [a]

[a] Dept. of Photogrammetry and Geoinformatics, Budapest University of Technology and Economics
Muegyetem rkp. 3, H-1111 Budapest, Hungary, barsi@eik.bme.hu

ISPRS Working Groups I/2, I/5, IC WG II/IV

KEY WORDS: Neural networks, self-organization, object description

ABSTRACT:

The paper presents a novel artificial network type, which is a generalization of the famous Self-Organizing Feature Map (SOFM). The newly developed Self-Organizing Neuron Graph (SONG) algorithm keeps the Kohonen-type learning rule of SOFM, but enables more flexibility by the use of undirected graphs. The SONG algorithm is based on the generalized adjacency between the graph nodes. After the short presentation of the algorithm, two main application areas are shown: the description of roads in junctions and the detection of building structures. In both areas the necessary preprocessing is reduced to minimal in order to visualize the power of the SONG technique. The road junction tests show, how the initial graph finds the right position, even for rotated images. The building tests demonstrate the effect of the increased graph complexity. The presented examples clearly prove, that the developed self-organizing method can be an efficient tool in analyzing high-resolution image data.

1. INTRODUCTION

The artificial neural networks have become very popular tools in many technical applications. The history of this achievement begins in the 40's with the work of W. McCulloch and W. Pitts. [Rummelhart 1988] In 1961 K. Steinbuch published his paper about the learning matrix, which is the first publication about the competitive learning technique and the "winner-takes-all" algorithm [Carpenter 1989]. The first mention of the self-organization can be found in the paper of Ch. von der Malsburg, although he worked out a special retina model and the work is rather from biologic aspects [Malsburg 1973] Ten years after the competitive learning algorithm, the Finnish scientist, T. Kohonen published the *instar* and *outstar* learning techniques [Carpenter 1989]. In 1984 he published the book "Self-organization and associative memory", which is after a revision and update currently the most famous work about neural self-organization [Kohonen 1984, Kohonen 1995].

The Kohonen-type neural network is the Self-Organizing Feature Map (SOFM), which will be described in Chapter 2. The author of the current paper has developed a novel network type, which is an extension of the SOFM technique. The algorithm is described in Chapter 3, which is followed by the documentation of several applications in processing high-resolution image data.

2. SELF-ORGANIZING FEATURE MAPS

The Self-Organizing Feature Map is an unsupervised, competitive learning technique, which works on a regular neuron grid of given lattice topology. The algorithm requires a given set of data points in form of vectors:

$$\mathbf{x} = [\xi_1, \xi_2, \ldots, \xi_n] \in \mathbb{R} \tag{1}$$

The neurons are stored similarly in vector form:

$$\mathbf{m} = [\mu_1, \mu_2, \ldots, \mu_n] \in \mathbb{R} \tag{2}$$

supposed we have n-dimensional data space.

The algorithm consists of two steps:

1. *ordering*, which is a phase delivering only rough weights
2. *tuning*, which refines the neurons weight vectors.

Each step has the same sub steps, namely

1. the selection of the winner neuron
2. the weight update for the winner and its neighbors.

The winner is selected by the evaluation of a distance measure between the data points and the neuron weights:

$$\|\mathbf{x} - \mathbf{m}_c\| = \min_i \{\|\mathbf{x} - \mathbf{m}_i\|\} \tag{3}$$

where

c is the identifier of the winner neuron.

The reformulated condition gives directly the identifier c:

$$c = \arg\min_i \{\|\mathbf{x} - \mathbf{m}_i\|\} \tag{4}$$

The distance measure can be any distance norm, in the practice mostly the Euclidean one is used.

In the weight update phase the weights of the winner and its neighboring neurons are modified as follows

$$\mathbf{m}_i(t+1) = \mathbf{m}_i(t) + h_{ci}(t)[\mathbf{x} - \mathbf{m}_i(t)] \tag{5}$$

where

$\mathbf{m}_i(t)$ is the current weight vector to be modified

$\mathbf{m}_i(t+1)$ is the weight vector of neuron i in the next iteration (called epoch)

$h_{ci}(t)$ a weight factor.

Equation (5) is the *Kohonen learning rule*; it means, that the weights after the modifications are calculated from the difference (distance) between the neuron and the data point multiplied by a factor. The factor is a monotonically decreasing function in time, and it consists of a second distance norm to consider the neuron neighborhood:

$$h_{ci}(t) = \begin{cases} \alpha(t) & \text{if } i \in N_c(t) \\ 0 & \text{if } i \notin N_c(t) \end{cases} \qquad (6)$$

where

$N_c(t)$ is the neighborhood function – it decreases in function of t

$\alpha(t)$ is the learning rate – also a monotonically decreasing function of t.

For detailed presentation of the algorithm please refer [Kohonen 1995].

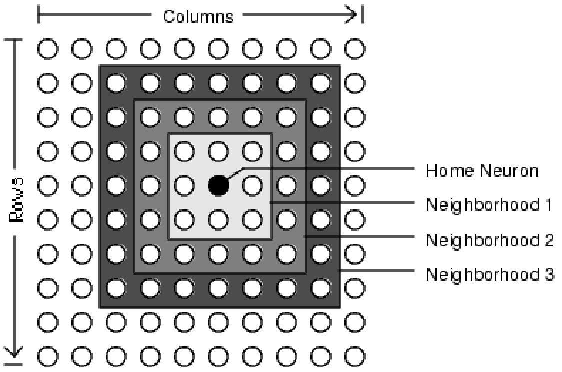

Figure 1. The regular neuron grid is the base of the SOFM algorithm. After the winner has been found, its neighborhood ($N_c(t)$) must be defined, as marked by concentric gray squares

3. SELF-ORGANIZING NEURON GRAPH

The newly developed Self-Organizing Neuron Graph (SONG) is a combination of undirected graphs and the Kohonen learning rule. The base of the SONG method is the neuron graph, which is represented by a neuron weight vector (as in Equation (2)), and an adjacency matrix, which contains the connectivity information. The matrix is defined as

$$\mathbf{A}_{ij} = \begin{cases} 1 & \text{if node i and j are connected} \\ 0 & \text{otherwise} \end{cases} \qquad (7)$$

The matrix $\mathbf{A}$ is square and symmetric. The above, generally spread definition expresses only the direct connections, i.e. if a neuron can be reached from other neuron only via another neuron, this relation is not stored in $\mathbf{A}$. Fortunately the meaning of adjacency has been generalized [Bollobás 1998, Balakrishnan 2000, Buckley 1990].

Following the generalized adjacency term, the notation of the above matrix is $\mathbf{A}^1$. The general adjacency means the number of minimal connections between the nodes. This matrix is $\mathbf{A}^n$, if the greatest distance between two nodes is maximally $n \in \mathbb{N}$. The matrix $\mathbf{A}^n$ is fully filled except the main diagonal, which remains zero. Beside the full adjacency, partial adjacency is also computable: $\mathbf{A}^k$, where $k < n$ and $k \in \mathbb{N}$.

The generalized adjacency ($\mathbf{A}^n$ or $\mathbf{A}^k$) can be derived from $\mathbf{A}^1$ by the special k-th power of the original matrix. The essence of the method is the followings: $\mathbf{A}^k_{ij} = k$ if $\mathbf{A}^k_{ij} \neq 0$ and $\mathbf{A}^s_{ij} = 0$, where $s = 1, 2, \ldots, k-1$. [Henley 1973, Bronstein 2000, Swammy 1981]

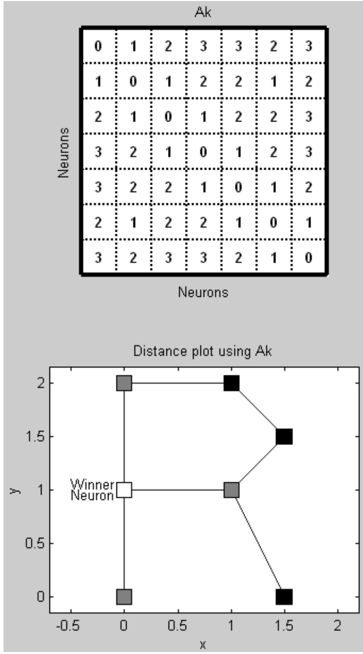

Figure2. The generalized adjacency matrix expresses the whole connectivity information. The neurons of the graph (the letter R) are numbered clockwise from 0,0. The winner neuron is the second, so the second row of the matrix shows the neighborhood values [Barsi 2003a]

The SONG algorithm keeps the main method of the SOFM technique (ordering and tuning phases etc.), but modifies the factor definition in the Kohonen learning rule as followings:

$$h_{ci}(t) = \begin{cases} \alpha(t) & \text{if } \mathbf{A}^k_{ci} < d(t) \\ 0 & \text{otherwise} \end{cases} \qquad (8)$$

where

$\alpha(t)$ is a learning rate function, similarly to the one of SOFM

$\mathbf{A}^k_{ci}$ is the generalized adjacency value between the winner neuron (c) and the i-th neuron

$d(t)$ is a monotonically decreasing distance function.

The functions $\alpha(t)$ and $d(t)$ can be linear, exponential, hyperbolic or other decreasing functions, which can be parameterized with start and end value. $\alpha(t)$ must be between 1 and 0, $d(t)$ between k and 1.

The neural network model of SONG can manage the SOFM models, because the regular neuron grid can be represented by graphs.

A detailed description of the SONG algorithm is presented in [Barsi 2003a].

4. APPLICATIONS

4.1 Road junctions

The road junction tests are performed on high-resolution black-and-white orthophotos. The imagery was taken about the region of Frankfurt am Main, Germany, with a ground resolution of 0.4 m. Two four-arm junctions and a three-arm junction of rural region are selected as representatives to show, how the developed method works.

The junction cut-offs were preprocessed with simple image processing operators: intensity threshold followed by several morphology operations (dilatation, erosion and connected component labeling). The preprocessing has produced the necessary data points; their creation has great importance, because SONG organizes the graph to these points!

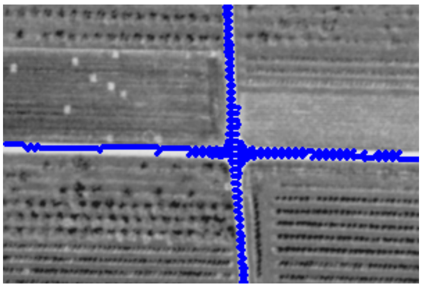

Figure 3. The first four-arm junction with the road pixels marked by simple image processing operators. Resampling was performed to get speed acceleration, which caused the zig-zag pattern

The initial neuron graph came from a standard imagination of a four-arm road junction (Figure 4).

The graph consists of 5 neurons, the data point set of 533 points. The ordering phase had 300 epochs; the starting and ending neighborhood were 2 and 1 respectively. The learning function started with 0.1 and was decreased to zero. In the tuning phase only the direct neighborhood was considered; the learning rate was decreased from 0.01 to 0 in 500 epochs. The result of the run can be seen in Figure 5.

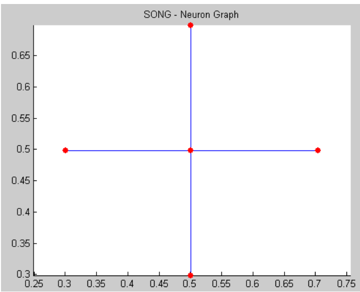

Figure 4. The initial graph of a four-arm junction with local neuron coordinates as weights

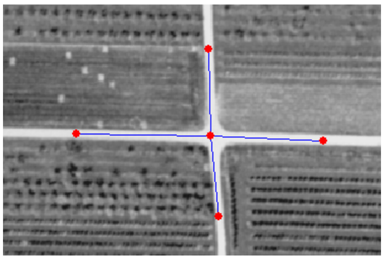

Figure 5. The initial neuron graph has found the data points along the roads

The images were rotated to prove the rotation invariancy.

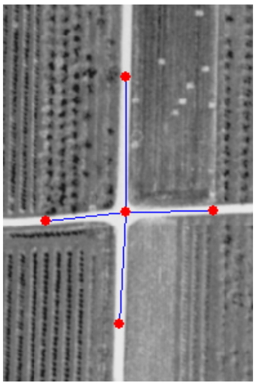

Figure 6. The algorithm on 90 degree rotated image

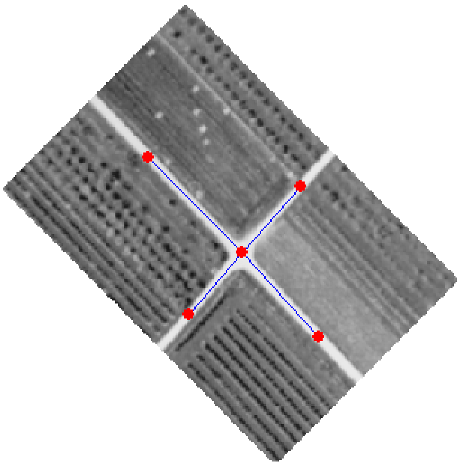

Figure 7. The resulting graph on 45 degree rotated image

The same graph was applied on another four-arm junction, where the angle of the road was not rectangular and the original image is similar to Figure 7.

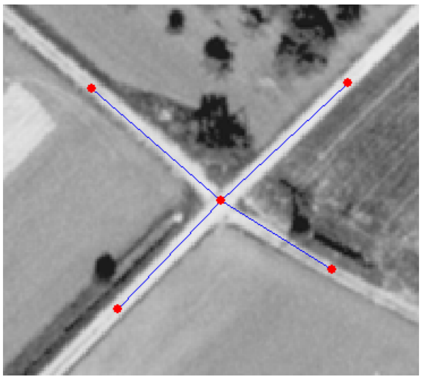

Figure 8. The result of the run on other junction cut-off

The test with the three-arm junction was performed in the same way. After the intensity threshold, morphologic operators result 437 data points (Figure 9). The applied initial neuron graph was very simple, as Figure 10 shows.

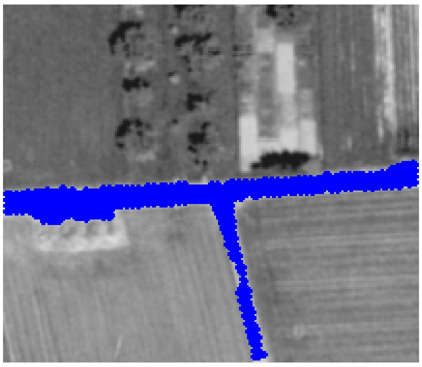

Figure 9. Data points on a three-arm road junction

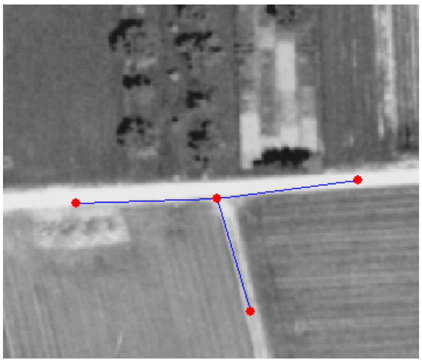

Figure 10. The final neuron graph describes the roads in the three-arm junction

The algorithm was controlled as follows: the ordering phase had 100 epochs, the learning rate boundaries were 0.01 and 0, and the neighborhood descreased from 2 to 1. The tuning was somewhat longer with 500 epochs, it focused only on the winner neurons (there were no neighborhood considered), and the learning rate values were 0.001 and 0.
The rotated tests were completed in the same manner, but only the 45 degree rotated version is shown in Figure 11.

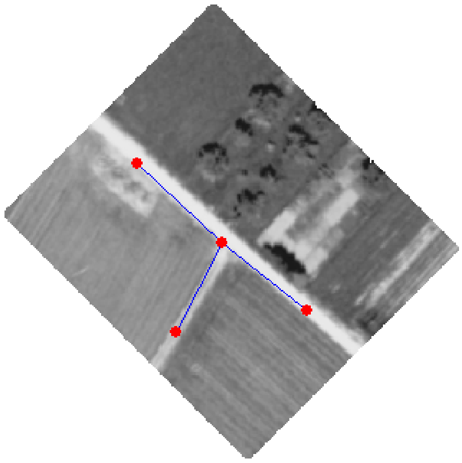

Figure 11. The rotated image was no problem for the detection of the three-arm junction

4.2 Building description

The analysis of the man-made objects has other great application field, namely the buildings. The detection of the building can be started by an image segmentation, which must be followed by a description. The coming chapter demonstrates, how can be applied the novel development for building description.
The input image is a 1 m color IKONOS image, which was a free demo in the Internet. The image was taken on 09.08.2000 about Singapore.
The preprocessing consists of a rule-based classification by intensity values of the RGB channels. The classification was followed by different binary morphology operators. The final data set had 2120 points, which was resampled (Figure 12).

Figure 12. The building data points projected on the original IKONOS image

The initial neuron graph had in the tests 11, 13, 19 and 23 neurons. The increased complexity is shown in Figure 13.

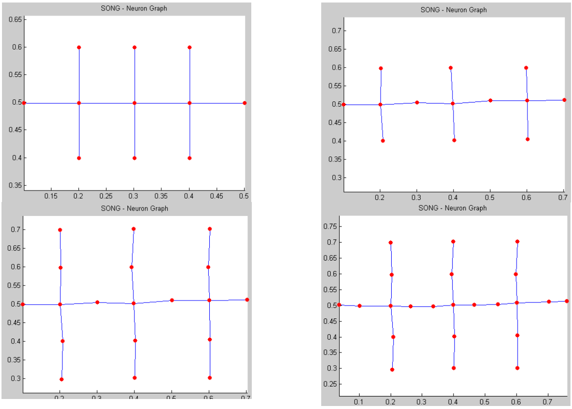

Figure 13. The initial graphs for the building tests. The complexity increases: the number of neurons are 11, 13, 19 and 23 respectively

The control parameters of the building graphs can be read in Table 1.

	Case A	Case B	Case C	Case D
$\alpha(0)$	0.01	0.01	0.001	0.001
$N_c(0)$	4	6	8	10
Epochs	200	200	600	600
$\alpha(0)$	0.0001	0.001	0.00001	0.0003
$N_c(0)$	2	2	1	1
Epochs	500	1000	3000	1500

Table 1. The main control parameters of the building tests. Case A had 11 neurons, Case B 13, Case C 19 and Case D 23. The first three parameters are for the ordering phase, while the last two for the tuning phase

In all four cases the final learning rate values were zero in both phases. The neighborhood was decreased till one is reached. The increased graph complexity can achieve better roof cover as the following figures have proved.

Figure 14. The result of the simplest case

Figure 15. Better fit to the roof pixels can be obtained with the complexity grow

Figure 16. The most complex neuron graph together with the roof data points. One can notice, how the graph spans out the data points

5. CONCLUSIONS AND OUTLOOK

The developed Self-Organizing Neuron Graph was tested in two application fields. The description of a road junction can be achieved by a simple neuron graph, even if the junction roads are oriented occasionally. One of the strongest features of the SONG method is this robustness, which was presented by the three- and four-arm junction examples.

The algorithm can be coupled to a junction detection step, where the approximate junction position is used to limit the work area. (The detection of the junction can happen by artificial neural networks, as [Barsi 2003c] illustrates.) Thereafter even very simple operators can mark the road pixels, which deals as data points for SONG. Of course, if more sophisticated road detection technique is involved, the quality of the result is better guaranteed.

It must be underlined, that the presented preprocessing methods or the possible sophisticated solutions (like thematic classifications) segment the raw image. The segmentation result is mainly two groups: roads (data points) and non-roads. This operation causes a kind of independency from the neighborhood. It means, that the land cover types of the parcels among the roads shall not be considered.

The result of the SONG run can serve as vector input for geographic information systems (GIS). Such systems contain large databases with roads, where the exact position of all

junction roads is not always secured. The proposed method can help in the analysis of the quality of these databases. The analysis can be extended with the calculation of the orientation angles of roads.
Beside this quality management aspect the algorithm is suitable to extend the road extraction techniques in order to give better description in the connections of road segments.
The building detection tests study another aspect of the developed algorithm. The simplest neuron graph finds the structure of the object, but by additional neurons the description accuracy is raised.
The high-resolution images taken by satellite sensors have huge information content, in which analysis the newly developed neural technique can help. The shown results have proved the usability in road and building detection, but further examples could also mentioned. The neuron graph can be a neuron chain, where linear structures become detectable (e.g. the Venetian Canal Grande on IKONOS image [Barsi 2003b]). Or the structure of the graph is similar to regular geometric figures, like [Barsi 2003a] demonstrates in the Pentagon test (Quickbird image after the Attack!).
The control of the proposed algorithm is nowadays a manual task: the right parameter set requires several trials with control variations. The development of the near future will focus on the automatic setting of the control values. Better object description and better understanding of the phenomena can be obtained by performing more tests of different objects (targets).

ACKNOWLEDGEMENT

The author expresses his thanks to the German Federal Office for Cartography and Geodesy (BKG) for the orthoimages, furthermore to the Alexander von Humboldt Foundation for the scholarship, when the work was started. Special thanks go to the Hungarian Higher Education Research Program (FKFP) for financing the workshop participation.

REFERENCES

R. Balakrishnan, 2000, *A Textbook of Graph Theory*, Springer, New York

A. Barsi, 2003a, Neural Self-Organization Using Graphs, in: *Machine Learning and Data Mining in Pattern Recognition, Lecture Notes in Computer Science*, Vol. 2734, Springer, Heidelberg, pp. 343-352

A. Barsi, 2003b, Graph Based Neural Self-Organization in Analyzing Remotely Sensed Images, *IEEE International Geoscience and Remote Sensing Symposium (IGARSS) 2003*, Toulouse, Vol. VI, pp. 3937-3939

A. Barsi – C. Heipke, 2003c, Detecting road junctions by artificial neural networks, *2nd GRSS/ISPRS Joint Workshop on Remote Sensing and Data Fusion over Urban Areas (URBAN 2003), IEEE-ISPRS-ASPRS-EARSEL*, Berlin, pp. 129-132

B. Bollobás, 1998, *Modern Graph Therory*, Springer, New York

I. N. Bronstein – K. A. Semendjajev – G. Musiol – H. Mühlig, 2000, *Taschenbuch der Mathematik*, Harri Deutsch, Frankfurt am Main

F. Buckley – F. Harary, 1990, *Distance in Graphs*, Addison-Wesley, Redwood City

G. A. Carpenter, 1989, Neural Network Models for Pattern Recognition and Associative Memory, *Neural Network*, No. 2, pp. 243-257

E. J. Henley – R. A. Williams, 1973, *Graph Theory in Modern Engineering*, Academic Press, New York

T. Kohonen, 1984, *Self-Organization and Associative Memory*, Springer, Berlin

T. Kohonen, 1995, *Self-Organizing Maps*, Springer, Berlin

Ch. von der Malsburg, 1973, Self-Organization of Orientation Sensitive Cells in the Striate Cortex, *Kybernetik*, No. 14, pp. 85-100

D. E. Rummelhart – J. L. McClelland (Eds.), 1988, *Parallel Distributed Processing, Explorations in the Microstructure of Cognition*, Vol. I-II., MIT Press, Cambridge

M. N. S. Swammy – K. Thulasiraman, 1981, *Graphs, Networks and Algorithms*, Wiley, New York