Optica Applicata, Vol. XXVIII, No. 2, 1998

Method for extraction of contours for identifying characteristic information in noisy images

TADEUSZ NIEDZIELA, ARTUR STANKIEWICZ*, LESZEK R. JAROSZEWICZ, IDZI MERTA

Institute of Applied Physics, Military University of Technology, 01–489 Warszawa, ul. Kaliskiego 2, Poland.

*Institute of Air Force Technology, ul. Księcia Janusza 42, 00–961 Warszawa, Poland.

The problem of extraction of a contour from highly noised images is discussed. The method for contour extraction investigated is known in the literature as regularisation with discontinuities. This method, when applied to highly noised images, gives good results and, moreover, may by adapted to a simple cellular neural network. This possibility allows a real-time contour extraction to be performed. The idea of the method of regularisation with discontinuities is presented and one of its variants for the case of temperature noise is used for illustration. The method discussed is examined by means of different functions of the extinction noise temperature.

1. Introduction

Automatic recognition of image objects is based on extraction of their characteristic features [1], [2]. One of them is the object contour which, in turn, can be defined in different ways. Many papers dealing with contour extraction and regularisation have been published recently. In the case of clean images, the problem is trivial and can be solved using the well-known gradient or laplacian method. On the other hand, contour extraction from noised (camouflaged) images is not effective with the use of these methods. Several methods for contour extraction from noised images with subsequent regularisation are discussed in the literature. The regularisation method based on energy defined as a negative sum of scalar products of gradient vectors in neighbouring points of image has been proposed. The best knwon method for contour extraction from noised images applies operators choosing contour points as peaks of Gaussian-smoothed image gradients [3]. In this method, the operator finds directional gradient maximum parallel to gradient direction. This method is different from simple application of laplacian described by CANNY [4].

Contour extraction from noised images can also be performed by regularisation with discontinuities [5]–[9]. The latter term generally means the smoothing of a signal or its transform outside the points in which the gradient magnitude exceeds a given threshold. In the case of image analysis, regularisation with discontinuities consists in periodic averaging of the image or its transform in the areas with low gradient and, alternately, in the areas with high gradient as contour points. This method is time-consuming due to a large number of calculations and cannot be applied to fast contour extraction in the case of serial calculations. However, due to their locality, such a process can be easily performed in a neural network with suitable architecture [10], [11]. The locality of calculations means that only the image pixels which are placed inside the circle of a given radius are simultaneously introduced into calculations. Another advantage of the method presented here is small (equal to 1) radius of the neighbourhood which is satisfactory for practical accomplishment of the neutral network.

Parallel calculations in neural networks offer the possibility of real-time treatment of an image. Consequently, results are calculated irrespective of image dimensions. This feature is specially important when applying the regularisation method as an initial treatment and extraction of object features from an image.

The results of regularisation presented in this work have been obtained by means of software simulating the operation of a neural network. The network architecture employed here consists of two coupled neural layers. One of them describes the image – the results of regularisation, while the other describes the contour. Regularisation equations are derived from the definition of the eigenfunction of energy (the cost function) of states of the neural network. Afterwards, equations of the neural network movement in the space of its states are derived. Introduction of an additional temperature noise into the regularisation network constitutes the new element of the method discussed here. This step makes it possible to omit many local energy minima which are the main threat to correctness of regularisation results.

2. Functional scheme of the neural network

A functional scheme of the neural network performing regularisation with image discontinuities is presented in Fig. 1. This network consists of three general matrix elements: the image matrix d_t and two main layers of the neural network. The first layer of the network f_t represents the image being regularised, while the second layer l_t, represents the binary contour of the image regularised. Both layers are con-

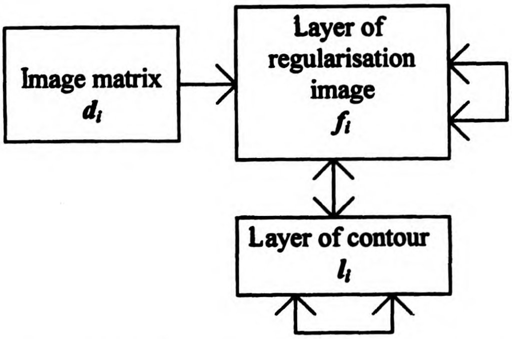

Fig. 1. Functional scheme of the architecture of the neural network performing image regularisation with discontinuities

nected in a specific way described in detail in the following sections. In general, the layers f_t and l_t are connected recurrently, while the layer f_t is undirectionally connected with the matrix d_t. Moreover, local connections occur between neurons inside the f_t and l_t layers.

3. Definition of energy

The essence of the method applied is the equation describing the energy of neural-network states. For the one-dimensional signal, this equation has the form of a sum proposed by GEE [7]

$$E = \sum_i [\varkappa^2(f_i - \Gamma(d)_i)^2 + \lambda^2(\nabla f)_i^2(1-l_i) + \alpha l_i] \tag{1}$$

where the subscript i stands for the number of signal samples, d_i – for the input signal, f_i – for the regularised signal, and l_i – for the contour matrix. The last matrix is a zero-unitary matrix that takes the value of unity if the contour occurs in the i-th point and the value of zero, otherwise. The coefficients $\varkappa$, l and α can be interpreted as the weights of particular energy terms. The coefficient $\varkappa$ describes the weight of the trend of the regularised signal to the input signal. The coefficient l describes the weight of the term smoothing the regularised signal. The last coefficient α stabilises the energy giving the weight of the cost of introducing the unity into the contour matrix. Such a construction of this coefficient prevents the occurrence of a global minimum in case all the matrix elements should be unitary. The operator $\Gamma(d)$ is an input operator acting on the signal d. In this work, the input operator of the form $\Gamma(d) = d$ is accepted.

In the case of a two-dimensional image, the signal gradient becomes a vector with components equal to two image derivatives calculated in orthogonal directions. In this case, the energy of the neural network states can be expressed in the form proposed by HERTZ [6] and GEE *et al.* [7]

$$E = E_1 + E_2 + E_3 + E_4 \tag{2}$$

where E_1 is the term responsible for the trend of the regularised signal to the original one

$$E_1 = \sum_{i,j} \varkappa^2(f_{i,j} - d_{i,j})^2 \tag{3}$$

where i and j are the image coordinates, E_2 is the term responsible for the smoothing of the regularised image containing the spatial derivatives of the image

$$E_2 = \sum_{i,j} \lambda^2 [(f_{i,j} - f_{i+1,j})^2(1 - l_{i,j}^x) + (f_{i,j} - f_{i,j+1})^2(1 - l_{i,j}^y)], \tag{4}$$

and E_3 describes the cost of unity in the contour

$$E_3 = \sum_{i,j} \alpha(l_{i,j}^x + l_{i,j}^y). \tag{5}$$

This term is introduced to remove the global minimum of energy (energy would be equal to zero). This minimum may result from application of the previous energy term (4). This occurs when all the elements of the matrix l are qual to 1, and it means that the system tends to introduce only units into this matrix.

The last term E_4 is related to correction of the quality of the contour obtained (the energy increases if any irregularities and unclosed ends of the contour are created) and does not exist for the case of one-dimensional signal [7]

$$E_4 = -\sum_{i,j} \gamma S^x_{i,j} S^y_{i,j} S^x_{i+1,j+1} S^y_{i+1,j+1} \tag{6}$$

where: $S_{i,j} = 2l_{i,j} - 1$ is a bipolar equivalent of a unipolar contour value $l_{i,j}$ and γ is the weight coefficient of this energy term.

Such an expression assumed for the energy of neural-network states describes the architecture and dynamics of this network. In the following, it is applied to define the rules of optimisation of the network.

The method for regularisation with discontinuities can also be part of the system for extracting characteristic information from images in various narrow spectral regions. In this case, the energy of the neural network comprises the states of all images. The terms of energy in the multispectral version take the following forms:

$$E_1 = \varkappa \sum_{i,j} \sum_{n}^{N} [f^n_{i,j} - d^n_{i,j}]^2 \tag{7}$$

is the term responsible for the tendency of the fitted images f^n to respective originals d^n, where: i, j are the coordinates of a pixel, n is the number of the image, and N is the number of images;

$$E_2 = \lambda \sum_{i,j} \sum_{n}^{N} [(f^n_{i+1,j} - f^n_{i,j})^2 (1 - l^x_{i,j}) + (f^n_{i,j+1} - f^n_{i,j})^2 (1 - l^y_{i,j})] \tag{8}$$

is the term smoothing the images in those places where the contour does not occur (*i.e.*, $l^x_{i,j} = 0$, $l^y_{i,j} = 0$);

$$E_3 = \alpha \sum_{i,j} (l^x_{i,j} + l^y_{i,j}) \tag{9}$$

is the term removing the global minimum of energy introduced by E_2 (in the case when all values of $l^x_{i,j}$ and $l^y_{i,j}$ are equal to 1). The form of the fourth term of energy does not undergo changes, because in this method the contour image is generated on the basis of all spectral images and has identical structure of data as for monospectral case.

The possibility of simultaneous usage of a few images to generate the fitted image and the contour will allow the algorithms or the neural networks designed on the basis of the method presented for wide applications in multispectral detection systems.

3.1. Minimisation of energy

The states of fitting and the contour define a point in the space of states $f \times l$. The number of dimensions in this space is equal to the total number of neurons in the layers f and l. The energy E is a function of states in this space and is defined in each point of the space. It may be treated as a potential. The direction of energy decrease is opposite to its gradient vector. This means that the state of a system should be shifted along the direction of the following vector

$$\vec{W} = -\mathrm{grad}_{f\times l} E. \tag{10}$$

The whole operation of energy decrease can be used to define changes of given points of fitting $\Delta f_{i,j}$ and changes of contour $\Delta l_{i,j}$ in a single step

$$\Delta f_{i,j} = -\mathrm{step}\frac{\partial E}{\partial f_{i,j}} \tag{11a}$$

and

$$\Delta l_{i,j} = -\mathrm{step}\frac{\partial E}{\partial l_{i,j}}. \tag{11b}$$

One can write

$$\frac{\partial E}{\partial f_{i,j}} = \sum_{i,j}\{2\varkappa^2(f_{i,j}-d_{i,j})+2\lambda^2[(f_{i,j}-f_{i+1,j})(1-l^x_{i,j})+(f_{i,j}-f_{i,j+1})(1-l^y_{i,j})$$
$$+(f_{i,j}-f_{i-1,j})(1-l^y_{i-1,j-1})+(f_{i,j}-f_{i,j-1})(1-l^x_{i-1,j+1})]\}. \tag{12}$$

Due to two possible states of a single element $l_{i,j}$ in the software simulating the network, the contour is updated in a slightly different way – the change in the value of a point from zero to one or inversely is performed only if the energy decreases.

The optimisation of states of the contour image and the fitted images in the case of contour extraction from images in a few narrow spectral regions is made according to the direction of the energy gradient decrease in relation to their states of individual images. Omitting the term E_4, the derivative of energy over the state of the fitted image takes the form

$$\frac{\partial E}{\partial f^n_{i,j}} = 2\lambda\begin{bmatrix}(f^n_{i,j}-f^n_{i-1,j})(1-l^x_{i-1j})+(f^n_{i,j}-f^n_{i+1,j})(1-l^y_{i,j})+\\(f^n_{i,j}-f^n_{i,j+1})(1-l^x_{i,j})+(f^n_{i,j}-f^n_{i,j-1})(1-l^y_{i,j-1})\end{bmatrix}+2\varkappa(d^n_{i,j}-f^n_{i,j}). \tag{13}$$

The most convenient example for presentation is the one-dimensional case. For $l_i = 1$, the energy becomes

$$E(l_i = 1) = \sum_i[\varkappa^2(f_i-\Gamma(d)_i)^2+\alpha l_i], \tag{14}$$

while for $l_i = 0$, it becomes

$$E(l_i = 0) = \sum_i[\varkappa^2(f_i-\Gamma(d)_i)^2+\lambda^2(\nabla f)^2_i] \tag{15}$$

and the energy difference between the states which differ by the l_i value takes the following form:

$$\Delta E = E(l_i = 0) - E(l_i = 1) = \lambda^2 (\nabla f)_i^2 - \alpha. \tag{16}$$

The magnitude of this energy should decrease after introducing $l_i = 1$, *i.e.*, $\Delta E > 0$. Hence, we obtain the following inequality:

$$0 < \lambda^2 (\nabla f)_i^2 - \alpha, \tag{17}$$

i.e.,

$$\alpha/\lambda^2 < (\nabla f)_i^2. \tag{18}$$

Let us introduce the following substitution:

$$h \equiv \sqrt{\alpha}/\lambda, \tag{19}$$

thus

$$h^2 < (\nabla f)_i^2. \tag{20}$$

The parameter h is the minimum value of the gradient module $|\nabla f_i|$ in the i-th point in which the contour exists. In other words, the parameter h is the threshold of the gradient module $|\nabla f_i|$ for introducing the unity in l_i which indicates that contour exists in that place. In the cellular neural network, the parameter h describes the threshold of the function of transfer of neurons from the layer $l_{i,j}$.

In the following, the results of the analysis of a sample image are presented. The values of individual function parameters have been selected experimentally. The coefficient h is helpful in defining the dependence between α and λ since the gradient threshold is approximately known. The process of obtaining the parameter h can be automated. In this case, h is calculated from the difference between the values of averaged image points in different image areas.

3.2. Temperature noise

A serious threat to the method for contour extraction described is the occurrence of local minima of energy for neural-network states describing inaccurately the contour

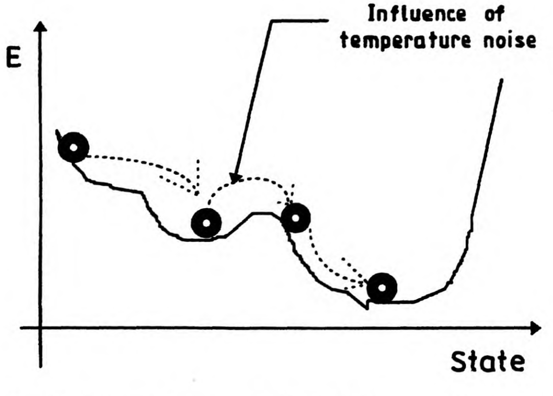

Fig. 2. Schematic presentation of the network state achieving the minimum

location. To reduce the possibility of achieving a local minimum the by network, the idea of temperature noise as a random change of neuron states has been introduced. This idea is illustrated in Fig. 2.

The noise introduced (for the matrix $l_{i,j}$ consists in random (with a certain probability) change in the neuron state. In the case of neurons of the layer $f_{i,j}$, the noise is a random distribution of the value of a neuron in this layer around its former value described by a certain function of probability of the value of this distribution.

It has been found experimentally (by computer simulation) that the introduction of the temperature noise of neurons in the layer $l_{i,j}$, describing the contour location, has a great impact on the contour quality in contradistinction to the neurons in the layer $f_{i,j}$, the values of which describe the regularised image.

Practical application of the method described is presented by the example of the analysis of a highly noised and low-contrast image shown in Fig. 3. In Figure 4, the result of contour extraction with the use of the temperature noise is demonstrated. Figure 5 presents the fitted image (the layer f of the neural network).

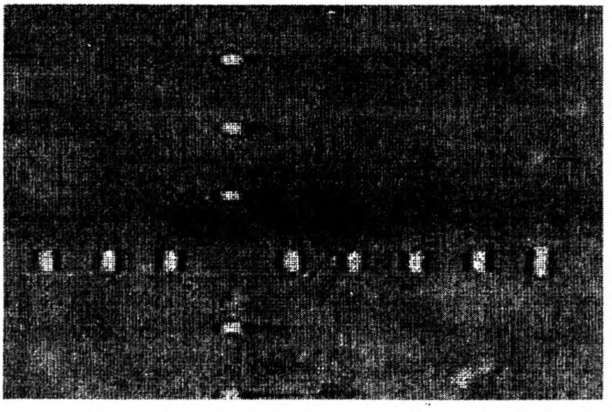

Fig. 3. Noised image

Fig. 4. Contour of image obtained by regularisation with discontinuities supported by the application of the temperature noise

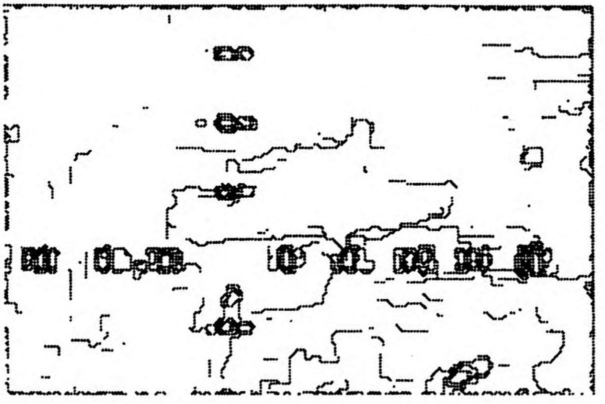

Fig. 5. Fitted image

3.3. Annealing

After introducing the temperature noise, the problem of how to adjust its level appears. A large value of the noise temperature (higher than the level of energy peak) causes nearly free movement of the network in the space of its states. This allows approaching the neighbourhood of the global minimum (corresponding approximately with the ideal contour) by the network state. The high noise value does not allow the network state to remain in the minimum and enforces its continuous movement in the space of states. The network state can be "switched" close to the global minimum by annealing, *i.e.*, continuous temperature decrease. In the neighbourhood of the global minimum, there exist also peaks. For this reason, adjustment of the annealing function is crucial for efficiency of the method in bringing the state to the global minimum.

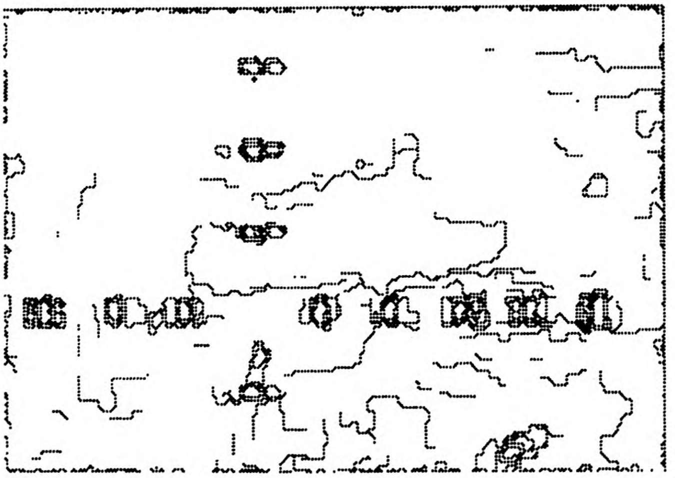

Fig. 6. Effect of regularisation with linear annealing function

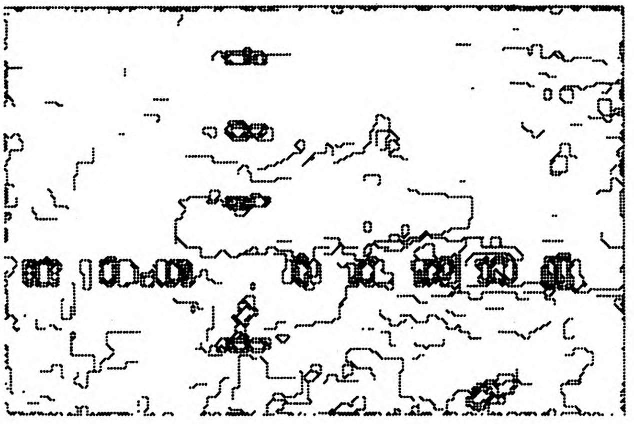

Fig. 7. Effect of regularisation with quadratic annealing function

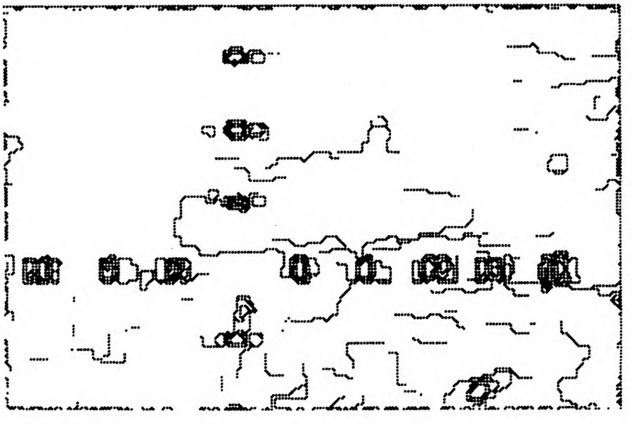

Fig. 8. Effect of regularisation with root annealing function

Figures 6–8 present examples of contours obtained by image regularisation for the time function of temperature decrease (annealing) defined respectively as linear, quadratic and root annealing ones. In all those examples, annealing stand decreasing probability of changing contour matrix element state to an opposite value is observed. As one can see, the choice of annealing function can crucially affect the contour extraction. The annealing function depends on the depth function of local energy minima, the distance from the global minimum, and the dynamics of energy minimisation selected. A more precise analysis of the effect of different annealing functions on regularisation is a separate problem surpassing the scope of this paper

4. Architecture of the cellular neural network

The idea of architecture of the neural network presented earlier should be developed in practice. The neural network structure depends on the neuron model and for

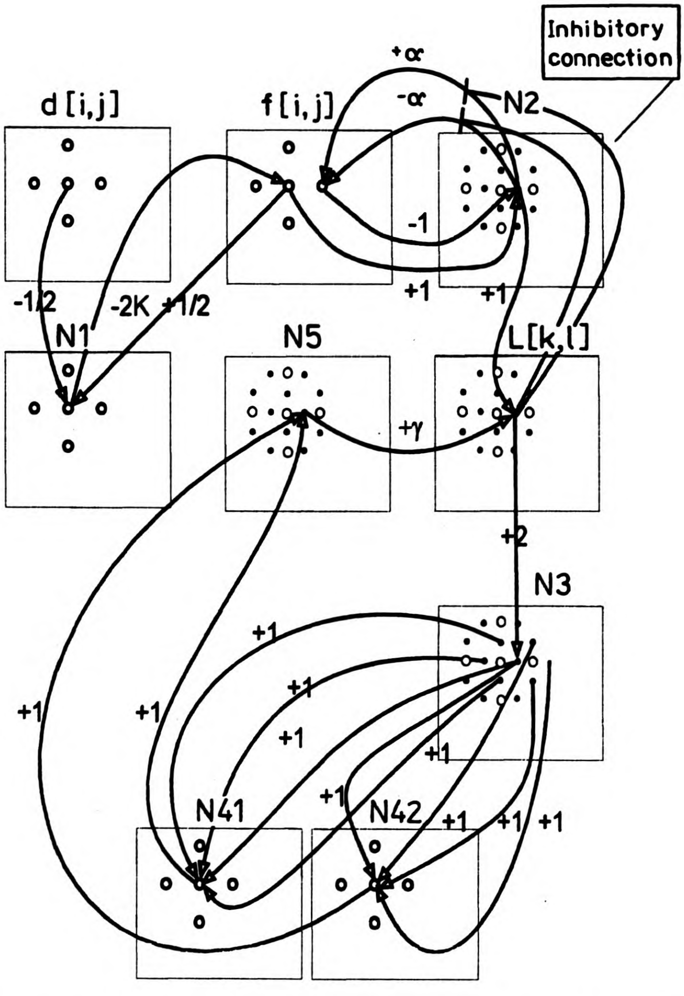

Fig. 9. Block diagram of the cellular neural network accomplishing the method presented. The black dots represent neurons localised where the contour points probably are. The localisations of neurons indicated by light dots correspond to the localisations of pixels in the picture. The bold symbols represent matrices of **f** – fitted date, **d** – image date, N1 – fitted-image differences, N2 – fitted smoothing, N3 – unipolar–bipolar transfer, N41, N42 and N45 – calculate correction of r egularisation

the classical case may have up to fourteen layers. For the neuron model used (see Subsect. 4.1) the neural network implementing the method described here consists of an image data matrix **d[i,j]** and eight layers according to the block diagram presented in Fig. 9.

This figure presents individual neuron layers, sample connections between neurons of different layers, and weights of these connections. Inhibitory (blocking) connections can be replaced with an additional neuron layer with threshold transfer functions of neurons. To make the diagram clear, the connections are presented in their simple form.

The neuron functions in the layers f[i,j](N1, N2) are linear regarding the sum of input enforcements. The neurons in the layers l[i,j] have the threshold function including the product of synapses signals. In turn, the neurons in the layers N3 and N5 react to the sum of enforcements minus 1 and the sum of enforcements minus 2, respectively.

The architecture of the cellular neural network can be expressed also by means of a pattern of interneuron connections and functions of signal transfer of individual neurons belonging to particular layers [12].

4.1. Patterns of interneural connections

The architecture of the cellular neural network accomplishing the method described is presented by the patterns of connections. The patterns of this network have been described in more detail in [13]. The neural network accomplishing the above-mentioned method consists of the following layers: *f*, ***fx*1**, ***fx*2**, ***fy*1**, ***fy*2**, as well as ***lx*** and ***ly***. In the layers ***f***, ***fx*1**, ***fx*2**, ***fy*1**, ***fy*2**, linear transfer functions $f(x) = x$, are applied whereas in the layers ***lx*** and ***ly*** the following threshold function is used:

$$f(x) = \begin{cases} 0 & \text{for} \quad x < 0 \\ 1 & \text{for} \quad x \geqslant 0 \,. \end{cases} \tag{21}$$

All of the neurons applied have construction presented schematically in Fig. 10.

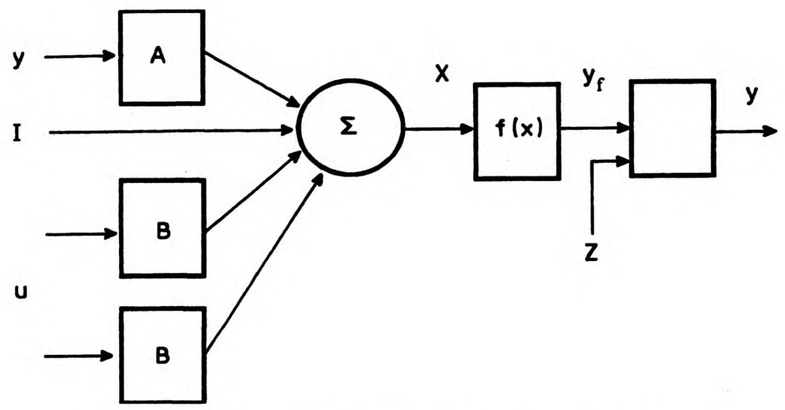

Fig. 10. Schematic of the neuron model, where: I — steering signal from neurons in the given layer, u — steering signal from neurons in other layers. If $Z = 1$ then $y = y_f$, if $Z = 0$ then $y = 0$

The neurons have different transfer functions in different layers. The model of a neuron presented differs from the standard neuron of a neural network in that it has an additional multiplication element at its output. This element has been introduced in order to simplify the architecture of the network in which the problem of discontinuity is being accomplished.

The layer ***f*** layer represents the fitted image. The input signals for this layer are the layers ***fx*1**, ***fx*2**, ***fy*1** and ***fy*2** and the image ***d***. The connections of these layers the layer f are represented by the matrices $\boldsymbol{Bfx}1$, $\boldsymbol{Bfx}2$, $\boldsymbol{Bfy}1$, $\boldsymbol{Bfy}2$ and $\boldsymbol{Bd}$. Local connections inside the layer are represented by the matrix A. The forms of these matrices (patterns of connections) are as follows:

$$A = 1+\text{step}\cdot 2\cdot k^2,$$
$$Bd = -\text{step}\cdot 2\cdot k, \quad Z = 1,$$
$$Bfx1 = \text{step}\cdot 2\cdot \lambda^2,\; Bfx2 = \text{step}\cdot 2\cdot \lambda^2,\; Bfy1 = \text{step}\cdot 2\cdot \lambda^2,\; Bfy2 = \text{step}\cdot 2\cdot \lambda^2. \quad (22)$$

It results from the above relations that they are one-element matrices.

The layers $fx1$, $fx2$, $fy1$, and $fy2$ have been created additionally and they do not represent a specific object by their values. The input signal for these layers is the layer f. The value of Z depends on the layers lx and ly. Let us denote Mx and My as the matrices (patterns) acting locally (in the neighbourhood of a single neuron of the layer) on lx and ly. The value Z can be presented as $Z = 1-Mxlx-Myly$. The forms of the matrices (patterns) of connections for individual layers are given in Tab. 1.

Table 1. Form of the patterns of connections for individual layer

$fx1$	$fx2$	$fy1$	$fy2$
$A=0$	$A=0$	$A=0$	$A=0$
$My=0$	$My=0$	$Mx=0$	$Mx=0$
$Bf=\begin{bmatrix}0&0&0\\0&1&-1\\0&0&0\end{bmatrix}$	$Bf=\begin{bmatrix}0&0&0\\-1&1&1\\0&0&0\end{bmatrix}$	$Bf=\begin{bmatrix}0&0&0\\0&1&0\\0&-1&0\end{bmatrix}$	$Bf=\begin{bmatrix}0&-1&0\\0&1&0\\0&0&0\end{bmatrix}$
$Mx=\begin{bmatrix}0&0&0\\0&1&-1\\0&0&0\end{bmatrix}$	$Mx=\begin{bmatrix}0&0&0\\-1&1&1\\0&0&0\end{bmatrix}$	$Mx=\begin{bmatrix}0&0&0\\0&1&0\\0&-1&0\end{bmatrix}$	$Mx=\begin{bmatrix}0&-1&0\\0&1&0\\0&0&0\end{bmatrix}$

Table 2. Form of the patterns of connections for layers lx and ly

lx	ly
$A=0$	$A=0$
$I=\alpha$	$I=\alpha$
$B=\lambda^2\begin{bmatrix}0&0&0\\0&-1&0\\0&1&0\end{bmatrix}$	$B=\lambda^2\begin{bmatrix}0&0&0\\0&-1&1\\0&0&0\end{bmatrix}$

In the case of the layers lx and ly (Tab. 2), the input signal is the matrix f. Due to simplicity, common serial calculation methods can be more useful here. This can also be solved in a "neural" way.

The last terms in the equations optimising the contour can be accomplished as a simple logic system and given to the input of neurons of the layers lx and ly with the weight 2γ.

5. Summary

In this work, the method for regularisation with discontinuities is presented in a version supported by the application of the temperature noise. This method was used for extraction of contours from highly noised images. The examples presented in figures show relatively high efficiency of the method proposed. The locality of calculations and their simplicity give the possibility of easy implementation in an actual solution of a cellular neural network. This should allow a contour to be extracted in a very short time. The advantages of the method presented (the simplicity of the technical implementation and the efficiency regarding noised images) suggest its application in devices of automated image recognition operating in real time.

Acknowledgements — The work was performed within the research project No. 8T11C 036 12 sponsored by the State Committee for Scientific Research (KBN) in the years 1997–1998.

References

[1] Tadeusiewicz R., *Systemy wizyjne robotów przemysłowych*, (in Polish), [Ed.] WNT, Warszawa 1992.
[2] Hu N., Yu K., Hsu Y., Opt. Eng. **36** (1997), 2828.
[3] Stankiewicz A., Merta I., Jaroszewicz L. R., [in] Int. Workshop on Cellular Neural Network and Their Applications, CNNA'96, Seville, Spain, 1996, p. 207–211.
[4] Canny J., IEEE Trans. Pattern Anal. Machine Intel. **8** (1986), 679.
[5] Geman S., Geman D., IEEE Trans. PAMI **6** (1984), 721.
[6] Hertz J. A., Phys. Scripta **39** (1989), 161.
[7] Gee A., Doria D., Proc. SPIE **1709** (1992), 304.
[8] Stankiewicz A., Merta I., Jaroszewicz L. R., [in] XI School of Optoelectronics, Ustroń 1996, pp. 125–131, (in Polish).
[9] Petrou M., Papachristou P., Kittler J., Opt. Eng. **36** (1997), 2835.
[10] Wong H., Guan L., Opt. Eng. **36** (1997), 3297.
[11] Aizenberg I. N., J. Electron. Imaging **6** (1997), 272.
[12] Kacprzak T., Ślot K., *Sieci neuronowe komórkowe*, (in Polish), [Ed.] WNT, Warszawa 1995.
[13] Stankiewicz A., Niedziela T., Research Works of the Institute of Air Force Technology, (in Polish), the paper being refereed.

Received May 4, 1998
in revised form August 8, 1998