İstanbul Yerbilimleri Dergisi, C.26, S.1, SS. 53-64, Y. 2013

TRAINING OF CELLULAR NEURAL NETWORKS AND APPLICATION TO GEOPHYSICS

HÜCRESEL SİNİR AĞLARININ EĞİTİMİ VE JEOFİZİK UYGULAMASI

Davut Aydogan

Istanbul Üniversitesi, Mühendislik Fakültesi, Jeofizik Mühendisliği Bölümü, 34320, Avcılar, Istanbul

ABSTRACT

In this study, to determine horizontal location of subtle boundaries in the gravity anomaly maps, an image processing method known as Cellular Neural Networks (CNN) is used. The method is a stochastic image processing method based on close neighborhood relationship of the cells and optimization of A, B and I matrices known as cloning templates. Template coefficients of continuous-time cellular neural networks (CTCNN) and discrete-time cellular neural networks (DTCNN) in determining bodies and edges are calculated by particle swarm optimization (PSO) algorithm.

In the first step, the CNN template coefficients are calculated. In the second step, DTCNN and CTCNN outputs are visually evaluated and the results are compared with each other. The method is tested on Bouguer anomaly map of Salt Lake and its surroundings in Turkey. Results obtained from the Blakely and Simpson algorithm are compared with the outputs of the proposed method and the consistence between them is examined. The cases demonstrate that CNN models can be used in visual evaluation of gravity anomalies.

Key words: Cellular Neural Network, Particle Swarm Optimization, Templates, Gravity Anomalies, Lineaments.

ÖZ

Bu çalışmada, gravite anomali haritalarında gözle görülmeyen sınır ve uzanımları tanımlamak için Hücresel Sinir Ağları (HSA) olarak bilinen bir görüntü işleme tekniği kullanılmıştır. Yöntem, şablon katsayıları olarak tanımlanan A, B ve I matrislerinin en iyileme ve komşu hücre ilişkilerine dayandırılmış stokastik bir görüntü işleme tekniğidir. Kütle ve kenarların saptanması için sürekli (SCNN) ve ayrık (ACNN) hücresel sinir ağlarına ait şablon katsayıları Parçacık Sürüsü Optimizasyon (PSO) algoritması kullanılarak hesaplanmıştır.

İlk etapta, Hücresel Sinir Ağı (HSA) şablon katsayıları hesaplanmıştır. İkinci etapta, ACNN ve SCNN çıkışları kalitatif olarak değerlendirilmiş ve sonuçlar birbirleri ile kıyaslanmıştır. Yöntem Türkiyede Tuz Gölü ve çevresine ait Bouguer anomali haritası üzerinde test edilmiştir. Önerilen yöntem çıkışları Blakely ve Simpson algoritmasından elde edilen sonuçlar ile kıyaslanmış ve aralarındaki uyumluluk incelenmiştir. Olgular, hücresel sinir ağ modellerinin jeofizik anomalilerin kaynak çizgiselliklerinin değerlendirilmesinde kullanılabileceğini göstermektedir.

Anahtar Kelimeler: Hücresel Sinir Ağ, Parçacık Sürüsü Optimizasyonu, Şablonlar, Gravite Anomalileri, Çizgisellikler.

INTRODUCTION

Image enhancement methods are widely used in the interpretation of geophysical data. Due to their local dependency, CNNs have found a significant place in image processing applications. Dynamic behavior of CNN networks can be controlled through 19 parameters defined as template coefficients. The literature includes many algorithms for the calculation of these parameters. The trial and error method is based on the experiences of the developer. Another method is to use the template coefficients previously calculated with different algorithms for different purposes in the literature (Aydogan, 2012). Yet another method preferred for obtaining stable template coefficients is to calculate the parameter values with the use of optimization algorithms such as genetic algorithm (GA), particle swarm optimization (PSO) and recurrent perception learning algorithm (RPLA). CNN is widely used in image processing because of being created by space invariant connection geometry (Aydogan et al. 2005; Aydogan, 2007; Guzelis, 1992; Dogaru, 2008; Basturk and Gunay, 2009).

Since potential field data are an indicator of buried fault, contacts and edges of the basin and uplift, and other tectonic lineaments, source boundaries can be mapped to enhance the lineaments in accordance with gravity and magnetic anomalies. In the context of image processing techniques, different algorithms based on gradients have been developed by many authors for automatic extraction of lineaments. Radon transform was used for automatic interpretation of lineaments from potential field anomalies by Zhang et al. (2006). A model was developed utilizing intensity, texture and shape descriptors by Buckingham et al., (2003 a, b) for content base magnetic image retrieval (CBMIR). Blakely and Simpson (1986) developed classical method which is known as boundary analysis and they used gradient to highlight lineaments in gravity and magnetic data. Thurston and Smith (1997), Verduzco et al. (2004) and Cooper and Cowan (2006, 2007) used filters based on local phase, inclination, slope gradient and theta map as edge detectors in interpretation of potential field.

In this study, a CNN model based on PSO algorithm is adopted for template coefficients that can be used in determining the bodies and edges in geophysical images. The CNN model is tested on the Bouguer anomaly map of Salt Lake and its surroundings.

CNN IN IMAGE PROCESSING

CNN is a relatively new circuit structure for image processing, pattern recognition and other important applications in the context of neural network. A cell is the basic unit of a CNN. Space invariant of the process applied to each image pixel is the convenient feature of CNN for image processing. Local connectivity is the most important characteristic of CNN. Each pixel in the image corresponds to a cell in a CNN. A CNN is a network structure processed by nonlinear dynamic systems called cell by using three independent factors called input, bias and initial state. A standard CNN structure is depicted in Fig. 1a. Cells are placed in the cartesian coordinate system. In structure, a cell on the row ***i*** and column ***j*** are expressed as **C**(*i,j*) (*i=1, 2, ... M, j=1, 2, ... N*).

The sphere of influence with radius ***r*** of a **C**(*i,j*) CNN cell is determined with neighborhood term described below (Chua and Yang 1988a,b; Chua and Roska 2002):

$$S_r(i,j)=\left\{C(k,l)\Big|_{\substack{1\le k\le M,\\ 1\le l\le N}} \max\left(|k-i|,|l-j|\le r\right)\right\}, \tag{1}$$

As shown in Figure 1a, the structure of CNN defines as a 2D array of MxN identical cell arranged in rectangular grid (*i=1, 2 ..., M* and *j=1, 2 ..., N*). In image processing, the mathematical expression of the state equation of every **C**(*i,j*) cell depicted in Fig. 1b is defined as independent from circuit theory notation as follows (Chua and Yang 1988a,b). It is

$$\dot{x}_{ij}=\frac{dx_{ij}}{dt}=-x_{ij}+\sum_{C(k,l)\in S_r(i,j)}A(i,j;k,l)y_{kl}+\sum_{C(k,l)\in S_r(i,j)}B(i,j;k,l)u_{kl}+I_{ij} \tag{2}$$

In this place, subscripts (*i,j*) define cell location and (*k,l*) define neighborhoods, $\boldsymbol{x}_{ij}$ is called the state of **C**(*i,j*) cell, $\boldsymbol{y}_{kl}$ is the output of cells, $\boldsymbol{u}_{kl}$ is the input of cells. $\boldsymbol{I}_{ij}$ is the value of bias. ***A***(*i,j;k,l*) is feedback operator or feedback cloning template. ***B***(*i,j;k,l*) is input control operator.

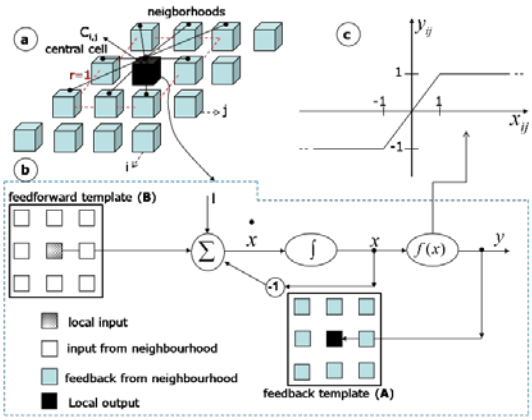

Figure 1. *CNN structure. a) 2D CNN array on a square grid, r=1 (3x3 square neighborhood), Local connection structure of cell, r, shows the neighboring cells around the central cell. The rows and columns of the cells in the model are presented with i and j, respectively. b) Kernel scheme of CNN Structure in the image processing, c) Piecewise linear function response (PWL).*

Şekil 1. CNN yapısı. a) Bir kare grid üzerinde 2B CNN dizisi, r=1 (3x3 kare komşuluk), Hücrenin yerel bağlantı yapısı, r, merkez hücre çevresinde komşu hücreleri göstermektedir. Modeldeki satır ve sütunlar, sırası ile, i ve j ile gösterilmiştir. b) Görüntü işlemede CNN yapısının çekirdek şeması, c) Parça parça doğrusal fonksiyon cevabı (PWL).

The output function in Eq. 2 is given by Chua and Yang (1988a, b);

$$y_{ij} = f(x_{ij}) = 0.5\left\{\left|x_{ij}+1\right| - \left|x_{ij}-1\right|\right\}. \qquad (3)$$

Most of the CNN applications only use space invariant standard CNNs with 3x3 neighborhood, and sphere influence ***r=1***. In this way, Eq. 1 can be denoted as follows:

$$\dot{x}_{ij} = -x_{ij} + \sum_{k=-r}^{r}\sum_{l=-r}^{r} A_{kl}\, y_{i+k,j+l} + \sum_{k=-r}^{r}\sum_{l=-r}^{r} B_{kl}\, u_{i+k,j+l} + I_{ij} \qquad (4)$$

In view of space invariance, the contribution came from feedback operator ***A***(*i,j;k,l*) can be written as

$$\sum_{C(k,l)\in S_r(i,j)} A(i,j;k,l)\,y_{kl} = \sum_{k=-r}^{r}\sum_{l=-r}^{r} a_{k,l}\, y_{i+k,j+l}$$

$$\overset{\Delta}{=} \begin{bmatrix} a_{-1,-1} & a_{-1,0} & a_{-1,1} \\ a_{-0,-1} & a_{-0,0} & a_{-0,1} \\ a_{1,-1} & a_{1,0} & a_{1,1} \end{bmatrix} \otimes \begin{bmatrix} y_{i-1,j-1} & y_{i-1,j} & y_{i-1,j+1} \\ y_{i,j-1} & y_{i,j} & y_{i,j+1} \\ y_{i+1,j-1} & y_{i+1,j} & y_{i+1,j+1} \end{bmatrix}$$

$$= A \otimes Y_{i,j} \qquad (5)$$

(Chua and Roska 2002). The symbol $\otimes$ refers to the sum of the point multiplication. $\boldsymbol{Y}_{ij}$ matrix is the output image. Similar process can be written as follows for ***B***(*i,j;k,l*):

$$\sum_{C(k,l)\in S_r(i,j)} B(i,j;k,l)\,u_{kl} = \sum_{k=-r}^{r}\sum_{l=-r}^{r} b_{k,l}\, u_{i+k,j+l}$$

$$\overset{\Delta}{=} \begin{bmatrix} b_{-1,-1} & b_{-1,0} & b_{-1,1} \\ b_{-0,-1} & b_{-0,0} & b_{-0,1} \\ b_{1,-1} & b_{1,0} & b_{1,1} \end{bmatrix} \otimes \begin{bmatrix} u_{i-1,j-1} & u_{i-1,j} & u_{i-1,j+1} \\ u_{i,j-1} & u_{i,j} & u_{i,j+1} \\ u_{i+1,j-1} & u_{i+1,j} & u_{i+1,j+1} \end{bmatrix}$$

$$= B \otimes U_{i,j} \qquad (6)$$

The contribution came from threshold expressions can be written as $\boldsymbol{I}=\boldsymbol{I}_{i,j}$.

In practice, the pattern of template coefficients that determines the network's dynamic structure operates with **C**(***A***, ***B*** and ***I***) and when this pattern is approximated to an equilibrium state, the result image can be obtained through states or outputs. The image processing algorithm using CNN is carried out the following steps given below:

Step 1: The image, the edges of which will be determined, is normalized and entered into the network.

Step 2: Template coefficients **C**(***A, B* and *I***) are placed in Eq. 4, the CNN state equation transformed into a difference equation by using forward Euler approximation, and the process is continued.

Step 3: The output image is monitored at different ***t*** times from the start until it is stable.

PSO ALGORITHM

Nowadays, many methods inspired by biological systems are used either as training algorithms or in the solution of problem systems. In social systems, which are actually a different kind of biological system, an individual's collective behaviors towards other individuals are examined. This kind of behavior is referred to as swarm intelligence. PSO is a population based stochastic optimization algorithm inspired by the way swarms behave (Kennedy and Eberhart, 1995). In the PSO technique, which is designed for the solution of nonlinear problems, possible solutions are designated as models (particles). At first, the system is initiated with a population of random solutions by randomly positioning each particle in the PSO algorithm into the problem space, and for the best solution particles are updated as iterations and the fittest value within the parameter space is sought. Potential solutions that are denoted as particles in the algorithm move within the problem space by following the best solutions.

The particles within the population have fitness values evaluated by the fitness function to be optimized, and velocities that direct birds in flight. Within the parameter space, particles fly by following the available optimum particles. In the iteration steps, each particle is updated according to two best values. First of these is the best fitness value of each particle in the iteration step and is referred to as the best local value ($\boldsymbol{p}_b$). The second one is the best fitness value of all particles in the population, and is referred to as the single global best value ($\boldsymbol{g}_b$) of the population. In PSO technique, position vector for *n*-particles consisting of *m* parameters (parameter space, parameter size) is

$$\mathrm{X_i} = \left[x_{i1}, x_{i2}, x_{i,3}, ..., x_{in}\right] \quad (i = 1,2,3,...,n) \quad (7)$$

and the velocity vector that indicates the change of each particle at each position is

$$\mathrm{V_i} = \left[v_{i1}, v_{i2}, v_{i,3}, ..., v_{in}\right] \quad (i = 1,2,3,...,n). \quad (8)$$

Although particle size varies with the problem to be optimized, number of particles can usually be selected from 10 to 50. In complex problems higher values can be used. After initially appointing random values within the parameter range for m parameters, $\boldsymbol{p}_b$ and $\boldsymbol{g}_b$ values are obtained according to the values obtained from the fitness function. Velocity and position of the particles within the population are updated with the equations respectively presented below:

$$\mathrm{V_i^{k+1}} = v_i^k + c_1 rand(.)\left[p_{bi}^{\ k} - X_i^k\right] + c_2 rand(.)\left[g_{bi}^{\ k} - X_i^k\right] \quad (9)$$

and

$$X_i^{k+1} = X_i^k + V_i^{k+1}. \quad (10)$$

where *i* shows the number of particles in the population and *k* shows the number of iterations. Generation of random values in the range of (0-1), for each iteration is shown with the rand (.) symbol. $\boldsymbol{c}_1$ and $\boldsymbol{c}_2$ in the equation are constants that can be selected from the range of (0-4) and denoted as the learning factors that direct particles towards local best and the global best positions, respectively. While $\boldsymbol{c}_1$ directs particle movement according to their own individual experiences, the constant $\boldsymbol{c}_2$ directs movement according to the experiences of the particles within the population.

A flow diagram of the method is presented in Fig. 2. As it can also be seen from the Eqs. 4 and 5 above, the PSO technique can be applied in the optimization of nonlinear problems without requiring any derivative information. Since there is a few numbers of parameters to be adjusted for implementing the algorithm, it is quite easy. PSO technique can be successfully applied in many areas of engineering such as function optimization, fuzzy system control and training of artificial neural networks. Some of the recent studies that utilize PSO algorithms in geophysics were conducted by Rahmat, 2003; Shaw and Srivastava, 2007; Sanyi et al., 2009; Martinez et al., 2010; Santos, 2010; Srivastava and Agarwal, 2010; Pekşen et al., 2011.

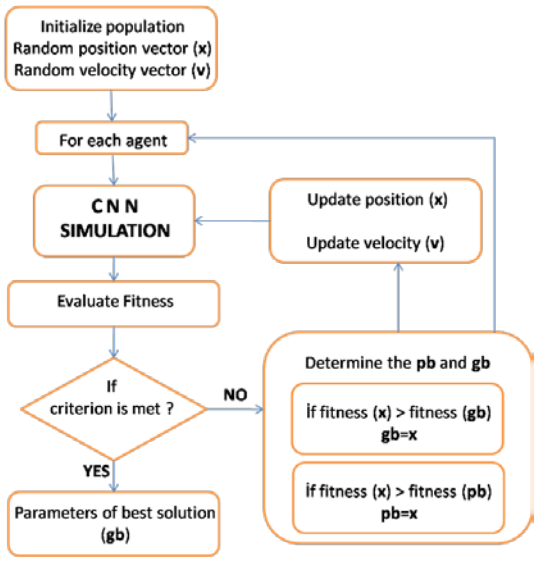

Figure. *2. Flowchart of the PSO algorithm.*
Şekil 2. PSO algoritmasının akış şeması.

PSO BASED TEMPLATE LEARNING

With the purpose of determining subsurface body distributions and boundaries that cause geophysical anomalies, the template coefficients of the developed CNN model are calculated by using the PSO approach. The setup designed in the template coefficient calculation process is presented in Fig. 3. Ten images sets artificially created on binary system at 93 x 136 pixel resolution are used in the training process. These training sets are used as input to the CNN model for determining the objects and their boundaries. Target images are loaded to the model as output. The optimization vector for a single CNN cell consists of 19 real numbers as follows:

$$\eta = (a_1, a_2, a_3, a_4, a_5, a_6, a_7, a_8, a_9, b_1, b_2, b_3, b_4, b_5, b_6, b_7, b_8, b_9, I). \quad (11)$$

In case that the feedback template coefficients are symmetrically selected according to the positional invariance assumption, stable template coefficients can be obtained.

$$a_1 = a_2 = a_3 = a_4 = a_6 = a_7 = a_8 = a_9 = 0, a_5 = a$$
$$b_1 = b_2 = b_3 = b_4 = b_6 = b_7 = b_8 = b_9 = b, b_5 = b_0$$

With these assumptions, the optimization vector in DTCNN case is

$$\eta_{DTCNN} = (a, b_0, b, I). \quad (12)$$

In the case of CTCNN, with the addition of the time interval DT the following equation is obtained:

$$\eta_{CTCNN} = (a, b_0, b, I, DT). \quad (13)$$

In the training process, population size is taken equal with the number of iterations. The initial values of the position and velocity vectors developed according to Eqs. 12 and 13 are selected randomly. Training factors of PSO, c_1 and c_2 parameter values that will change in iteration stages are taken as c_{1max} =2.5, c_{1min} =.5, c_{2max} =2.5 and c_{2min} =.5. In general terms, training of CNN template coefficients can be defined as an error minimization problem. In the cases of DTCNN and CTCNN, the objective or cost function to be minimized can be respectively denoted as follows:

$$\min_{A,B,I} E = (A, B, I),$$
$$\min_{A,B,I,DT} E = (A, B, I, DT). \quad (14)$$

In the present study, for the DTCNN model the objective function is denoted as follows

$$E_{DTCNN} = \sqrt{\sum_{i=1}^{n}\sum_{j=1}^{n}\left(Y_{i,j}(A,B,I) - Y_{i,j}\right)^2 / n}, \quad (15)$$

while for the CTCNN model it is denoted as follows:

$$E_{CTCNN} = \sqrt{\sum_{i=1}^{n}\sum_{j=1}^{n}\left(Y_{i,j}(A,B,I,DT) - Y_{i,j}\right)^2 / n}. \quad (16)$$

Here, while $\boldsymbol{Y}_{i,j}$ signifies the actual output, $\boldsymbol{Y}_{i,j}(\boldsymbol{A,B,I})$ or $\boldsymbol{Y}_{i,j}(\boldsymbol{A,B,I,DT})$ denotes the CNN model output (target). In Eqs. 15 and 16, n signifies the total pixel number of each image of the training sets.

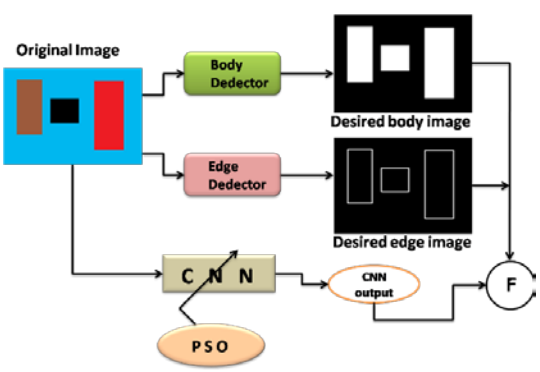

Figure. *3. Template learning using PSO.*
Şekil 3. PSO kullanılarak şablon eğitimi.

PSO parameter values calculated in the training process with population groups of 10, 20, 30, ..., 100 are shown in Tables 1 and 2 for DTCNN and in Tables 3 and 4 for CTCNN. The template coefficients obtained by using these values and the setup in Fig. 3 are presented in Figs. 4 and 5 for DTCNN and Figs. 6 and 7 for CTCNN. While the Local cost values presented in the tables signify the minimum cost values obtained for each population, the Global cost and Global Avg. cost include maximum and average cost values, respectively. *DT* symbolizes the time interval calculated for the CTCNN model. It is observed that the template coefficients obtained for the population where minimum Local cost values and maximum Global cost values are included produce successful results. It is also observed that the Global Avg. cost values remained within the range of average values. In application stages it is observed that after the results are obtained in consequence of a few iterations, there is no change in the output image no matter how much iteration is made and a stable status is established.

Table 1. *PSO parameters for the template calculation in the DTCNN_BODY process. Local cost, Global cost and Global Avg. cost respectively represent the minimum, maximum and average values of the objective function on different populations. Itera: is the number of iterations where the objective function reaches the minimum value. Popsize: is the population number used in the PSO process.*

Çizelge 1. DTCNN_BODY işleminde şablon hesaplaması için PSO parametreleri. Local cost, Global cost and Global Avg.cost, sırası ile, amaç fonksiyonunun farklı popülasyonlardaki minimum, maksimum ve ortalama değerlerini simgelemektedir. Itera: amaç fonksiyonunun minimum değere ulaştığı andaki yineleme sayısıdır. Popsize: PSO sürecinde kullanılan popülasyon sayısıdır.

Popsize	Itera	Global cost	Global Avg.cost	Local cost
10	5	101,5066	53,6691	8,6843
20	5	163,4435	33,5719	2,3770
30	6	163,4435	42,9485	3,7488
40	8	163,4435	75,0754	3,5609
50	**10**	**163,4435**	**67,9850**	**2,4354E-16**
60	5	163,4435	65,5706	3,5411
70	5	163,4435	73,4538	6,4333E-15
80	4	163,4435	81,8513	1,4558
90	5	163,4435	78,2097	2,2006E-15
100	3	163,4435	81,4034	2,1650

Table 2. *PSO parameters for the template calculation in the DTCNN_EDGE process. Local cost, Global cost and Global Avg. cost respectively represent the minimum, maximum and average values of the objective function on different populations. Itera: is the number of iterations where the objective function reaches the minimum value. Popsize: is the population number used in the PSO process.*

Çizelge 2. DTCNN_EDGE işleminde şablon hesaplaması için PSO parametreleri. *Local cost, Global cost* and *Global Avg.cost,* sırası ile, amaç fonksiyonunun farklı popülasyonlardaki minimum, maksimum ve ortalama değerlerini simgelemektedir. *Itera:* amaç fonksiyonunun minimum değere ulaştığı andaki yineleme sayısıdır. *Popsize:* PSO sürecinde kullanılan popülasyon sayısıdır.

Popsize	Itera	Global cost	Global Avg.cost	Local cost
10	9	87,9858	38,9845	2,6620
20	14	217,6336	58,7524	0,1508
30	30	217,6336	60,8328	0,1217
40	15	217,6336	93,8327	0,0957
50	10	217,6336	114,1818	0,1706
60	56	217,6336	112,9012	0,0776
70	**13**	**217,6336**	**109,1445**	**0,0270**
80	15	217,6336	119,0905	0,1054
90	83	217,6336	112,8112	0,0828
100	44	217,6336	114,6899	0,0760

Table 3. *PSO parameters for the template calculation in the CTCNN_BODY process. Local cost, Global cost and Global Avg. cost respectively represent the minimum, maximum and average values of the objective function on different populations. Itera: is the number of iterations where the objective function reaches the minimum value. Popsize: is the population number used in the PSO process. DT: is the time interval calculated in the case of CTCNN.*

Çizelge 3. CTCNN_BODY işleminde şablon hesaplaması için PSO parametreleri. *Local cost, Global cost* and *Global Avg.cost,* sırası ile, amaç fonksiyonunun farklı popülasyonlardaki minimum, maksimum ve ortalama değerlerini simgelemektedir. *Itera:* amaç fonksiyonunun minimum değere ulaştığı andaki yineleme sayısıdır. *Popsize:* PSO sürecinde kullanılan popülasyon sayısıdır. *DT*: CTCNN durumunda hesaplanan zaman aralığı.

Popsize	Itera	DT	Global cost	Global Avg.cost	Local cost
10	9	14,3483	160,2269	93,5252	5,3155
20	6	4,4987	163,4435	69,0760	5,1305
30	6	3,8866	163,4435	55,4032	5,1278
40	4	0.8689	163,4435	67,5293	2,8438
50	5	6,1406	163,4436	71,3255	3,5108
60	5	7,5475	163,4436	76,2339	4,8401
70	5	2,0923	163,4435	75,3767	4,9492
80	**5**	**6,7698**	**163,4435**	**86,3000**	**2,5526**
90	4	3,8315	163,4435	76,4662	5,2933
100	6	2,9130	172,6474	77,3387	3,6635

Table 4. *PSO parameters for the template calculation in the CTCNN_EDGE process. Local cost, Global cost and Global Avg. cost respectively represent the minimum, maximum and average values of the objective function on different populations. Itera: is the number of iterations where the objective function reaches the minimum value. Popsize: is the population number used in the PSO process. DT: is the time interval calculated in the case of CTCNN.*

Çizelge 4. CTCNN_EDGE işleminde şablon hesaplaması için PSO parametreleri. Local cost, Global cost and Global Avg.cost, sırası ile, amaç fonksiyonunun farklı popülasyonlardaki minimum, maksimum ve ortalama değerlerini simgelemektedir. Itera: amaç fonksiyonunun minimum değere ulaştığı andaki yineleme sayısıdır. Popsize: PSO sürecinde kullanılan popülasyon sayısıdır. DT: CTCNN durumunda hesaplanan zaman aralığı.

Popsize	Itera	DT	Global cost	Global Avg.cost	Local cost
10	8	3,3134	202,5319	85,3688	0,1062
20	12	5,7456	217,6336	54,8831	0,0466
30	**3**	**6,9565**	**217,6336**	**84,0818**	**0,0053**
40	17	3,4286	217,6336	92,8216	0,0753
50	3	3,8028	217,6336	91,6328	0,0382
60	54	0,8504	217,6336	110,4090	0,0079
70	10	3,5874	221,0650	115,3889	0,0229
80	5	4,5622	230,7001	113,8314	0,0083
90	17	5,9903	217,6336	117,7345	0,0162
100	14	2,5123	218,3352	115,8650	0,0305

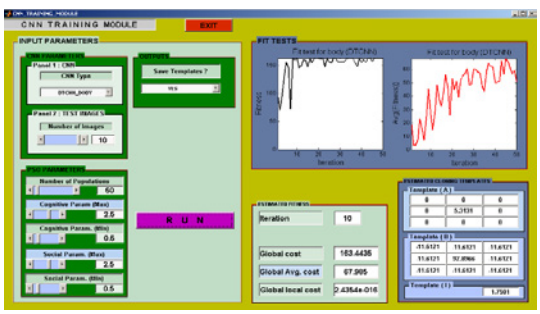

Figure. 4. *Graphical User Interface of the DTCNN_BODY operation for the template calculations.*

Şekil 4. Şablon hesaplamaları için DTCNN_BODY işleminin grafiksel kullanıcı arayüzeyi.

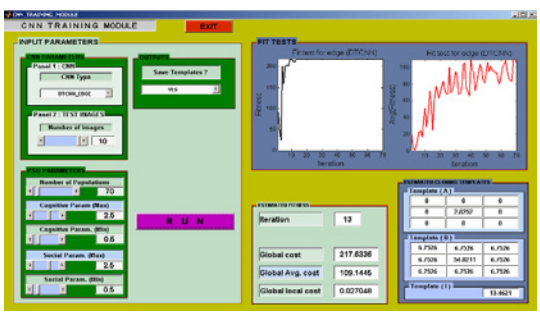

Figure. 5. *Graphical User Interface of the DTCNN_EDGE operation for the template calculations.*

Şekil 5. Şablon hesaplamaları için DTCNN_EDGE işleminin grafiksel kullanıcı arayüzeyi.

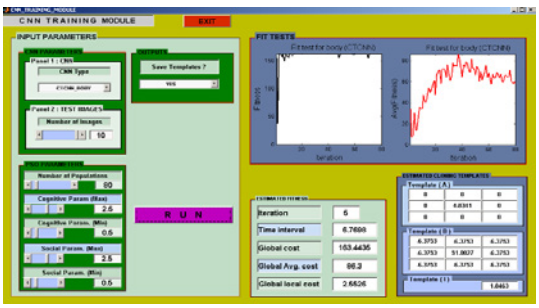

Figure. 6. *Graphical User Interface of the CTCNN_BODY operation for the template calculations.*

Şekil 6. Şablon hesaplamaları için CTCNN_BODY işleminin grafiksel kullanıcı arayüzeyi.

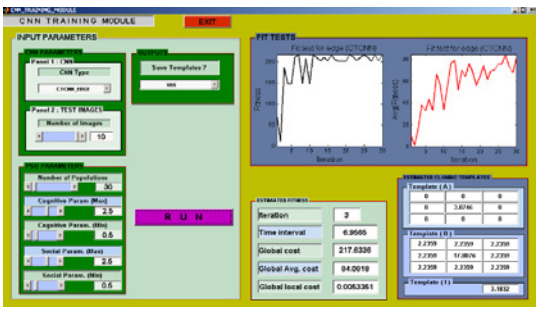

Figure. 7. *Graphical User Interface of the CTCNN_EDGE operation for the template calculations.*

Şekil 7. Şablon hesaplamaları için CTCNN_EDGE işleminin grafiksel kullanıcı arayüzeyi.

APPLICATION TO FIELD EXAMPLE

In this section, to test the reliability of the proposed CNN model approach, the PSOCNNPOT-TOOL program developed in Matlab environment is used for determining the subsurface body distribution and boundaries that cause the Bouguer anomalies of the Salt Lake and its surroundings, and some practical findings are presented and briefly discussed.

Most of the surface area of the Salt Lake basin is covered by largely undeformed Neogene to Quaternary strata that bury many of the important pre-Neogene structures (Çemen et al. 1999). However, several northwest striking faults are present as prominent topographic and structural features (Fig. 8). In addition, there is a channel shaped basin to the west of the Salt Lake, connecting the Salt Lake and Haymana basins (Aydemir and Ateş 2008). A northwest-striking fault zone is the most prominent structural feature along the eastern margin of the Salt Lake basin. Arpat and Şaroglu (1975) have named this the Tuzgolu fault zone. This fault is about 200 km long and can be divided into a north-

ern and southern part at the town of Aksaray. In the western part of the basin, two parallel fault zones extend from Yeniceoba and Cihanbeyli northwestward. The northernmost of the two faults is called the Yeniceoba fault zone and the southern one is named the Cihanbeyli fault zone.

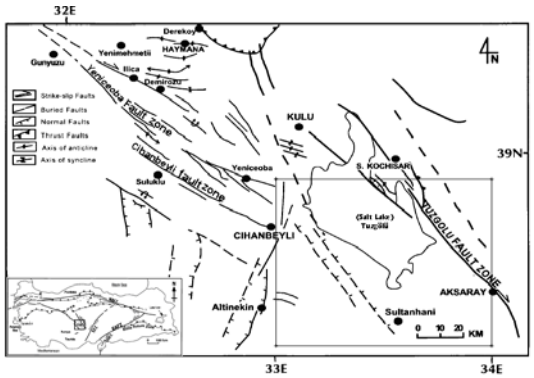

Figure. 8. *Tectonic and location map of the study area (simplified from Çemen et al., 1999).*
Şekil 8. Çalışma alanının tektonik ve yer bulduru haritası (Çemen ve diğ., 1999'dan basitleştirilmiştir).

In the context of image processing, potential field data are quite different from binary data, and consequently, extraction of discontinuities turns out to be complex for only single CNN. In this study, to obtained necessary CNN model outputs, we propose the extraction process which is compose of two successive steps (Fig. 9). Firstly, the proposed algorithm starts with CNN1. By assuming a CNN cell responsible for an image pixel, to unify the coding of input and output image, the best way is to normalize the pixel values. After normalization process, the outcomes of the normalization must be rounded to the closest integer, to compensate the calculation inaccuracies. Normalized gravity data is named as GRN. The initial state X and input U are equal to the normalized input image (GRN) and gravity data (GR), respectively. The output of the CNN1 is a marker of sources to remain. This procedure is used to detect causative subsurface sources. The cloning templates which are used in the nonlinear CNN1 are capable of only a detection of sources. For DTCNN and CTCNN models, the temple coefficients used in CNN1 are depicted in Fig. 4 and Fig. 6, respectively. After the CNN1 operation reaches a stable state, the resulting image belongs to the subsurface body distribution. Secondly, while the initial state matrix for CNN2 is taken as the same one in CNN1, the resulting output from the first step is assigned to the input of CNN2. To detect the edges, for DTCNN and CTCNN models, the temple coefficients used in CNN2 are illustrated in Fig. 5 and Fig. 7, respectively.

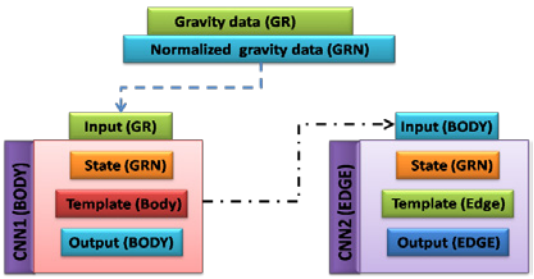

Figure. 9. *Representation of CNN designed for the detection of source and edges from gravity images.*
Şekil 9. Gravite görüntülerinden kaynak ve sınırların saptanması için tasarlanan CNN gösterimi.

In this study, Bouguer gravity anomaly map compiled by Ugurtas (1975) is used. The Bouguer anomaly map covering the Salt Lake basin in the study area is digitized with a sampling interval of 1 km. For detecting the source boundaries and orientations, the Bouguer anomaly map shown in Fig. 10 is normalized to be loaded to the program in DTCNN and CTCNN processes, and converted into grayscale format. The body distribution that causes anomaly is obtained through 8 iterations in the DTCNN_BODY process presented in Fig. 11, and stable results are obtained in 3 iterations for DTCNN_EDGE (Fig. 12), which is used for determining the body boundaries that cause anomalies. A further increase in the iteration value does not affect the stability in the obtained result and no further variation is observed in the output of the model. This choice satisfies the needed accuracy. Such a performance can make CNN approach very suitable for a wide range of image processing tasks. By repeating the same procedure for CTCNN_BODY with 9 iterations and time intervals of 0.5, similar results are obtained as presented in Fig. 13. Stable results are also obtained in this case by decreasing time interval against the increasing iteration value. As it is clear from these CNN outputs, the tracks of many NW-SE-oriented bodies are seen as basic structural features. It is observed that stable results can be achieved with 5 iterations and 0.5 time interval value in the case of CTCNN_EDGE (Fig. 14).

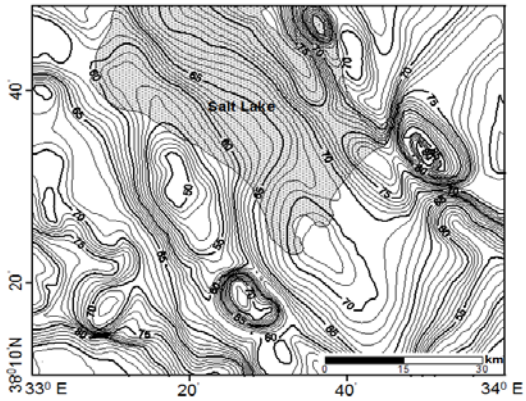

Figure.10. *Bouguer gravity anomaly map of Salt Lake and its vicinity (compiled by Ugurtas (1975)), contour interval is 1 mGal.*

Şekil 10. Tuz Gölü ve çevresine ait Bouguer anomali haritası (Ugurtas (1975) tarafından derlenmiş), kontur aralığı 1 mGal.

The Salt Lake fault zone is the most predominant basic structural feature to the east of the Salt Lake basin. The causative sources and lineaments in the regions with steady changes that are easily spotted in the proposed CNN outputs. Apart from these, we also detect many lineaments produced by small bodies and whose effects are invisible to the eye in the anomaly map. Furthermore, to compare the proposed approach with classical derivative techniques, Blakely and Simpson method (1986) is applied on the same gravity anomaly map and the result is given in Fig. 15. As seen in the figure, the boundaries of the sources with smaller areas in the model could not be exactly detected. In comparison to the results of the proposed CNN approach, the results of CNN could be argued to better represent. A combined look at Figs. 11-14, 15 shows that the obtained results are consistent with the results of previous studies conducted in the same region.

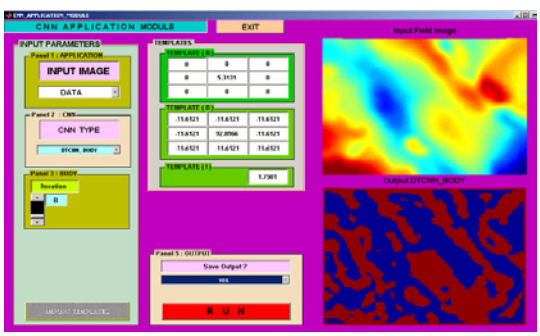

Figure. 11. *DTCNN_BODY output obtained from Bouguer gravity anomaly map of Salt Lake and its surroundings.*

Şekil 11. Tuz Gölü ve çevresine ait Bouguer anomali haritasından elde edilen DTCNN_BODY çıktısı.

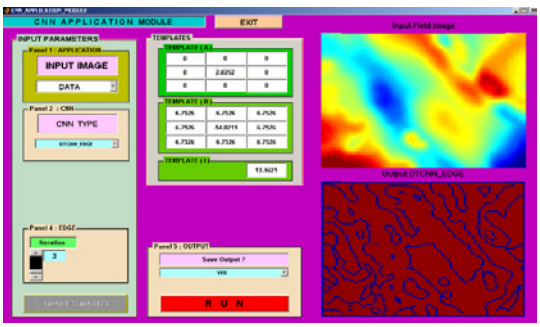

Figure. 12. *DTCNN_EDGE output obtained from Bouguer gravity anomaly map of Salt Lake and its surroundings.*

Şekil 12. Tuz Gölü ve çevresine ait Bouguer anomali haritasından elde edilen DTCNN_EDGE çıktısı.

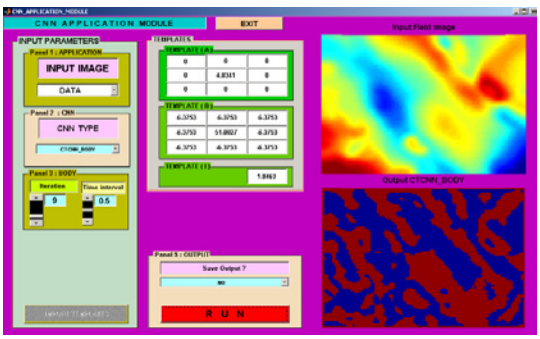

Figure. 13. *CTCNN_BODY output obtained from Bouguer gravity anomaly map of Salt Lake and its surroundings.*

Şekil 13. Tuz Gölü ve çevresine ait Bouguer anomali haritasından elde edilen CTCNN_BODY çıktısı.

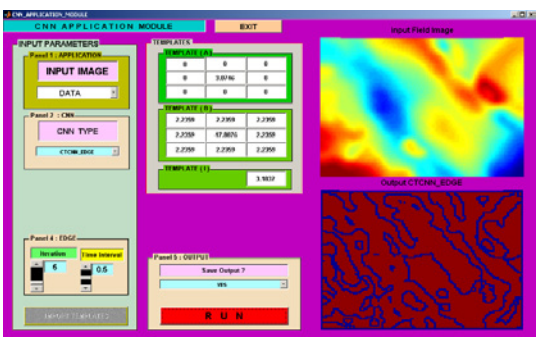

Figure. 14. *CTCNN_EDGE output obtained from Bouguer gravity anomaly map of Salt Lake and its surroundings.*

Şekil 14. Tuz Gölü ve çevresine ait Bouguer anomali haritasından elde edilen CTCNN_EDGE çıktısı.

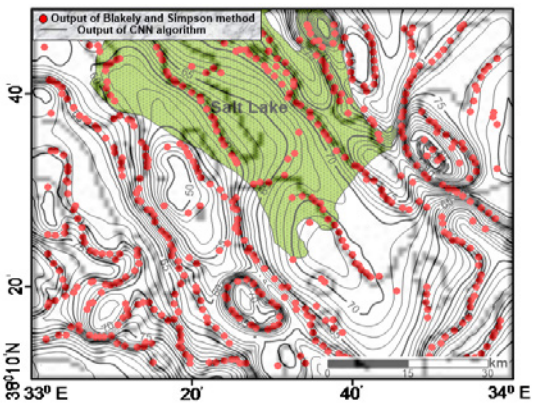

Figure. 15. *Boundary analysis map obtained from Blakely and Simpson method(1986). Lineaments are shown on the Bouguer anomaly map.*

Şekil 15. Blakely ve Simpson(1986) yönteminden elde edilen sınır analizi haritası. Çizgisellikler Bouguer anomali haritasında gösterilmiştir.

CONCLUSIONS

In this study, a CNN model, the template coefficients of which are trained by the global optimization method PSO, is developed for the purpose of determining subsurface bodies that cause geophysical anomalies. The PSO approach is used in calculating the template coefficients to be used in the extraction of the bodies and boundaries for different types of CNN models. In order to obtain stable template coefficients, different populations are included in the scope, and the template coefficients calculated on the basis of the values in the population, where both local minimum values and global maximum values of the objective function are achieved, are used. The feedback and control template play an important role in the CNN application. In terms of the estimation power and the computation time, CNN can be considered to be an effective and powerful tool in solving geophysical problems.

For a comparison of the performance of the proposed CNN, one of the classical methods named as Blakely and Simpson algorithm is reiterated for studies with the same purposes. While source boundaries masked by large bodies cannot be clearly detected in studies using boundary analysis, the proposed CNN approach allows obtaining close-to-reality visible results more effectively. The performance of CNN is quite good. Such a performance can make CNN method very suitable for a wide range of image processing tasks.

Consequently, it could be argued that the proposed approach can be employed as a simple and powerful tool for efficient body and edge detection in normalized digitized gravity images. As in other geophysical methods, the method is a practical and convenient approach that can be used to investigate and detect structural features in large areas.

PROGRAM DESCRIPTIONS

The program PSOCNNPOTTOOL is developed with Matlab 7.12.0(R2011a) on Windows 7 environment. It consists of three graphical user interfaces (GUI) and operation processes are carried out with the PSO CNNPOTTOOL, which includes CNN_TRAINING_ MODULE and CNN_APPLICATION_MODULE interfaces. The template coefficients calculated for DTCNN and CTCNN are obtained with the use of the CNN_TRAINING_MODULE, and the target outputs with the theoretically created image and field image are obtained from the CNN_APPLICATION_MODULE. The program includes 6 sub folders. The first of these, namely cloning template, is the part where the template coefficients calculated with the program and used in CNN models are kept. The second sub folder named as field images is required for the CNN_APPLICATION_ MODULE and is where the field data are kept. First images, as the third sub folder, are required for the images used in the PSOCNNPOTTOOL interface. The fourth sub folder input image is the part that includes the training images used in the training process. Training data sets are automatically loaded to the program. The fifth sub folder is outputs, where the outputs obtained with the use of the CNN_APPLICATION_MODULE interfaces are kept. The last one is the test images sub folder that holds application and test purposed data.

After the program is unzipped, graphical user interface (GUI) program is run with >>PSOCNNPOTTOOL from Matlab command window. According to the purpose, either training or application part is operated. Concerning the CNN model template coefficients, the user has to run the training part of the program for different number of image sets to be prepared at binary scale for both CNN types (DTCNN and CTCNN) and purposes (body and edge). The application part can be used after this.

ÖZET

Bu çalışmada, farklı disiplinlerde yaygın bir kullanım alanına sahip olan Hücresel Sinir Ağları (HSA) algoritması gravite anomalilerine neden olan kütle sınırlarının saptanması amacı ile kullanılmıştır. Önerilen algoritma, görüntü işleme konularında kullanılan stokastik bir yöntem olarak bilinmektedir. Yöntemin temeli, şablon katsayıları olarak bilinen üç farklı matrisin eğitilmesi ilkesine dayanmaktadır. Matrisler, değişik veri setleri üzerinde farklı optimizasyon yöntemleri kullanılarak eğitilebilmektedir. Bu çalışmada, gravite değerlerine neden olan kütle sınırlarının saptanması amacı ile kullanılan şablon katsayıları Parçacık Sürüsü Optimizasyon (PSO) algoritması kullanılarak eğitilmiştir. Bu amaca yönelik olarak Matlab programlama dili kullanılarak bir arayüzey programı hazırlanmıştır. Eğitilen matris katsayıları kuramsal veri setleri üzerinde test edilerek uygulanabilirlikleri araştırılmıştır. Test işlemleri sırasında şablon katsayılarının önemli röl oynadıkları gözlemlenmiş olup sonuçta tatmin edici sonuçlar elde edilmiş ve yöntemin arazi verileri üzerindeki uygulanabilirliğini test etmek amacı ile Tuz Gölü ve çevresine ait Bouguer anomali haritası kullanılmıştır. Kenar saptama işlemlerinde sıklıkla kullanılan ve klasik yöntemlerden olan Blakely ve Simpson algoritması kullanılarak elde edilen sonuçlar önerilen yöntem sonuçları ile kıyaslanmış olup tatmin edici sonuçlar elde edilmiştir. Sonuç olarak, jeofizikte kütle sınırlarının görsel(kalitatif) saptanması ve yorumlanmasında, uygun şablon katsayılarının kullanılması koşulu ile tatminkar sonuçların elde edilebileceği kanısına varılmıştır.

ACKNOWLEDGEMENTS

This work is supported by the Department of Scientific Research Projects of İstanbul University with the number BYP/21676. I thank Prof. Dr. Z. Mümtaz Hisarlı(reviewer) and Assoc. Prof. Dr. Bülent Oruç (reviewer) for their reading and constructive comments on the manuscript. I appreciate very much their help.

REFERENCES

Arpat, E., Şaroglu F., 1975, Türkiye'deki bazi önemli genç tektonik olaylar, Turkish Geological Society Bulletin 18, 91–101.

Aydemir, A., Ateş, A., 2008, Determination of hydrocarbon prospective areas in the Tuzgolu (Salt lake) Basin, central Anatolia, by using geophysical data, Journal of Petroleum Science and Engineering 62, 36-44.

Aydogan, D., 2007, Processing the Bouguer anomaly map of Biga and the surrounding area by the Cellular Neural Network: Application to the southwestern Marmara region, Earth Planets Space 59, 201-208.

Aydogan, D., 2012, CNNEDGEPOT: CNN based edge detection of 2D near surface potential field data, Computers&Geosciences 46, 1-8.

Aydogan, D., Elmas, A., Albora, A. M., Ucan, O.N., 2005, A new approach to the structural features of the Aegean sea: Cellular Neural Network, Marine Geophysical Researches 26, 1-15.

Basturk, A., Gunay, E., 2009, Efficient edge detection in digital images using a cellular neural network optimized by differential evolution algorithm, Expert Systems with Applications 36, 2645–2650.

Blakely, R. J., Simpson, R. W., 1986, Approximate edges of source bodies from magnetic or gravity anomalies, Geophysics 51, 1494–1498.

Buckingham, A. J., Dentith M. C., List, R. D., 2003a, Semi-automated magnetic image retrieval, ASEG Extended Abstracts (2), 1 – 5.

Buckingham, A.J, Dentith, M.C, List, R.D, 2003b, Towards a system for content-based magnetic image retrieval, Exploration Geophysics 34(3), 195 - 206.

Çemen, I., Göncüoglu, M.C., Dirik, K., 1999, Structural evolution of the Tuzgölü basin in Central Anatolia, Turkey, The Journal of Geology 107, 693-706.

Chua, L.O., Yang, L., 1988a, Cellular neural networks: theory, IEEE Transactions on Circuit and Systems 35, 1257-1271.

Chua, L.O., Yang, L., 1988b, Cellular neural networks: applications, IEEE Transactions on Circuit and Systems 35, 1272-1290.

Chua, L.O., Roska, T., 2002, Cellular neural networks and visual computing, Cambridge University press.

Cooper, G.R.J., Cowan, D.R., 2006, Enhancing potential field data using filters based on the local phase, Computers & Geosciences 32, 1585-1591.

Cooper, G.R.J., Cowan, D.R., 2007, Enhancing linear features in image data using horizontal orthogonal gradient ratios, Computers & Geosciences 33, 981–984.

Dogaru, R., 2008, Systematic Design for Emer-

gence in Cellular Nonlinear Networks, Studies in Computational Intelligence 95, Springer.

Guzelis, C., 1992, Supervised learning of the steady-state outputs in generalized cellular networks, Cellular Neural Networks and their Applications, CNNA-92, Second International Workshop, 74-79.

Kennedy, J., Eberhart, R. C., 1995, Particle Swarm Optimization, Proceedings IEEE International Conference on Neural Networks 4, 1942-1948.

Martinez, J.L.F., Gonzalo, E.G., Naudet, V., 2010, Particle swarm optimization applied to solving and appraising the streaming-potential inverse problem, Geophysics 75, 4, WA3–WA15.

Pekşen, E., Yas,T., Kayman, A.Y., Özkan, C., 2011, Application of particle swarm optimization on self-potential data. Journal of Applied Geophysics, 75, 305–318.

Rahmat, S.Y., 2003, Genetic Algorithm (GA) and Particle Swarm Optimization (PSO) in Engineering Electromagnetics, ICECom 2003, 17'th International Conference on Applied Electromagnetics and Communications, 1 - 3 October , Dubrovnik, Croatia.

Santos, F.A.M., 2010, Inversion of self-potential of idealized bodies' anomalies using particle swarm optimization, Computers & Geosciences 36, 1185–1190.

Sanyi, Y., Shangxu, W., Nan, T., 2009, Swarm intelligence optimization and its application in geophysical data inversion, Applied Geophysics 6, 2, 166 – 174.

Shaw, R., Srivastava, S., 2007, Particle swarm optimization: A new tool to invert geophysical data, Geophysics 72, 2, F75–F83.

Srivastava, S., Agarwal, B. N. P., 2010, Inversion of the amplitude of the two-dimensional analytic signal of the magnetic anomaly by the particle swarm optimization technique, Geophys. J. Int. 182, 652–662.

Thurston, J.B., Smith, R.S. 1997, Automatic conversion of magnetic data to depth, dip, and susceptibility contrast using the SPI (TM) method, Geophysics 62, 807–813.

Ugurtas, G. 1975, Geophysical interpretation of part of the Tuz Gölü basin, M.T.A. Bulletin 85, 38-45.

Verduzco, B., Fairhead, J.D., Green, C.M. 2004, New insights into magnetic derivatives for structural mapping, The Leading Edge 23, 2, 116–119.

Zarandy, A., 1999, The art of CNN template design, International Journal of Circuit Theory and Applications 27, 5-23.

Zhan, L., Wu J., Hao T., Wang, J., 2006, Automatic lineament extraction from potential-field images using the Radon transform and gradient calculation, Geophysics 71, j31-j40.