Edge detection for Very High Resolution Satellite Imagery based on Cellular Neural Network

Juan Manuel Núñez

Centro de Investigación en Geografía y Geomática "Ing. Jorge L. Tamayo", D.F., México
`jnunez@centrogeo.org.mx`

Abstract. In the context of Very High Resolution (VHR) satellite imagery, edge detection is one of the most important and difficult steps in image processing and pattern recognition. This paper presents the use of Cellular Neural Network (CNN) in edge detection and shows its capacity to locate and identify discontinuities in the gray levels of the objects that appear in VHR image. The results show that the characteristics of the CNN, in terms of its local connectivity, are those that allow greater extraction of continuous edges. The multi-layer processing structure design of the CNN allows to identify a definitive edge in urban environment. To evaluate the results, a metric of peak signal to noise ratio (PSNR) has been introduced as a manner to rank the accuracy of the resultant edge determined by the assessed methods. The extracted VHR features with the CNN edge detector include accuracy of edge location and better linking of edge segments.

Keywords: Cellular Neural Network, Very High Resolution satellite imagery, urban environment, edge detection, local connectivity.

1   Introduction

Today an important number of earth observation platforms are equipped with Very High Resolution (VHR) optical imagers with the purpose of increasing the visibility of terrestrial features, urban objects in particular, by reducing spectral heterogeneity per-pixel and thereby improving object segmentation [1, 2].

Successful delineation and segmentation of fine details in VHR satellite imagery, such as objects related the urban environment (buildings, cars, planters, roads crosswalks and individual trees), can be associated with the information content available in spatial, spectral and radiometric resolution in the imagery. The main idea behind edge detection is to find where abrupt changes in the intensity of an image have occurred. Therefore, edge is the connected boundary between two different regions that define different objects. Some of the usual alternatives for edge-detection algorithms in digital images are based on filtering the image through different masks such as Sobel, Prewitt, Roberts and LoG operators [3]. Although these filters work suitably and fast, they have some problems such as discontinuity in the edges, because they are very sensitive to noise which affects the images, by registering unwanted and

isolated edges. Other sophisticated methods have been developed in order to avoid this problem and find better solutions, e.g. the Canny edge detector [4, 5].

Due of the above, there is a need to explore and develop other methods, in order to improve the existing results In this paper two edge-detection methods are applied in VHR satellite imagery using cellular neural networks (CNN), to evaluate the property of the local connectivity likely associated with the high conservation of edges in the image pattern recognition. CNN is a new proposal for processing information based exclusively on local interactions. Edge-detection problem using CNN in the context of satellite imagery has presented good results in locating the edge in low spatial resolution images [6]. In a dense urban environment sensed by VHR optical sensors, the edge detection problem is very similar to the task of edge-detection problem in images affected by noise, due to buildings and trees that hide some objects of the scene [7]. The use of automatic edge detectors with a high degree of connectivity and reconstruction of edge segments lines facilitates the processes of lines's vectorization, which are used in managing geospatial databases.

The rest of this paper is organized as follows: first, a brief review the CNN model and the necessary architecture for edge detection in image processing is introduced; second, edge-detection simulations with very high-resolution satellite imagery are proposed. For the first case, we worked with a single-layer CNN application and for the second we used a multilayer CNN application. In both cases, a satellite image WorldView-3 (Mexico City) has been used. The result of choosing templates for edge detection using CNN is evaluated in comparison with other edge-detection algorithms. Finally, there is a discussion about the edge detection using CNN application.

2 Brief Description of CNN Model in Image Processing

Cellular neural networks (CNNs) introduced by [8] are locally connected analog arrays, that process large amounts of information in real time. This architecture is capable to perform time-consuming tasks such as pattern recognition and static or moving image processing [9]. In a CNN model, the basic processing unit is called cell, and its local connectivity is the most important characteristic because each cell is connected exclusively to its neighbors, therefore the adjacent cells that are defined in a neighborhood are the only ones that can interact directly with each other [10].

For a two-dimensional CNN with *MxN* array, the dynamics of each cell can be described by the following state equations:

$$
\begin{cases}
\dot{X}_{ij}(t) = -X_{ij}(t) + \sum_{C(k,l) \in N_r(i,j)} A(i,j;k,l) y_{kl}(t) + \sum_{C(j,l) \in N_r(i,j)} B(i,j;k,l) u_{kl} + I_{ij} \\
Y_{ij}(t) = f(X_{ij}(t)) = \frac{1}{2} \left(\left| X_{ij}(t) + 1 \right| - \left| X_{ij}(t) - 1 \right| \right)
\end{cases}
\tag{1}
$$

where $X_{ij}(t)$, u_{ij} and $Y_{ij}(t)$ are the state, the input and the output of a cell $C(i,j)$ in the grid. The input is static, time independent, while the state and output vary with

time. Matrices $A(i,j;k,l)$ (feedback coefficients) and $B(i,j;k,l)$ (control coefficients) denote the connection templates from cell $C(k,l)$ to cell $C(i,j)$. I_{ij} represent the bias in the grid. The control template represents the coupling coefficients of the cells and it defines entirely the behavior of the CNN model with a given static input $|u_{ij}| \leq 1$ and initial condition $X_{ij}(0) = 0$. The output $Y_{ij}(t)$ is described as piecewise linear equation. A space-invariant standard CNN with 3x3 and neighborhood radius of $r = 1$ is defined uniformly by a string of 19 real numbers, called the cloning template which together with initial condition and static input can determine entirely the dynamical properties of the CNN [11]. Considering the invariant space, the interaction between the feedback operator A and the control operator B with the output and the initial state respectively can be written as follows [12]:

$$\dot{X} = X + A \otimes Y + B \otimes U + I \tag{2}$$

where $\otimes$ refers to the point-by-point multiplication, that in discrete mathematics corresponds to spatial convolution [13]. From this state equation the interconnection pattern is defined by the A and B cloning templates, the bias term I and the initial condition. They all together determine the evolution of a final state of CNN. Since image processing's point of view, these parameters determine the type of operation performed in an input image U, to obtain an output image Y, assuming an initial state of the network $X(0)$. Therefore, in an image processing context, the CNN can be understood as a sort of information processing system with interaction dynamic spatial-temporal, which can transform two-dimensional input images in a two-dimensional output image as well.

The cloning template elements constitute the CNN analog program and its determination is one of the most important problems in image processing studies [14, 15]. Regardless of what method was used to obtain the CNN parameters, it should be clear that each set of cloning template coefficients, along with the initial conditions, determines the type of processing over the input. In this paper we took advantage of existing cloning templates in the literature to solve edge-detection problem in the context of noisy images, in order to develop a CNN application related to VHR satellite imagery edge detection.

3 CNN Simulations for Edge Detection Applications

Two practical cases have been developed. The first is a single-layer CNN application whereas the second is a multilayer CNN application. In both cases a satellite image WorldView-3 (WV-3) (September 2014) for Mexico City with a GSD: 0.314 m (PAN) and 16 bits per pixel (unsigned) radiometric resolution has been used. The proposed method followed next steps: Step 1, for practical simulations MatCNN was used, a CNN toolbox for Matlab [16]. To begin implementing MatCNN in Matlab, a script file was written to define the CNN environment and to initialize multiple variables that determined the configuration of the CNN simulations. Feedback, cloning

template and a threshold value were introduced; and the time step=0.2 and number of iterations=25 were optimized for the simulation. Step 2, after CNN network configuration, the edge detection can be performed. For CNN results, the largest region was detected and the rest of the detected regions were sent to the background using a median filter with a neighborhood of 3-by-3, recovering pertaining information to the edge only. Step 3, in order to validate the results, the CNN edge-detection proposal was compared with other classics edge-detection operators also implemented in the Matlab environment: Prewitt, Roberts, LoG and Canny. To illustrate the performance of the single-layer and multilayer CNN, it was assessed local connectivity's property associated with the high preservation of boundaries in the edge-detection task through several parameters related with the pixel connection neighboring. Finally, the aspect of peak signal to noise ratio (PSNR) was calculated through computer simulations to compare the success of the edge detection.

3.1   Case 1. Single-layer CNN for Linear and Curved Edge Detection

In Chua's review [17], we find information about how the CNN can be chosen to perform many basic image processing operations. The author developed many CNN cloning templates. One of them is the template called Edgegray CNN, which can detect the edges in gray-scale images. The template has the form:

$$A = \begin{bmatrix} 0 & 0 & 0 \\ 0 & a & 0 \\ 0 & 0 & 0 \end{bmatrix}, B = \begin{bmatrix} c & c & c \\ c & b & c \\ c & c & c \end{bmatrix}, I = z \tag{3}$$

where a, b, c and z are real numbers. These parameters are optimized in order to satisfy the performance of the CNN. This is described through its global task and local rules [18]. Some parameters groups have been suggested from the literature, therefore in order to evaluate this first practical case, the chosen template is:

$$\{a, b, c, z\} = \{2, 8, -1, 0.5\} \tag{4}$$

This selection was made to evaluate the next example of edge detection from a VHR satellite image. In Figures 1 and 2, different visual results from the proposed method can be observed, compared with other edge detectors. Figure 1(a) shows an image cut of 100 x 100 pixel-wide image produced by the WV-3 satellite. Figure 1(b) shows the linear truth shape that defines the edge to extract. The program output for linear edge detection using the proposed CNN template is shown in Figure 1(c). The use of a median filter with a 3-by-3 neighborhood, to recover only information related to the edge, is shown in Figure 1(d). Figures 1(e, f, g and h) are the result of using Canny, LoG, Sobel and Prewitt operators applied to the same image, respectively. It notes that it is not possible to obtain a continuous edge, solely several separate segments.
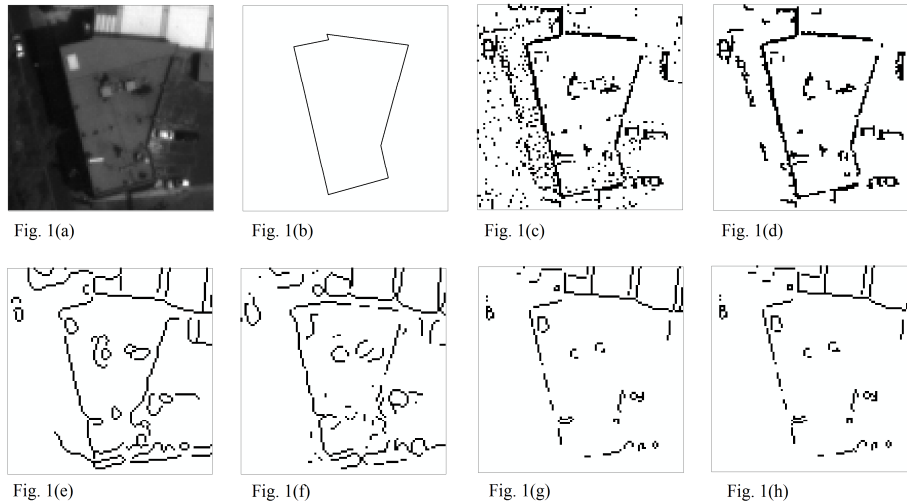
Fig. 1(a) Fig. 1(b) Fig. 1(c) Fig. 1(d)

Fig. 1(e) Fig. 1(f) Fig. 1(g) Fig. 1(h)

Fig. 1. Simulation result for linear edge detection

A second image cut of 100 x 100 pixel-wide image also produced by the same satellite is shown in Figure 2(a). Figure 2(b) shows the curved truth shape that defines the edge to extract. The program output for curved edge detection after using the same CNN template is shown in Figure 2(c). The use of a median filter with a 3-by-3 neighborhood, to recover only information related to the edge, is shown in Figure 2(d). Figures 2(e, f, g and h) give the same outputs for the filters proposed in Figure 1.
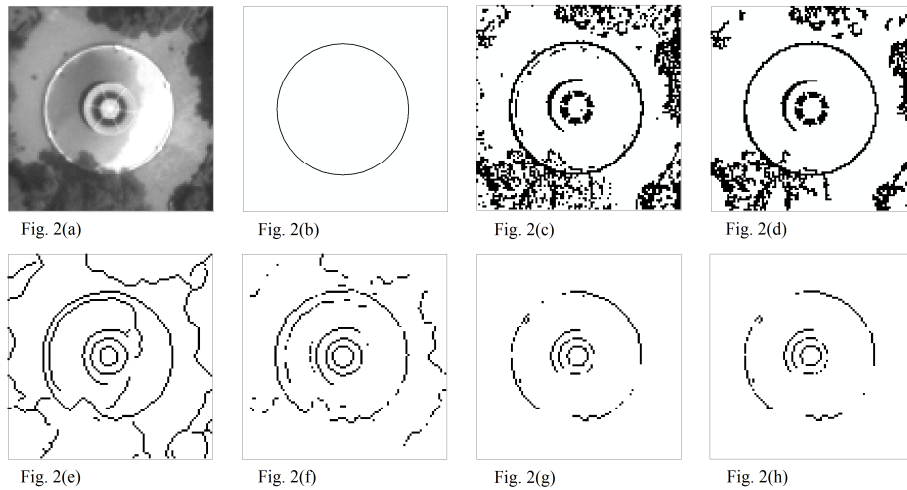

Fig. 2(a) Fig. 2(b) Fig. 2(c) Fig. 2(d)

Fig. 2(e) Fig. 2(f) Fig. 2(g) Fig. 2(h)

Fig. 2. Simulation result for curved edge detection

3.2   Case 2. Multi-layer CNN for Edge Detection

In the preceding exercise, a single-layer CNN was presented. This concept can be extended to multi-layer architecture, the CNN that includes a very broad multi-layer analog array which also has local interconnections [19]. The organization of this CNN is very similar to that of a multi-layer feedforward perceptron [20], although in this case, the state in each layer has continuous-variable dynamics rather than a bistable state. Multi-layer CNN array is an example of the process which is feasible through a building block approach. The example shown is a 3-layers CNN information processing system, that combines diffusion layers, threshold and edge detection. The CNN cloning templates are given in Table 1.

Table 1. Diffusion, threshold and edge detection templates [21]

Layer	A	B	I
Diffusion	$\begin{bmatrix} 0.25 & 0.5 & 0.25 \\ 0.5 & -2 & 0.5 \\ 0.25 & 0.5 & 0.25 \end{bmatrix}$	$\begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}$	0
Threshold	$\begin{bmatrix} 0 & 0 & 0 \\ 0 & 2 & 0 \\ 0 & 0 & 0 \end{bmatrix}$	$\begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}$	0
Edge detection	$\begin{bmatrix} 0 & 0 & 0 \\ 0 & 2 & 0 \\ 0 & 0 & 0 \end{bmatrix}$	$\begin{bmatrix} 0.25 & 0.25 & 0.25 \\ 0.25 & -2 & 0.25 \\ 0.25 & 0.25 & 0.25 \end{bmatrix}$	-1.5

The WV-3 image in Figure 3 is a planter located in a public square. The planter's edge building material is very similar to the square's background. A diffusion layer controlled with the values of the selected template was applied to Figure 3(a). The resulting image is displayed in Figure 3(b). The second process is thresholding and binarization of the previous image; the result is shown in Figure 3(c). Finally, an edge-detection layer with selected values of a template for a binary image was applied on the last image; the result is shown in Figure 3(d).
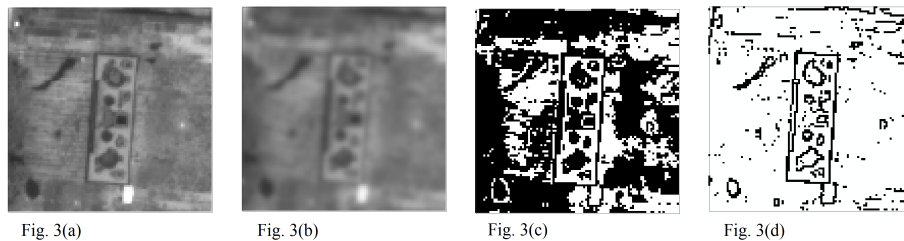

Fig. 3(a)   Fig. 3(b)   Fig. 3(c)   Fig. 3(d)

Fig. 3. Results of applying linear templates in a multilayer CNN

The simulation result is shown in Figure 4. Edge detection results using CNN approach after using a median filter with a 3-by-3 neighborhood are evident from Fig. 4(a) and 4(b). Classic Canny and LoG operators are shown in Fig. 4(c) and 4(d) respectively.
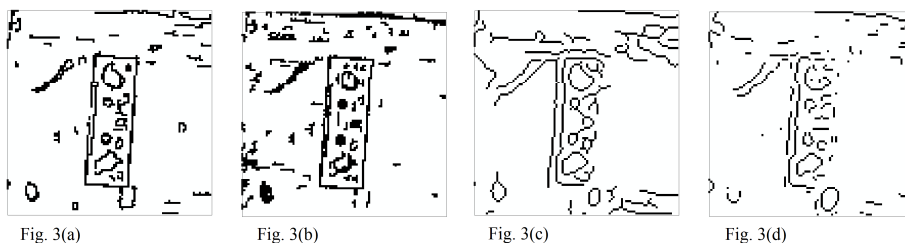
| Fig. 3(a) | Fig. 3(b) | Fig. 3(c) | Fig. 3(d) |

Fig. 4. Simulation result for edge detection using multi-layer CNN, single-layer CNN, Canny and LoG

4   Analysis of Simulation Results

In order to evaluate the accuracy of set performance of the CNN under different shapes of edge, we will introduce the Peak Signal to Noise Ratio (PSNR) as a quality metric in order to estimate the number of connected pixels, denoted by p_{ij}, involved in edge definition.

$$PSNR = 20 * \log_{10} \frac{MAX_p}{\sqrt{MSE}} \qquad (5)$$

where MAX_p is the maximum possible pixel value of the image represented by bits per pixel. Mean Square Error (MSE) is the cumulative squared error between the extracted edge and the truth shape that defines the edge. The MSE is given by:

$$MSE = \frac{1}{MxN} \sum_{i=1}^{M} \sum_{j=1}^{N} \left(\tilde{p}_{ij} - \hat{p}_{ij} \right)^2 \qquad (6)$$

where $\tilde{p}_{ij}$ are the total pixels that define the edge in an ideal image, $\hat{p}_{ij}$ are the connected pixels which belong to the detected edge at the output by using the edge-detection method selected. The result will show the difference between the methods, which allows to know the edge's shape for each example. This kind of metric is a benchmark to evaluate the grade of connectivity of the resultant edge determined by the assessed methods [6]. Results of PSNR calculation for both the single-layer CNN for linear and curved edge detection (Figures 1 and 2) and multi-layer CNN for edge detection (Figure 4) are shown in Table 2.

Table 2. Edge-detection results: Comparisons CNN with other methods

Method	Percent of con-nected pixels	8-con-nected segments	Average pixels per edge segment	MSE	PSNR
Case 1. An example of linear edge detection					
Single-layer CNN	37.3	4	25	2.220	44.7
Canny	28.3	5	17.2	3.534	42.6
LoG	27.2	8	8.6	4.162	41.9
Sobel	21.9	10	5.9	4.368	41.7
Prewitt	21.1	20	3.1	4.410	41.7
Case 1. An example of curved edge detection					
Single-layer CNN	72.5	5	37.4	0.504	51.5
Canny	43.8	6	18.2	2.103	44.9
LoG	43.4	7	16	2.132	44.8
Sobel	40.7	9	11	2.340	44.4
Prewitt	38.0	10	9.6	2.560	44
Case 2. An example of Multi-layer CNN					
Multi-layer CNN	59.9	6	7.5	0.578	50.5
Single-layer CNN	29.4	9	6.4	0.828	48.6
Canny	13.4	13	2.2	1.988	45.1
LoG	10.0	19	2.1	2.592	43.6

For each of the evaluation methods, the table shows the connectivity analysis, which includes the percent of pixels that belongs to the detected edge in reference to the truth edge, the number of connected segments that defines the edge for 8-connected neighborhood as well as the pixels average per edge segment. A higher percent of connected pixels that belongs to the detected edge means a lower MSE, and as can be noticed, this generates a high value of PSNR. A higher value of PSNR generally indicates that the reconstruction is of higher quality. Here, the "signal" is the truth shape that defines the edge to extract in the original image, and the "noise" is the error represented as the non-identified pixels of the edge by the segmentation process.

It can be observed first, that the percent of pixels which belong to the detected edge obtained with the proposed CNN approach is composed of a large number of pixels, in comparison with the other edge-detection methods for all cases. Furthermore, the number of connected segments is always the smallest and the pixels average per segment has the largest number of pixels. In terms of PSNR, the single-layer CNN proposal showed the best result; however, for a curved edge the results are better than those from the linear edge. This can be explained in terms of the object's height to be detected and the context of its shadow that describe the edge-detection task. The multi-layer CNN proposal detected the edges as well as the ideal and noiseless edge detection. In fact, it is remarkable that the multi-layer CNN proposal is more accurate than the other evaluated methods. The method based on multilayer CNN determines the ideal edges of the image in detail and with high precision. It can be seen that the

single-layer CNN performance is satisfactory but the output image has too much noise. Results from Canny and LoG operators respectively are just unsatisfactory.

5   Conclusions

The edge in a very high spatial resolution image gives an indication of the physical extent of an object within the image, and its accurate segmentation is very important because the performance of the tasks after the edge detection, such as image segmentation and image registration, depend on the information of the edge. The edge is the connected boundary between two different regions that define different objects. The algorithms dedicated to obtain the edge must meet two principles: detection of all abrupt changes in the intensity of an image that defines an object and the largest number of connected regions that describes these changes. So that, local connectivity's property in an edge detection task is essential and this property is intrinsic to the CNN model.

In each of the simulations, single or multi-layer CNNs for edge-detection simulations in VHR satellite imagery, similar to the edge detection problem in images affected by noise, have been considered because of the objects that hide some other objects of the scene. Satisfactory results could be achieved by choosing standard templates among a few experiments available in the literature. Experimental results have shown that the Single-layer CNN for linear and curved edge detection is an excellent alternative for connected edges detection in the VHR context. In case of the multi-layer CNN, composed of cells that have several single-layer arrays stacked one above the other, it is possible a better edge-detection task than when using the other evaluated methods.

References

1. Poli, D., Remondino, F., Angiuli, E., Agugiaro, G.: Radiometric and geometric evaluation of GeoEye-1, WorldView-2 and Pléiades-1A stereo images for 3D information extraction. ISPRS Journal of Photogrammetry and Remote Sensing 100, 35–47 (2015)
2. Myint, S.W., Gober, P., Brazel, A., Grossman-Clarke, S., Weng, Q.: Per-pixel vs. object-based classification of urban land cover extraction using high spatial resolution imagery. Remote sensing of environment 115(5), 1145–1161 (2011)
3. Shrivakshan, G.T., Chandrasekar, C.: A comparison of various edge detection techniques used in image processing. International Journal of Computer Science Issues 9(5), 272–276 (2012)
4. Maini, R., Aggarwal, H.: Study and comparison of various image edge detection techniques. International journal of image processing 3(1), 1–11 (2009)
5. Canny, J.: A computational approach to edge detection. IEEE Transactions on Pattern Analysis and Machine Intelligence (6), 679–698 (1986)

6. Gazi, O.B., Belal, M., Abdel-Galil, H.: Edge Detection in Satellite Image Using Cellular Neural Network. International Journal of Advanced Computer Science & Applications, 5(10), 61–70 (2014)

7. Bedawi, S.M., Kamel, M.S.: Segmentation of very high resolution remote sensing imagery of urban areas using particle swarm optimization algorithm. In: Campilho A., Kamel M (eds.) ICIAR 2010, Part I. LNCS, vol. 6111, pp. 81–88. Springer Heidelberg (2010)

8. Chua, L.O., Yang, L.: Cellular neural networks: theory. IEEE Transactions on Circuits and Systems 35(10), 1257–1272 (1988)

9. Chua, L. O., & Yang, L.: Cellular neural networks: Applications. IEEE Transactions on Circuits and Systems 35(10), 1273–1290 (1988)

10. Li, H., Liao, X., Li, C., Huang, H., & Li, C.: Edge detection of noisy images based on cellular neural networks. Communications in Nonlinear Science and Numerical Simulation 16(9), 3746–3759 (2011)

11. Matsumoto, T., Chua, L.O., Suzuki, H.: CNN cloning template: connected component detector. IEEE Transactions on Circuit and System 37 (5), 633–635 (1990)

12. Chua, L. O., Roska, T.: Cellular neural networks and visual computing: foundations and applications. Cambridge University Press (2002)

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

14. Wang, W., Yang, L. J., Xie, Y. T., & An, Y. W.: Edge detection of infrared image with CNN_DGA algorithm. Optik-International Journal for Light and Electron Optics 125(1), 464–467 (2014)

15. Zarándy, Á.: The art of CNN template design. International Journal of Circuit Theory and Applications 27(1), 5–23 (1999)

16. Karacs, K., Cserey, G. Y., Zarandy, A., Szolgay, P., Rekeczky, C. S., Kek, L., Szabó, V., Pazienza, Roska, T.: Software Library for Cellular Wave Computing Engines in an era of kilo-processor chips, Version 3.1. Cellular Sensory and Wave Computing Laboratory of the Computer and Automation Research Inst., Hungarian Academy of Sciences and the Jedlik Laboratories of the Pazmany P. Catholic University, Tech. Rep (2010)

17. Chua, L. O.: CNN: a version of complexity. Int J Bifurcat Chaos 7 (10), 2219–2425 (1997)

18. Zhang, M., Min, L., Zhang, X.: Automatic Robust Designs of Template Parameters for a Type of Uncoupled Cellular Neural Networks. In: Foundations of Intelligent Systems. Springer Berlin Heidelberg. pp. 577–590 (2014)

19. Chua, L.O., Yang, L., Krieg, K.R.: Signal processing using cellular neural networks. Journal of VLSI signal processing systems for signal, image and video technology 3(1-2), 25–51 (1991)

20. Mrugalski, M.: Advanced neural network-based computational schemes for robust fault diagnosis. Springer. (2014)

21. Parmaksızoğlu, S., & Alçı, M.: A novel cloning template designing method by using an artificial bee colony algorithm for edge detection of cnn based imaging sensors. Sensors 11(5), 5337–5359 (2011)