MOBILE ROBOTS GUIDANCE BY USING CELLULAR NEURAL NETWORKS

I. Gavriluţ*, V. Tiponuţ, A. Gacsádi***

**University of Oradea, Electronics Department, Str. Universităţii, 1, 410087, Oradea, e-mail: gavrilut@uoradea.ro, agacsadi@uoradea.ro.*
***Politehnica University from Timişoara, Applied Electronics Department, B-dul Vasile Pârvan, 2, 300223, Timişoara, e-mail: virgil.tiponut@etc.upt.ro.*

Abstract: The paper presents some vision-based algorithms for mobile robots guidance in an environment with obstacles. Cellular Neural Networks (CNNs) processing techniques are used here for real time motion planning to reach a fixed target. The CNN methods are considered an advantageous solution for image processing in autonomous mobile robots guidance.

Keywords: image processing, CNN, path planning, mobile robots.

1. INTRODUCTION

An important research theme in mobile robotics is the design of path planning system. These systems should provide trajectory for one or more robots from an initial position to a target position, so that the obstacles located between these positions must be avoided. If we take into account that the obstacles as well as the target can move and that the obstacles can have any shape, it becomes clear that this problem is not a trivial one.

The path planning is a complex process starting with the perception of the environment based on maps or sensory information obtained through robot's sensors. The most frequent sensors for mobile robot are the visual sensors such as cameras using CCD arrays, and range sensors such as laser, IR, or sonar. Though recent research using a camera includes efficient localization methods due to the wealth of information, efficient processing using limited computing power is still not an easy task.

Using cellular neural networks (Chua and Yang, 1994), (Roska and Chua, 1993), which have very short image processing time a good displacement speed for mobile robots, can be obtained. The choice of CNNs for the visual processing is based on the possibility of their hardware implementation in large networks on a single VLSI chip.

This paper proposes a new path planning method by using cellular neural networks. The methods were simulated using the "Cadetwin" environment (***C**NN **A**pplication **D**evelopment **E**nvironment and **T**oolkit under **Win**dows*) (***, CadetWin, 1999). At the end, we present a comparison between these methods, used for trajectory planning, based on CNN algorithms.

2. CELLULAR NEURAL NETWORKS

A cellular neural network is an analog, nonlinear, dynamic, multi-dimensional circuit having locally recurrent topology (Chua and Yang, 1994)). The basic circuit units named cells or artificial neurons are connected only to its neighbor units. The basic cellular neural network (Chua and Yang, 1994), (Roska and Chua, 1993) has a two-dimensional rectangular structure composed from identical, nonlinear analog circuits (cells) arranged, for example, in M rows and N columns (see Figure 1).

Due to their locally connections, the field areas occupied on the chip by the connection wire is minimized so that these networks could be implemented in the present VLSI technology (Roska and Chua, 1993). Cells that are not directly connected together may affect each other indirectly because of the propagation effects of the continuous-time dynamics of cellular neural networks.

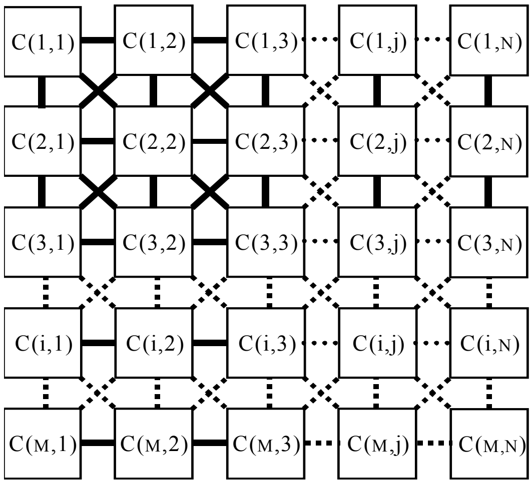

Fig. 1. A basic two-dimensional cellular neural network with M rows and N columns. The links between the cells indicate that there are interactions between the linked cells.

A CNN is entirely characterized by a set of nonlinear differential equations associated with the cells in the circuit. The mathematical model for the state equation of the single cell C(i,j) is given by the following set of relations:

$$\dot{x} = \frac{dx_{ij}}{dt} = -x_{ij} + \sum_{C_{kl} \in S_r} A_{ij,kl}\, y_{kl} + + \sum_{C_{kl} \in S_r} B_{ij,kl}\, u_{kl} + z_{ij}, \qquad (1)$$

where x_{ij} denotes the state of the cell C_{ij}; y_{kl}, u_{kl} denote the output and input respectively of cells C_{kl} located in the sphere of influence with radius *r*, S_r, from C_{ij} cell, $C_{kl} \in S_r$; A(ij,kl) and B(ij,kl) are the feedback and control templates respectively; z_{ij} is the bias term.

The equation, which expresses the output value of Cij cell, is given in the relation (2).

$$y_{ij} = f(x_{ij}) = \frac{1}{2}\left[\left|x_{ij} + 1\right| - \left|x_{ij} - 1\right|\right] \qquad (2)$$

where y_{ij} denotes output value of Cij.

In Figure 2 is presented how the two-dimensional signals are processed with a standard cellular neural network having templates of 3×3 dimensions. Applying the image U on the CNN input and having at state an initial image X, the CNN output image Y is obtained by using operators A, B, z, when that equilibrium point is reached.

Cellular neural networks are very suited for high-speed parallel signal processing like image or other two-dimensional signals processing (Gacsádi and Szolgay, 2003). In the same time CNN was used for solving partial differential equations (PDEs). For example, it is presented a numerical solution of a class of PDEs by using emulated digital CNN-UM implemented on FPGAs (Nagy and Szolgay, 2003).

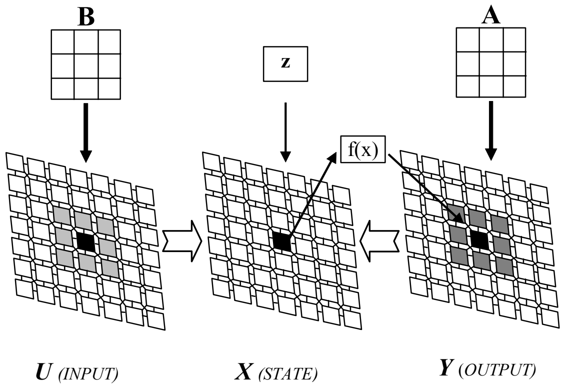

Fig. 2. Signals processing with a standard cellular neural network having templates of 3×3 dimensions.

3. MOBILE ROBOT PATH PLANNING BY USING CNN

By using cellular neural networks (Chua and Yang, 1994), (Roska and Chua, 1993), which have a very short image processing time, it can be obtained a good displacement speed for mobile robots. The CNN methods have been considered a promising solution for images processing in autonomous mobile robots guidance. The choice of CNNs for the visual processing is based on the possibility of their hardware implementation in large networks on a single VLSI chip.

A variety of approaches have been proposed to use CNNs for mobile robots path planning based on images in unstructured environments (Siemiatkowska and Dubrawski, 1998), (Gacsádi *et al*., 2002), (Gavriluț *et al*., 2005), (Gavriluț *et al*., 2006), (Vilarino and Rekeczky, 2004).

In (Gacsádi and Szolgay, 2000) it is presented an image based CNN algorithm for following a moving object by means of a video camera mounted on the arm of a robot having a two-degree freedom. Between two consecutive acquired images through complete parallel processing of two-dimensional signals with cellular neural networks the whole system can working in real time so that camera was oriented all the time towards the moving object.

Usually, for mobile robot path planning by using CNN, the environment with obstacles must be divided into discrete images and in this way it is possible to represent the workspace in the form of an M×N array, through a standard neural network having M×N cells. The processed images are gray-scale, having the value of the pixel in the interval [-1,1], known as the standard CNN domain. For binary images, these values could be only +1 for the black pixels and –1 for the white pixels.

In our paper, the mobile robot is considered to be placed in a plane workspace, in which there are only static obstacles (see Figure 3). The robot has to take the shortest way toward the target avoiding the obstacles located between the start and the target positions.

The gray-scale images of the environment with obstacles are acquired using a video camera then transferred to the cellular neural network or CNN chip, respectively. After CNN elementary preprocessing, the binary image of the workspace (see Figure 3) is obtained and will be used in the algorithm.

After a spatial discretization of the images, which corresponds to the CNN resolution, we suppose that each obstacle is represented by at least one pixel having a fixed value. The robot and target positions are each identified by a single pixel. In our example, the occupied pixels having values +1 represent the forbidden positions where the robot can't move and the pixels having values –1 represent the free positions accessible for the mobile robot.

The flowchart of the CNN algorithm used for image-based path planning in the case of a mobile robot is presented in Figure 4.

The proposed algorithm is able to detect, at each new cycle of the global algorithm, the possible cases with no free path, without obstacles, from the start to the target position and it will stop.

3.1. Distances evaluation

For optimal trajectory obtaining between start and target positions, the distances between the free points from the workspace where the mobile robot can be along the algorithm, and the target point must be determined.

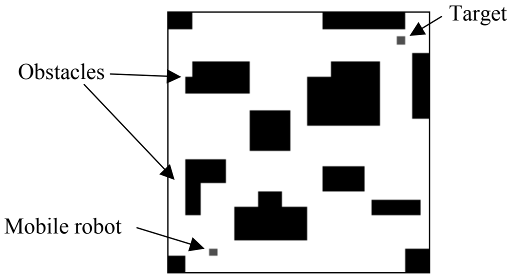

Fig. 3. The binary image of the workspace with obstacles.

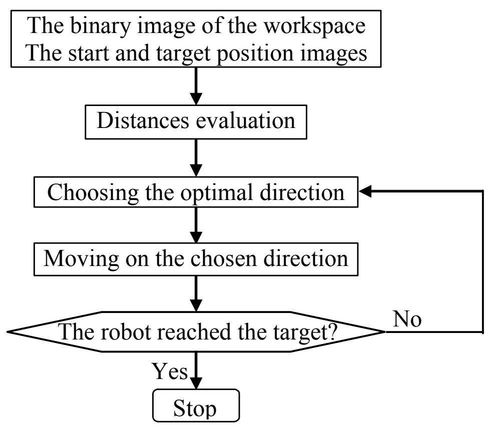

Fig. 4. Flowchart for mobile robot path planning algorithm by using CNN processing.

With this aim, in the image plane a wave is generated, having the source origin situated in the target point. The image for distance evaluation between two different positions in the workspace, one of them being the target position, could be achieved using the template EXPLORE (***, CSL - CNN Software Library, 1999) defined by relations (3).

$$A = \begin{pmatrix} 0 & a & 0 \\ a & 1 & a \\ 0 & a & 0 \end{pmatrix} \quad B = \begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix} \quad z = 0 \qquad (3)$$

The template mentioned above is nonlinear because parameter *a* is a nonlinear function, and depends on the difference y_{ij}-y_{kl}.

Through its propagation the wave searches all the possible directions in the environment, starting from the target point, as in Figure 5. An image in which all the pixels have the value +1, except a pixel that corresponds to the target position having –1 value will represent the state $x(t_0)$ of the network. This image is, in fact, the exactly opposite of the image containing the target point. The current binary image of the workspace is used as a mask image in this processing step. The mask image is that image used in CNN which allow pixels modification only in a specified zone from the input image. The pixels corresponding to the borders of the workspace, namely the pixels that are at the boundaries of the cellular neural network are always considered forbidden positions. As a result of these operations, the value of the pixel corresponding to the target position in the output image remains unchanged at its initial value –1 while the pixels having the value +1 are going to be the forbidden positions through which the robot cannot pass. All the other pixels will have values that proportionally increase with the distance between them and the target position. Thus, starting from the center of the wave source, the value of pixels is increasing approximately with a distance measure unit, when the wave radius is increasing by 1.

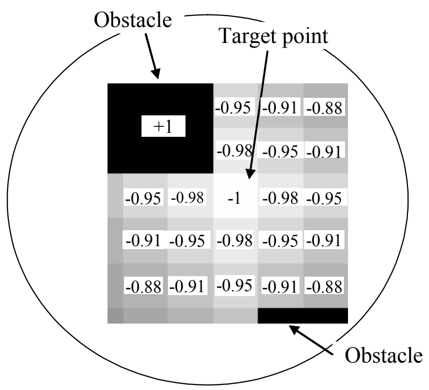

Fig. 5. The principle of determination of the distance between the target and the points from the workspace through wave propagation having the origin of the source in the target point; the values of pixels around target point are represented.

If the environment is without obstacles the wave has a circular propagation. Finally, the values of pixels in the image are proportionally with the distance from the target.
The image for distance evaluation between positions of the workspace and the target, except the obstacle positions, could be achieved using the template EXPLORE. The number of iterations in the case of this template will be gradually increased until the pixel corresponding to the start position will be modified.

3.1. Choosing the optimal direction

Choosing the optimal direction can be done, in this paper, in two ways: through successive comparisons or using the template PATH.

Through successive comparisons. The image having a single pixel, which gives the position of the start point, will be overlap with the image obtained by evaluating the distance between the target point and the free points from workspace.
The mobile robot trajectory is determined by choosing, at each step, the optimal direction given by the neighbor pixel having minimal value. This pixel is obtained, in this method, through successive comparisons between all pixels situated in the neighborhood (with r = 1) of the actual position of the robot.
The choice of the optimal direction is a processing step which is based on the possibility of extracting the pixel value from a gray-scale image if its position is given by an inactive pixel having the value -1, into a binary image used as a mask image.
The robot position is represented by an image having all pixels at value -1, excepting one single pixel with value +1, which indicates the start, or current position where the robot will choose a new moving direction. Through their inversion it is obtained a binary image having the same dimension, which will be further used as well as mask image.
In this processing step it is necessary the gray-scale image containing the free positions evaluating vis-à-vis target position and the mask image above mentioned. Using a local method, a neighbor cell from eight possible directions N, S, E, V, SE, NE, NV, SV, is chosen in such a way that the value of the cell is the smallest possible value. The path followed by the mobile robot choosing the direction corresponding to this cell assures the shortest way to get to the target. Practically, the pixel values from the robot neighborhood (Figure 5) will be compared in order to choose that pixel which has the minimal value. In this respect, elementary processing AMC (*Extended **A**nalogic **M**acro **C**ode and Interpreter*) are used here through the template family SHIFT (4), corresponding to the eight directions mentioned above.
In case of one template from the SHIFT family (***, CSL - CNN Software Library, 1999) only one

$$A = \begin{pmatrix} 0 & 0 & 0 \\ 0 & 2 & 0 \\ 0 & 0 & 0 \end{pmatrix} \quad B = \begin{pmatrix} se & s & sv \\ e & 0 & v \\ ne & 0 & nv \end{pmatrix} \quad z = 0 \qquad (4)$$

element from operator B is equal to 1, the other elements having value 0, so that: SHIFTE (e=1), SHIFTSE (se=1), SHIFTS (s=1), SHIFTSV (sv=1), SHIFTV (v=1), SHIFTVN (nv=1), SHIFTN (n=1), SHIFTNE (ne=1).

By using the template PATH. Having indicated the robot position by one black pixel, their neighborhood with radius r = 1 can be easily obtained using the template DILATION (***, CSL - CNN Software Library, 1999).
After some logical operation on that image, a mask image is obtained which will be overlapped on the wave image, presented in Figure 6a. The resulting image represents the wave only in the robot neighborhood, the other pixels from the image having value +1 (Figure 6b). Further, it must be chosen the pixel, which has the minimal value. In the example, presented in Figure 6, the value of the pixel is 0,63.
For that, it can be design a template, which applied on image shown in Figure 6b, will be obtained an image with all the pixels with value -1 excepting one, which has the value +1 and it indicates the future position of the robot.
By applying this template it can be indicate the next position of the robot in a fixed configuration of pixels, but in another configuration this it’s not good for that. The bias value must be the same, changed according with the new configuration, where the robot is situated.
From this reason, in our algorithm some processing operations will be made on the image presented in Figure 6a, so that by applying the same template on the image from Figure 6b it will be obtained an image which indicates next position of robot.
Through CNN processing the minimal value from the image, shown in Figure 6b, it will be memorized and then an image having all pixels at that value (in our example 0,63) is created.

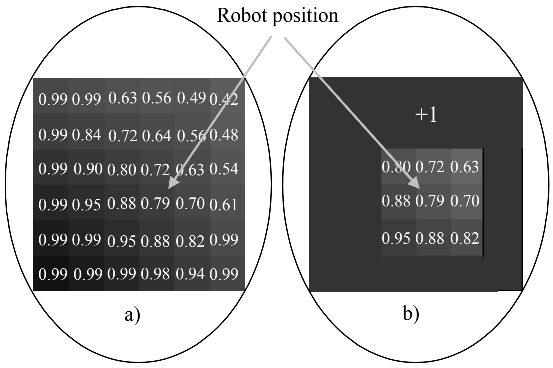

Fig. 6. The resulting image represent the wave only in robot neighborhood, a) pixels values in the wave image around robot position after applying template EXPLORE, b) image representing pixel values only in the robot neighborhood.

If that image is subtracted from the image presented in Figure 6a, an image having pixel values around the robot position (Figure 7a) or in the robot neighborhood (Figure 7b) is obtained.
The resulting image has approximately the same values around the robot current position, respectively of robot position in its trajectory toward target. So, having an adequate template applied through a mask image on the image like in Figure 7b, an image with the future position is obtained, for all position from free space where the robot can be situated on the its trajectory.
This template, named PATH, can be calculated and its form is given by the relation (5).

$$A=\begin{pmatrix}0&0&0\\0&0&0\\0&0&0\end{pmatrix}\quad B=\begin{pmatrix}0&-0.1&0\\-0.1&-8&-0.1\\0&-0.1&0\end{pmatrix}\; z=0.3. \qquad (5)$$

By applying this template, on the image from Figure 7b, the future value of every pixel depends on the current value of pixel as well as on the pixels values from its neighborhood. This template is applied at every step with a mask image which delimits the robot neighborhood, having all pixels of value +1 in outside of that.
In conclusion, if that template is applied to the input image like in Figure 7c or Figure 8a, and if on the state there is an image having all the pixels at value 0 (Figure 8), it is obtained an image that represents, by a black pixel, the future position of the robot (Figure 8c).

3.2. Moving on the chosen direction

The robot displacement toward the target was realized on the chosen direction that remains unchanged as long as the current direction allows the robot to move closer to the target. An example of full path computing result for the environment's image (32*32 pixels) from Figure 3, is shown in Figure 9.

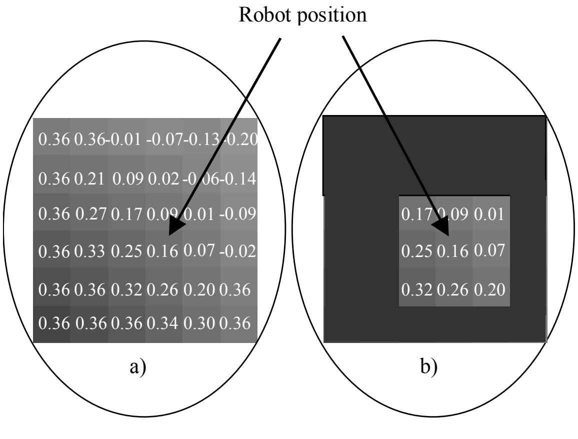

Fig. 7. The wave updating: a) updating value of wave after the image having all pixels at value 0,63 sustracting, b) pixels values from robot neighborhood.

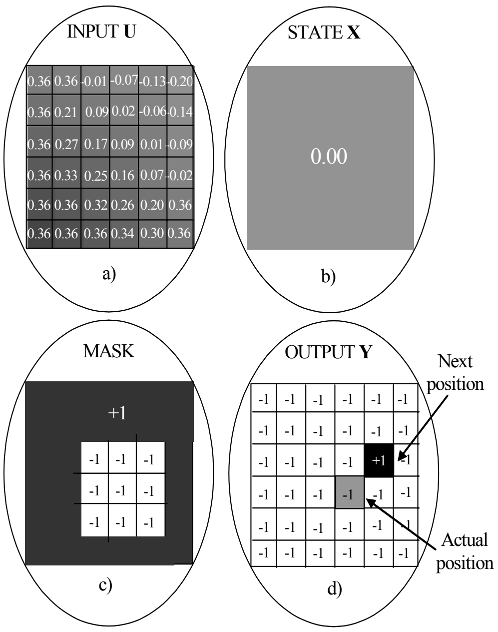

Fig. 8. Determination of the next position for mobile robot; a) the environment with obstacles image overlap on the updated wave image, b) image having all pixels at value 0, c) the mask image, d) the image having all the pixels at value -1 excepting that pixel which indicates the future position of the robot.

4. CONCLUSIONS

The experimental results of the path-planning algorithm were obtained using the simulation programs written in assembling language (*Extended **A**nalogic **M**acro **C**ode and Interpreter*). In our simulation we used 32×32 images.
The CNN algorithm was tested in both variants for next position choosing. The environment with obstacles was identical so that we made a comparison between these two methods presented in paper, regarding the total processing time necessary for robot to reach the target position. We use for that a personal computer having Pentium III processor at 1600 MHz.
In Figure 10, the dependence between the necessary time for processing and the trajectory length, expressed in number of pixels, is presented for

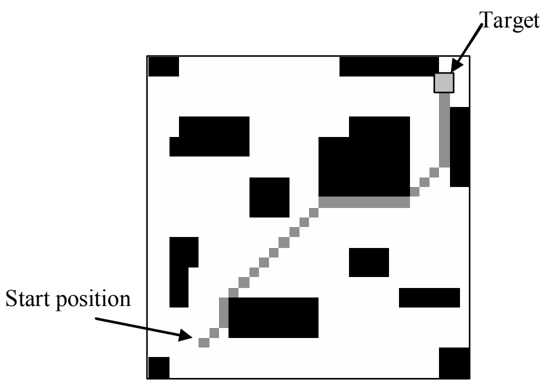

Fig. 9. An example of full path computing result.

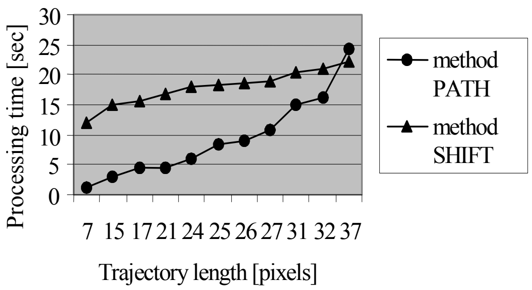

Fig. 10. Dependency between the trajectory length and the necessary processing time in case of the two methods for optimal direction choosing.

the both methods. The processing time is very short in case of applying the template PATH, for determining of next position. In case of the other method, through successive comparisons, the total processing time is shorter only in situation when the mobile robot is relative away from target.

The total processing time is the sum of the times corresponding to each step of the algorithm. The minimum processing time of the step in which the evaluating of distance between points from the free workspace and the target is made, depends on the distance between the start and the target position because is absolutely necessary that wave reaches, through propagation, the start position. If the wave front no reaches the target pixel then can be increased the image resolution at 64×64 pixels.

A necessary condition for the proposed algorithms to run correctly is that at each captured image, the start and target point must be identified. Putting this condition, the dimension of the image and the minimum sample step for spatial discrete representation of the captured image is obtained.

REFERENCES

Chua, L. O. and L. Yang, (1988). Cellular Neural Networks: Theory and Application, IEEE Trans. On Circuits and Systems, (CAS), Vol. 35, pp. 1257-1290.

Gacsádi, A., T. Maghiar and V. Tiponuţ, (2002). Path planning for a mobile robot in an environment with obstacles using cellular neural network, Intern. Workshop on Cellular Neural Networks and their Applications, (CNNA 2002), Frankfurt/Main, Germany, pp. 188-194.

Gacsádi, A. and P. Szolgay, (2000). An analogic CNN algorithm for following continuously moving objects, Proceedings of the IEEE Intern. Workshop on Cellular Neural Networks and their Applications, (CNNA 2000), pp 99-105, Catania.

Gacsádi, A. and P. Szolgay, (2003). Adaptive Image Enhancement by Using Cellular Neural Networks, Proceedings of the European Conference on Circuit Theory and Design, Cracow, Poland, September.

Gavriluţ, I., A. Gacsádi, L. Ţepelea and V. Tiponuţ, (2005). Motion planning for two mobile robots in an environment with obstacles by using cellular neural networks, The 7-th Intern. Symposium on Signals, Circuits and Systems, (ISSCS 2005) pp. 801-804, Iaşi, Romania.

Gavriluţ, I., A. Gacsádi, C. Grava, V. Tiponuţ, (2006). Vision based algorithm for path planning of a mobile robot by using cellular neural networks, Intern. Conf. on Automation, Quality &Testing, Robotics, AQTR 2006, pp. 306-311, Cluj-Napoca, Romania.

Kim, H., H. Song, T. Roska and L.O. Chua, (2002). Optimal Path Finding with Space and Timevariant, Metric Weights via Multi-layer CNN, Intern. Journal of Circuit Theory and Applications, (IJCTA), Vol. 30, pp. 247-270.

Liñán, G., R. Domínguez-Castro, S. Espejo and A. Rodríguez-Vázquez, (2001). ACE16k: A Programmable Focal Plane Vision Processor with 128 x 128 Resolution, - European Conference on Circuit Theory and Design (ECCTD ’01), pp. 345-348, Espoo.

Nagy, Z. and P. Szolgay, (2003). Numerical Solution of a Class of PDEs by Using Emulated Digital CNN-UM on FPGAs, Proceedings of the European Conference on Circuit Theory and Design, Cracow, Poland, September.

Roska, T. and L. O. Chua, (1993). The CNN universal machine: an analogical array computer, IEEE Trans. on Circuits and Systems II: Analog and Digital Signal Processing, Vol. 40, pp. 163-173.

Siemiatkowska, B. and A. Dubrawski, (1998). Cellular Neural Networks for Navigation of a Mobile Robot, L. Polkowski and A. Skowron (Eds.): RSCTC'98, LNAI 1424, pp. 147-154, Springer-Verlag Berlin Heidelberg.

Vilarino, D. L. and C. Rekeczky, (2004). Shortest path problem with pixels level snakes: Application to robot path planning, Proceedings of the IEEE Intern. Workshop on Cellular Neural Networks and their Applications, (CNNA 2004), pp. 135-140, Budapest.

***, CSL - CNN Software Library (Templates and Algorithms), Version 7.3, (1999). Analogical and Neural Computing Laboratory, Computer and Automation Institute, Hungarian Academy of Science, Budapest.

***, CadetWin-99, CNN application development environment and toolkit under Windows Version 3.0, (1999). Analogical and Neural Computing Laboratory, Computer and Automation Institute, Hungarian Academy of Science, Budapest.