ISTANBUL UNIVERSITY –
JOURNAL OF ELECTRICAL & ELECTRONICS ENGINEERING

YEAR : 2003 (635-640)
VOLUME : 3
NUMBER : 1

INTERPOLATION OF 2D SIGNALS USING CELLULAR NEURAL NETWORK

Alexandru GACSÁDI[1] **Péter SZOLGAY**[2]

[1] Electronics Department, University of Oradea, Str. Armatei Române, Nr.5, 3700 Oradea, Romania, Tel: +40-259-408195, Fax: +40-259-432789, e-mail: agacsadi@uoradea.ro.

[2] Analogical and Neural Computing Laboratory, Computer and Automation Institute, Hungarian Academy of Sciences, P.O.B.63, H-1502 Budapest, Hungary, Tel: +(361) 2095265, Fax: +(361) 2095264, e-mail: szolgay@sztaki.hu., affiliated also to Image Processing and Neurocomputing Department, Veszprém University, Egyetem u. 10. Veszprém, Hungary

ABSTRACT

*In this paper some new interpolation methods using cellular neural networks (CNN) are proposed. The templates have only linear feedback component with a 3*3 dimension. The templates were computed by using analytic methods. By implementing these interpolation methods on a "CNN-Universal Chip", the processing becomes completely parallel and it can be done in real time. The testing results achieved by using these interpolation methods are presented for increasing gray-scale image resolution, for image reconstruction and the encoding-decoding the images necessary for their transmission.*

Keywords: *interpolation, Cellular Neural Networks (CNN) and templates.*

I. INTRODUCTION

The Cellular Neural Networks (CNN) [1] are able to process 2D signals in real time. The CNN-chip is the central element of the supercomputer "CNN-Universal Machine" (CNN-UM) [2]. Nowadays, there already exists a 64*64 pixel "CNN Universal Chip"(CNN-UC) [3], working with linear templates having a 3*3 dimension. The CNN-UM already proved its usefulness in many applications related to the processing of images in real time.

The 2D signals used for interpolation will be considered as discrete 2D functions $v(x_i,y_j)$, having a M*N dimension domain. From now on, instead of "two-dimensional signals" the notion "image" will be used, even if the character of the signals is not the character of a strictly speaking image; for example, the control "images" of robots [4] or the "images" provided by the tactile sensor of robots. The (x_i y_j) variables represent the spatial co-ordinates of an image element at an arbitrary point, horizontally and on vertically, with $1 \leq i < M$ and $1 \leq j < N$. The function values, $v(x_i,y_j)$ for gray-scale images, are in the interval [-1,1] known as the standard domain in CNN. For binary images, these values could be only +1 (for the black pixels) and –1 (for white pixels). If a finite number P of values of the pixels $v(x_i,y_j)$, called knot values [5], are known at a certain moment, then the image interpolation method lies in the estimation of unknown

Received Date : 15.09.2002
Accepted Date: 25.12.2002

values of pixels computed depending on the known values of the pixels. A $v(x_i,y_j)$ function can be used for the $v(x_i,y_j)$ function estimation at the unknown points, if the following conditions are satisfied:

$v(x_i,y_j)=V_p$ and a p=1,2,…,P. (1)

The function $v(x_i,y_j)$ is obtained with the help of linear independent functions $v(x_i, y_j)$.

The following convention will be used for the simplification of the notations:

$$v(x_i,y_j)=v(i,j);\ \frac{\partial v}{\partial x_i}=\frac{\partial v}{\partial i};\frac{\partial v}{\partial y_j}=\frac{\partial v}{\partial j} \qquad (2)$$

for any (i,j), $1 \le i < M$; $1 \le j < N$.

The goal is to obtain a result image, indifferent to the interpolation method used, which has the smallest errors compared to the chosen image considered as a reference image. Another important aspect is the processing time of the interpolation methods. The processing time should be as short as possible, and more, the process should be done in real time. The processing becomes completely parallel and it can be done in real time if the CNN interpolation methods proposed are implemented on a "CNN Universal Chip" (CNN-UC). The processing speed increases by using the CNN, but unfortunately the direct use of an interpolation function is not possible. That is the reason why, when designing templates, we should aim to minimize the norms through which the surfaces, described by the values and amplitudes of pixels, can be achieved. Surfaces resulted must be as smooth as possible. The computed templates will have a similar effect to the given through interpolation with linear, square and spline cubic functions.

The positions of known image elements are fixed for all the input images tested with interpolation methods. The values of the elements will not change during the transient, due to the use of mask images.

The principle of enlargement of images lies in increasing the number of rows and columns for the image tested. New pixels will be obtained, and the values for these pixels will be computed through interpolation.

The dimension of the initial image described by the general variable v(i,j) is enlarged before linear interpolating. This is achieved by inserting pixels having a zero value in rows and columns of the initial image. These values introduced by inserting pixels in rows and columns (the zeros) will be replaced with the values that represent an average value. The average value is calculated with the values of neighbor pixels, through a linear interpolation process. Due to the sequential principle used and the big number of calculations, the processing time for this interpolation method is high and increases proportionally with the image dimension.

In paper [6] an algorithm is given for linear interpolation using CNN implementation for the enlargement of image, and at the same time keeping the size of pixels unchanged. Through this method, the final result is obtained in two steps, for each of them, using only control templates B. Another adaptive CNN algorithm for interpolation is provided in [7], for achieving a better behavior through interpolation for the edges of objects, using only control templates.

In section II, we give the design methods of the feedback templates for gray-scale image CNN interpolation. Some experimental results of testing the CNN interpolation methods running on software simulation and by using a 64*64 CNN-UC are presented in section III. Conclusions are given in section IV.

2. IMAGE INTERPOLATION, USING CNN WITH FEEDBACK TEMPLATES

In the case of using CNN for interpolation methods, it must be considered that even if the cells are only locally connected, the output and the state of a cell will be influenced by the cells located outside the neighborhood N_r. The bounded connection is given by the neighborhood's radius r. This influence is due to the propagation effect determined through the cell's dynamics. This effect takes place only if there is feedback from the output of the neighbor cells to the cell's state, made through a feedback template A. If these interpolation methods using CNN, only with feedback templates, are implemented on a CNN-chip, then the processing becomes completely parallel. In [8], the feedback template A is of

5*5 dimension. Nowadays, a 5*5 dimension template and the values obtained for this template can not be implemented on CNN-UM. In the CNN domain, it is usually very difficult to design the feedback templates A needed in processing gray-scale images. In this section, the constraints are defined for computing the feedback templates A needed in the interpolation methods are described in the following steps:

- When using CNN with feedback templates for interpolation methods, it is necessary to have a mask image, for fixing the values of the cells that are known before beginning the interpolation.
- When designing feedback templates, with any method, the necessity of knowing boundary conditions must be eliminated. This can be achieved through image extension at the boundaries, with virtual rows and columns having the same values as the neighbor cells. This is called the " zero-flux" condition [9].
- The image used for interpolation is applied to the network's state, namely x(i,j) = v(i,j). A cell from the CNN is characterized by the following state equation:

$$\frac{dx_{ij}}{dt} = -x_{ij} + \sum_{C_{kl}\in N_r} A_{ij,kl}y_{kl} + \sum_{C_{kl}\in N_r} B_{ij,kl}u_{kl} + z_{ij}$$

for any (i,j), $1 \le i < M$; $1 \le j < N$, (3)

where u_{kl} is the input of this cell, x_{ij} represents the state value of the CNN cell and y_{kl} is the output. The cells having the notation C_{kl} are the neighbor cells for the C_{ij}, namely $C_{kl} \in N_r$. The feedback template in the state equation has the notation $A_{ij,kl}$, the control template is $B_{ij,kl}$ and z_{ij} is the bias value. From now on, only feedback templates are going to be used, the B template and the bias value are equal to zero.

- In the case of using the structure of the template presented above, the necessary and sufficient condition of stability and keeping the output values in the interval [-1, +1], in any conditions, is:

$$\left| \sum_{C_{kl}\in N_r} A_{ij,kl}y_{kl} \right| \le 1 \qquad (4)$$

In this case, the state of network is identical to the network's output. This means that the state of every cell in the network can not achieve the saturation level:

$$x_{ij} \equiv y_{ij} \text{ for any (i,j), } 1 \le i < M;\ 1 \le j < N \qquad (5)$$

- The computed values of the feedback template must be obtained in a way that each cell from the network is in a stable state. The network achieves the stable state if the following condition is fulfilled [1]:

$$\lim_{t\to\infty} x_{ij}(t) = x_{ij}^* \text{ and } \left.\frac{dx_{ij}}{dt}\right|_{x_{ij}=x_{ij}^*} = 0 \qquad (6)$$

For any (i,j) $1 \le i < M$; $1 \le j < N$, it result:

$$\sum_{C_{kl}\in N_r} A_{ij,kl}y_{kl} = x_{i,j} = y_{i,j} \qquad (7)$$

- For the design of feedback templates we can use also methods for minimization of norms, minimization that constitute cost functions. By finding the global minimum for these cost functions, we can design other feedback templates for square or spline cubic interpolation [10].

The feedback template aintpol1.tem for linear interpolation obtained is: (8)

$$A = \begin{array}{|c|c|c|}\hline 0.125 & 0.125 & 0.125 \\ \hline 0.125 & 0 & 0.125 \\ \hline 0.125 & 0.125 & 0.125 \\ \hline\end{array} \quad B = \begin{array}{|c|c|c|}\hline 0 & 0 & 0 \\ \hline 0 & 0 & 0 \\ \hline 0 & 0 & 0 \\ \hline\end{array} \quad z = \boxed{0}$$

For this case, the network will reach stability, if for each unknown cell, the value as an average value computed from the values of the neighbor cells is obtained.

The cost function for square interpolation using CNN with feedback template is given as follows:

$$E(v(i,i)) = \sum_{i=1}^{M}\sum_{j=1}^{N}\left[(\frac{\partial v}{\partial i})^2 + (\frac{\partial v}{\partial j})^2 \right] \qquad (9)$$

The feedback template aintpol2.tem results for square interpolation: (10)

$$A = \begin{array}{|c|c|c|}\hline 0 & 0.25 & 0 \\ \hline 0.25 & 0 & 0.25 \\ \hline 0 & 0.25 & 0 \\ \hline\end{array} \quad B = \begin{array}{|c|c|c|}\hline 0 & 0 & 0 \\ \hline 0 & 0 & 0 \\ \hline 0 & 0 & 0 \\ \hline\end{array} \quad z = \boxed{0}$$

The cost function for spline cubic CNN interpolation using only feedback template is given in the next equation: (11)

$$E(v(i,i)) = \sum_{i=1}^{M}\sum_{j=1}^{N}\left[(\frac{\partial^2 v}{\partial i^2})^2 + 2(\frac{\partial^2 v}{\partial i \partial j})^2 + (\frac{\partial^2 v}{\partial j^2})^2 \right]$$

The feedback template aintpol3.tem results for spline cubic interpolation:

$$A = \begin{array}{|c|c|c|}\hline -0.05 & 0.3 & -0.05 \\ \hline 0.3 & 0 & 0.3 \\ \hline -0.05 & 0.3 & -0.05 \\ \hline \end{array} \quad B = \begin{array}{|c|c|c|}\hline 0 & 0 & 0 \\ \hline 0 & 0 & 0 \\ \hline 0 & 0 & 0 \\ \hline \end{array} \quad z = \boxed{0}$$

(12)

3. TESTING CNN INTERPOLATION METHODS

In this section, the experimental results obtained by using the "CadetWin" (CNN Application Development Environment and Toolkit under Windows) [11] are presented.

Figure 1 shows the doubling process of image dimension horizontally and vertically, but keeping unchanged the size of pixels as in the initial picture.

The results of image reconstruction for images in which the known pixels are in irregular through CNN interpolation using the feedback templates aintpol1.tem, aintpol2.tem, aintpol3.tem are shown in Figure 2.

The same interpolation method could be used for reducing the volume of data sets in the case of data transmissions. This is achieved by reducing the dimension of images through coding, at the emitter, and decoding the encoded imageat the receiver.

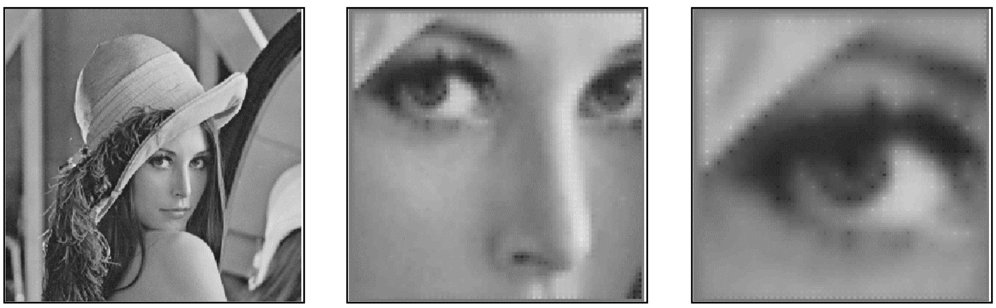
For encoding and decoding of the image, the interpolation method that uses the feedback template aintpol3.tem was used. In this way, the volume of data sets transmitted will be reduced by 25%, see Figure 3.

The feedback templates determined through CNN interpolation methods were tested on a 64*64 pixels dimension CNN-UC [3]. The results obtained for the image reconstruction using CNN interpolation, in which the known pixels are placed regularly, are shown in Figure 4. The pixels that have unknown values and the pixels with known values are inserted alternatively in the rows and columns. If the results obtained for the binary or gray-scale images interpolation in the case of simulating the method or testing the method on a chip are compared, it can be seen that better results are achieved with the feedback templates aintpol3.tem. This can be evaluated from the contrast of the image point of view.

In Table I the essential features for the CNN interpolation methods with the feedback templates aintpol1.tem, aintpol2.tem, aintpol3.tem are described from the processing time at simulation, and the processing time necessary for testing on a chip point of view.

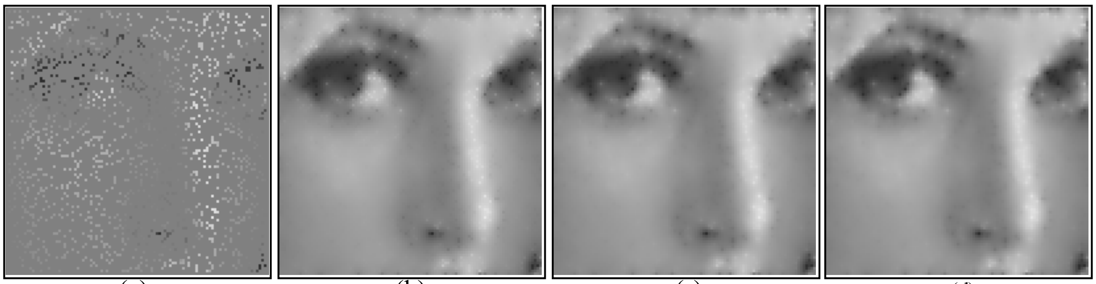
Table I: Essential features for CNN interpolation, by simulation and processing on a 64*64 chip.

Template	Simulation on a PC-300 MHz		Processing on a 64*64 CNN-UC
	Mean time for processing in τ_{CNN}	Mean complete time for processing	Mean complete time for processing
aintpol1	24	0.75s	3.4ms
aintpol2	24	0.75s	3.4ms
aintpol3	24	0.75s	3.4ms



(a) (b) (c)

***Figure 1**: Increasing resolution for gray-scale images using CNN interpolation with aintpol3.tem, through simulation: (a) initial gray-scale image; (b) result image obtained after the first enlargement; (c) result image obtained after the second enlargement.*



(a) (b) (c) (d)

***Figure 2**: Reconstruction of images using CNN interpolation methods through simulation: (a) input image, where only 15% of the values of pixels are known; (b) interpolation using aintpol1.tem; (c) square interpolation using aintpol2.tem; (d) spline cubic interpolation using the feedback template aintpol3.tem.*

(a) (b)

***Figure 3**: Using the feedback template aintpol3.tem for the encoding-decoding of images through simulation: (a) initial image before encoding, (b) image with rebuilt dimension after decoding.*

(a) (b) (c)

Figure 4: Image reconstruction using CNN interpolation with aintpol3.tem: (a) original image; (b) result image obtained through simulation; (c) result image achieved on a 64*64 CNN-UC

4. CONCLUSIONS

The interpolation of binary and gray-scale images using CNN only with linear feedback templates, dimensioned as a 3*3 matrix, can be achieved in real time by reducing the number of tests for the initial images on a chip. Due to the parallel processing on this chip, the running time stays unmodified even for the case of processing images having an increased dimension.

The interpolation methods analyzed above are useful for enlargement and interpreting the most important parts of biomedical images. The CNN interpolation methods could be used in reconstruction of partially known or deteriorated images.
Finally let us summarize some further directions of research in this field, namely:

- the robustness of the templates used in this contribution and the sensitivity to asymmetry should be analyzed;
- to implement an efficient method in image transmission;
- to optimize the parameters of the 64*64 CNN-UC.

5. ACKNOWLEDGEMENTS

The authors are thankful for the helpful discussion with Professor Tamás Roska.

REFERENCES

[1] L.O. Chua, L. Yang, "Cellular Neural Networks: Theory and Applications", *IEEE Trans. on Circuits and Systems*, Vol.35, pp. 1257-1290, 1988.
[2] T. Roska, L.O. Chua, "The CNN universal machine: an analogic array computer", *IEEE Trans. on Circuits and Systems*, Vol.40, pp. 163-173, 1993.
[3] G. Linan, S. Espejo, R. Dominguez-Castro, E. Roca, A. Rodriguez-Vazquez, "CNNUM3: A mixed-signal 64x64 CNN Universal Chip", *Proc. of Int. Conf. on Microelectronics for Neural, Fuzzy and Bio-inspired Systems (MicroNeuro99),* pp. 61-68, Granada, 1999.
[4] A. Gacsádi, P. Szolgay, "An analogic CNN algorithm for following continuously moving objects", *Proc. of IEEE Int. Workshop on Cellular Neural Networks and Their Applications, (CNNA 2000)*, pp. 99-104, Catania, 2000.
[5] W.E.L. Grimson, "*From images to surfaces-A Computational study of the human early visual system*", The MIT Press, London, England, 1983.
[6] P.L. Venetianer, F. Werblin, T. Roska, L. O. Chua, "Analogic CNN algorithms for some image compression and restoration tasks*", IEEE Trans. on Circuits and Systems,* Vol.42, pp. 278-284, 1995.
[7] Á. Zarándy, T. Roska, F. Werblin, L.O. Chua, "Intelligent image resolution enhancement by the CNN Universal Machine and its relevance to TV picture enhancement", *Proc. of Int. Symposium on Nonlinear Theory and Applications*, (NOLTA'95), Vol.2, pp. 701-706, Las Vegas, 1995.
[8] L. Nemes, G. Tóth, T. Roska, A. Radványi, "Analogic CNN algorithms for 3D interpolation-approximation and object rotation using controlled switched templates", *Int. Journal of Circuit Theory and Applications*, Vol 24, pp. 409-424, 1996.
[9] L.O. Chua, M. Hasler, G.S. Moschytz, J.Neirynck, "Autonomous cellular neural network: A unified paradigm for pattern formation and active wave propagation", *IEEE Trans. on Circuits and Systems,* Vol.42, pp. 559-577, 1995.
[10] W.E.L. Grimson "An implementation of a computational theory of visual surface interpolation", *Computer Vision, Graphics and Image Processing* 22, pp. 39-69, 1983.
[11] *"CadetWin",* User's Manual, Analogical and Neural Computing Laboratory, Computer and Automation Institute, Hungarian Academy of Sciences, Budapest, 1999.

Alexandru GACSÁDI received the Diploma and the Ph.D. degrees in electronics and telecommunications from the "Politechnica" University of Timişoara, Romania, in 1986 and 2001, respectively. He is currently an associate professor at the Electronics Department of the University of Oradea. His research interests include, image processing with cellular neural networks and its applications in robotics.
Péter SZOLGAY received the Diploma in electrical engineering from the Technical University in Budapest in 1974, and the Ph.D. degree in 1988, respectively. Currently, he is with the Analogic and Neural Computing Laboratory of Computer and Automation Institute of Hungarian Academy of Sciences. His research interests are in the areas of neural networks especially of cellular neural networks and its applications. He is presently teaching undergraduate courses on Analoque processor arrays and neural networks, on cellular neural networks applications and design methods based on programmable digital VLSI components in University of Veszprém.