Analogic CNN Computing : Architectural, Implementation, and Algorithmic Advances - a Review

Tamás Roska
Analogical and Neural Computing Laboratory,
Hungarian Academy of Sciences, H-1111, Kende u. 13-17., Budapest, Hungary.
Phone: +36-1-209-5263, FAX: +36-1-209-5264

ABSTRACT: *In this paper, first, an overview is given about the whole scenario of analogic CNN computing. Next, two areas on CNN Computing Technology are considered briefly: (i) the architectural advances, especially the variable resolution and adaptation in space, time, and value and (ii) the computational infrastructure from high level language and compiler to physical implementations. Three basic physical implementations are supposed : analogic CMOS, emulated digital CMOS and optical. The computational infrastructure is the same for all implementations, except the physical interfaces.*

1. Introduction

A few months ago, Intel shipped the first Tera FLOPS supercomputer consisting almost ten thousand 200 MHz Pentium microprocessors. In many image processing applications we really need this trillion operations per second, except the operations are special and do not require the 32 bit floating point accuracy. The alternative is the analogic CNN array computer performing about Tera equivalent operations per second, however, on a single chip.

Ten years ago, in the seminal, paradigm forming, and now historic paper L.O.Chua and L.Yang [1] introduced the Cellular Neural Network (CNN), now we call also Cellular Nonlinear Network, as a 2D or 3D regular array of locally interconnected nonlinear dynamic systems called neurons, or cells, whose global functionality is determined by a small number of parameters. These parameters define the local interconnection pattern, called *cloning template.* Once the cell is given the cloning template, or simply the template, specify the operation of the whole array. The cloning template is the protagonist in CNN. It is like a *gene* for spatio-temporal dynamics. Using very simple cells, even first order ones, practically all the simple and exotic spatio-temporal dynamic phenomena can be generated by „engineering" the cloning template. Like genes, cloning templates can define a whole universe of phenomena. Designing this template we can engineer this universe. Many useful templates were designed to implement useful image processing functions and it was shown early that quite a few neuromorphic models of the visual pathway can be represented by CNN models [8]. Due to local connectivity, CNN is very convenient in VLSI design.

The invention of the CNN Universal Machine architecture [3] put the CNN dynamics into a different perspective: the CNN spatio-temporal dynamics, via the cloning template, became the atom, the elementary instruction of a stored program in this new computational paradigm. A new world of analogic algorithms and software has been developing.

In this paper, first, an overview is given about the whole scenario of analogic CNN computing. Next, two areas on CNN Computing Technology are considered briefly: (i) the architectural advances, especially the variable resolution and adaptation in space, time, and value and (ii) the computational infrastructure from high level language and compiler to physical implementation. Three basic physical implementations are supposed : analogic CMOS, emulated digital CMOS, and optical.

In section 2, the main areas in forming the analogic CNN computing paradigm are described and their relation is shown. The main message is: three facts and trends, namely, (i) the implementation of morphology and PDE based algorithms, (ii) the present understanding of the living visual pathway, especially the retina, and (iii) the physical limitations of computing devices, all prove the need of this new computational framework. Section 3 deals with natural extension of the CNN-UM architecture to incorporate variable resolution and adaptation in space, time and signal values. In Section 4, the computational infrastructure is outlined; the same high level language, called Alpha, is used for all the different implementations.

2. The world of analogic CNN computing

In Figure 1, the main areas underlying and forming the analogic CNN computer are shown.

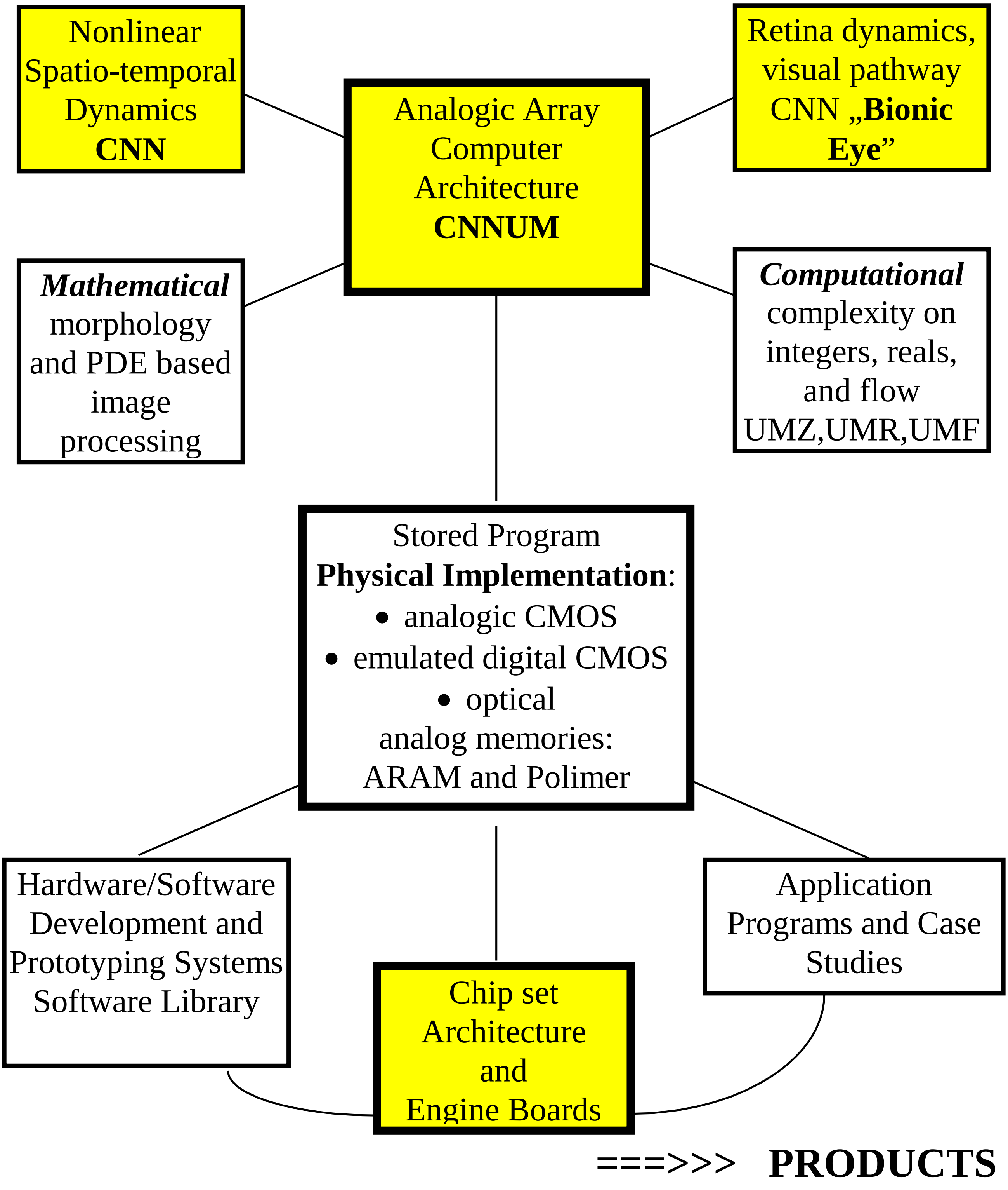

Figure 1. The main interacting areas in forming the analogic CNN computing paradigm

The three main pillars are

• the nonlinear spatio-temporal dynamics defined by the CNN (see e.g. the very recent account [29]),

• the CNN-UM architecture [3] as the computing framework and the analogic algorithms executed on this platform,

• the physiologically faithful CNN models of the visual pathway, especially the retina, embedded in the „Bionic Eye" multimodal framework implemented on the CNN-UM [6].

Recently, it became clear that the classical and well developed framework in image processing is not enough powerful to solve complex image processing and pattern recognition tasks. The new mathematical techniques, *mathematical morphology and especially the PDE related methods [22,24]*, however, need enormous computing power when implemented on standard, even parallel digital computers. On the other hand, it turned out that most of these methods are almost native in the analogic CNN computers [19, 14-16].

Computational complexity has been well studied and is directly related to the standard digital computers. Recently, computational complexity studies on reals (due to Blum, Shub,and Smale) challenged this framework by showing its limits when numerical algorithms on reals are considered. The Universal Machine on Integers (UMZ) is replaced by a Universal Machine on Reals (UMR) using the so called Newton Machine, which (by nature) remains iterative. The CNN-UM is , however, a continuous time, continuous value machine operating on flows (UMF) . The starting studies show the relation between UMZ, UMR, and UMF [21].

The first physical implementations of the CNN-UM architecture [4,5] proved the physical realizability of the concept. The 22x20 chip [4] has been able even to robustly classify slightly different textures [28 The new, more complex chips [25,26] increase the complexity into the 50x50 array domain and the forecasts for 1999 pointing towards chips over the 100x100 barrier. In addition, the first analog buffer data storage, the ARAM, has also be designed [27] providing the omly missing, non-standard chip in the CNN chipset and Engine Board [7].

The hardware and software development systems and software library provide the same computational infrastructure as for standard digital computing, and the chipset architecture and Engine Board make this technology completely transparent to digital systems as well as to bigger sensor arrays, respectively. We will deal with these questions in more details in Section 4.

After the first application case studies [17,18,20, etc.], real-life applications has already been started using emulators and prepare the way to products using more complex chips and Engine Boards.

3. Variable resolution and local adaptation

In what follows, we are introducing:

spatially global **Variable Resolution** in

- *space*
- *time*
- *signal value, and*
- *cell dynamics*

as well as

locally **Adaptive Space-variant** templates by four ways:

- *global template control* by downloading images
- *time invariant local control* via local template control memories (TCM)
- *slowly time varying plasticity* via local template control signals (TCS)
- *real - time local adaptation*

It is emphasized that all these features are introduced in such a way that the basic advantages of the CNN UM architecture remains valid. Recent studies on associative memory, learnong, and fuzzy CNN [9,10,17] show the need for these possibilities.

3.1 Variable resolution

Variable resolution is defined as follows:

- **space**

  variable grid size (***fine/coarse***); a few switches are needed in the analogic CMOS implementation, typical values for the grid ratios are 1: 3-5

- **time**

  fast and slow CNN time constants; typical values $\tau_{CNN} = 1$ and 5(10)

- **signal value**

  multi shade , *multi spectral (color), multi polarization* in the analogic and/or optical implementation, *multiple bit length* and exponent length in the emulated digital implementation

- **cell dynamics**

  complex cells; 1^{st} , 2^{nd}, 3^{rd} order or multi-layer first order cells are defined in a complex cell containing one, two or three state variables (capacitors). In the CNN-UM , the complexity of the extended cells are only slightly increased; using more LAM units, it will even remain constant.

3.2 Adaptive space variant templates

The introduction and definition of the ***locally*** **Adaptive Space-variant** templates is given as follows.

- **global template control** is made by downloading images, e.g. areas with predetermined different bias terms.
- **time invariant local control** is made via local template control memories (TCM).

The TCM values are calculated by template operations, then used in a ***Plasticity Rule*** to determine local template elements, e.g. local illumination is calculated and set to the local bias terms (z). The key issue is that the number of independently adapting values are small; instead of 19, one or two. For example, in a 4-element LAM if we have 2 TCM values (c1, c2) in a LAM: LAM [1-4] : a1 a2 c1 c2

then we may use e.g. z = c1

or A or B =

c2	c1	c2
c1	aoo or boo	c1
c2	c1	c2

- **slowly time varying plasticity** is governed by local template control signals (TCS).

During, or within, a finite potentiation time Tp, a ***Potentiation Rule*** operator (implemented in the LAOU) will determine the TCS value. Then this value is used in the ***Plasticity Rule*** to determine the change of the template element values. For example, decreasing or increasing the template value via the TCS, as a result of Short Term Potentiation (STP) or Long term Potentiation (LTP) in neuromorphic models as typical Potentiation Rules.

As in case of TCM, there is only one or two TCS value. For example, in a 4-element LAM we may have one TCM and one TCS value:

LAM [1-4]: a1 a2 c1 p1

TCM TCS

The *Potentiation Rule* operator is implemented in the Local Analog (Output) Unit, e.g. by a *Potentiation Rule circuit* controlled by the CNN cell variable (input, state, or output), and resulting in the Template Control Signal ***p1***. Again, p1 ***controls the template elements locally*** via the Plasticity Rule.

- **real - time local adaptation** means the local template control signal acts immediately, for example, the local illumination value controls the bias term

4. Computational infrastructure

The analogic CNN computers are built up using the *CNN Chip set architecture* shown in Figure 2.

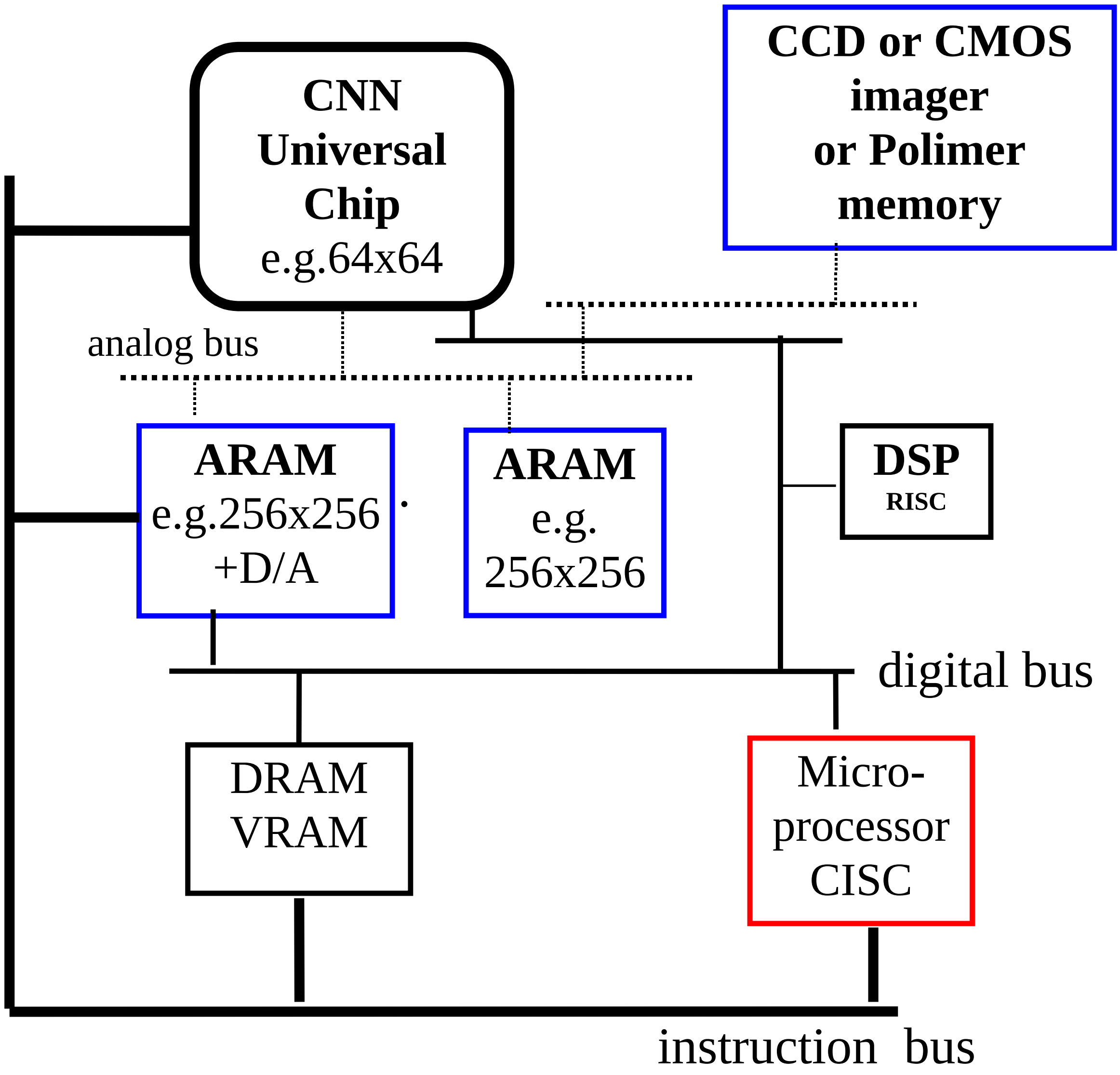

Figure 2. The CNN Chipset Architecture

This architecture when implemented as a self contained *Engine Board* or *Engine on a Chip* or an Engine Board connected to a PC, will be programmed by the high level analogic CNN language, called Alpha. The same Alpha source code is used for all other CNN implementations, including a single CNN-UM chip, an emulated digital CNN-UM chip, or a software simulator, etc..

For the time being, the CNN Applications Development Environment and Toolkit (CADETWin) [11] and the CNN Chip Prototyping System (CCPS) [12] are the *designer's frameworks and tools* for using the Alpha language. On Figure 3 we show the various levels of descriptions and controls representing an analogic CNN algorithm.

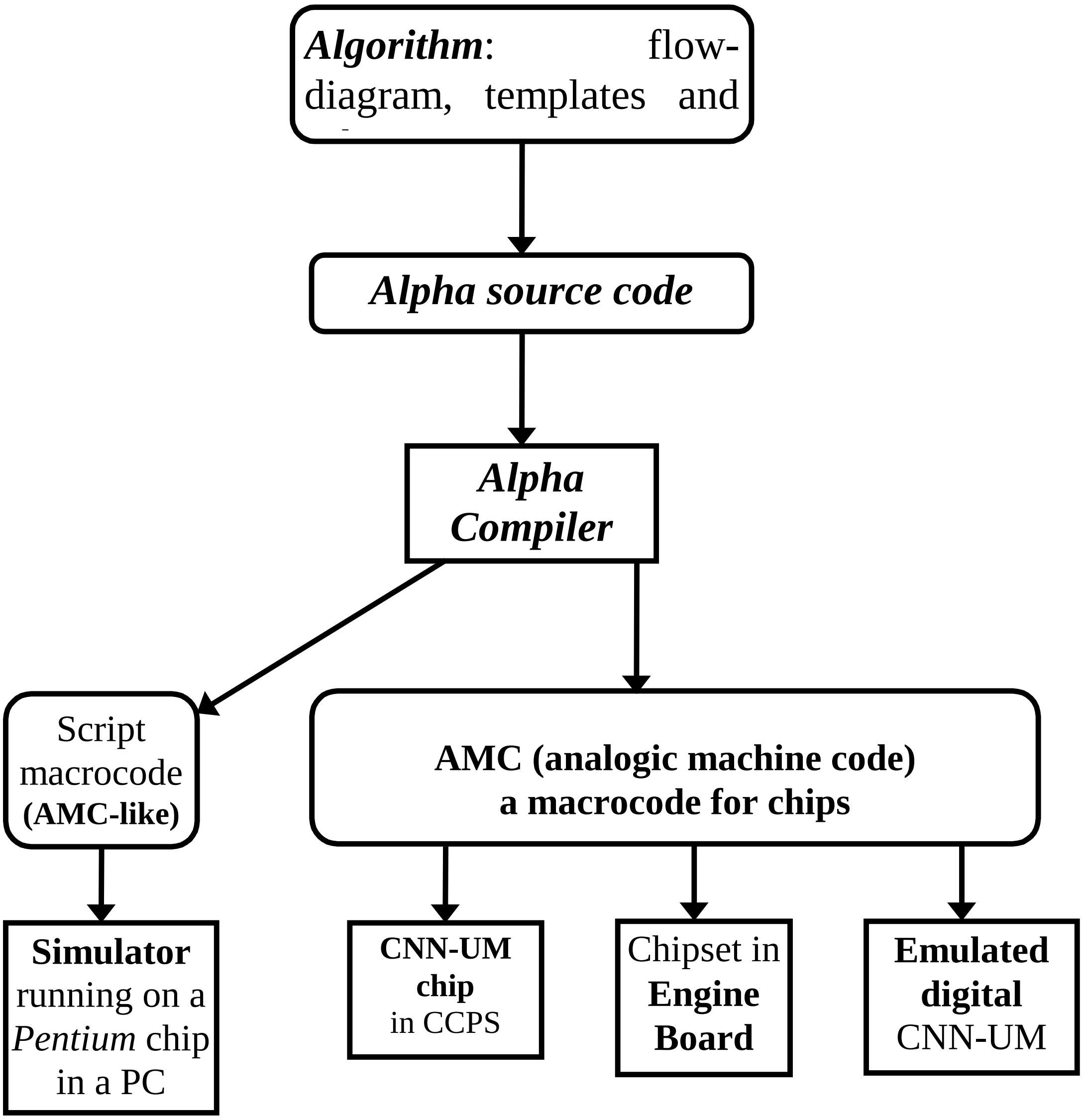

Figure 3. The levels of the software and the core engines

On the lowest level, the chips are embedded in their physical environment The AMC code will be translated into firmware and electrical signals.

The CNN *chips can be tested and prototyped* for specific applications by using the CNN Chip Prototyping System. The key advantage of this system is that it is invariant for all different chips, except the Platform hosting the chip (or chipset). This Platform has an interface (CNN Physical Interface, CPI), CNN chip designers can use it to design their own Platforms and then the whole arsenal of Alpha programs, subroutines , etc. can be used to program their chips.

5. Acknowledgments

This research has been supported by the Grant No. T002578 of the Hungarian National Science Fund (OTKA), by the Grant No. N68171-97-C-9038 of the Office of Naval Research, and by the Hungarian Academy of Sciences. The comments of Péter Szolgay, Ákos Zarándy, Tibor Kozek, Cs. Rekeczky, Angel Rodríguez-Vázquez, and Ricardo Carmona are gratefully acknowledged.

6. References

[1] L. O. Chua and L. Yang, „Cellular neural networks: Theory and Applications", *IEEE Transactions on Circuits and Systems*, Vol.35, pp.1257-1290, 1988.

[2] L. O. Chua and T. Roska, „The CNN Paradigm", *IEEE Transactions on Circuits and Systems-I*, vol.CAS-40, pp.147-156, March 1993.

[3] T. Roska and L. O. Chua, „The CNN Universal Machine: An Analogic Array Computer", *IEEE Transactions on Circuits and Systems-II*, vol. 40, pp. 163-173, March 1993.

[4] R. Dominguez-Castro, S. Espejo, A. Rodriguez-Vazquez, R. Carmona, „A CNN Universal Chip in CMOS Technology", *Proc. of the third IEEE Int. Workshop on Cellular Neural Networks and their Application (CNNA-94)*, pp. 91-96, Rome, 1994.

[5] J. M. Cruz, L. O. Chua, and T. Roska, „A Fast, Complex and Efficient Test Implementation of the CNN Universal Machine", *Proc. of the third IEEE Int. Workshop on Cellular Neural Networks and their Application (CNNA-94)*, pp. 61-66, Rome Dec. 1994.

[6] F.Werblin, T.Roska, and L.O.Chua, „The analogic cellular neural network as a bionic eye", *Int. J. Circuit Theory and Applications*, Vol. 23, pp.541-549, 1995

[7] T.Roska, „CNN Chip set architecture and the Visual Mouse", in *Proceedings of 4th International Workshop on Cellular Neural Networks and Their Applications*, pp. 369-374, Seville, June 1996.

[8] T.Roska, J.Hámori, E.Lábos, K.Lotz, L.Orzó, J.Takács, P.Venetianer, Z.Vidnyánszky and Á.Zarándy, "The Use of CNN Models in the Subcortical Visual Pathway", Special Issue on Cellular Neural Networks, *IEEE Trans. Circuits and Systems-I*, Vol.40. No.3. pp. 182-195, March 1993

[9] T. Kozek, T. Roska, L. O. Chua, „Genetic Algorithm for CNN Template Learning", *IEEE Transactions on Circuits and Systems*, vol. 40, no. 6, pp.392-402, 1993.

[10] P.Szolgay, I.Szatmári, and K.László, "A fast fixed point learning method to implement associative memory on CNN's", *IEEE Trans. on Circuits and Systems*, Ser. I., Vol. 44, pp. 362-366, 1997

[11] CADETWin (CNN Applications Development Environment and Toolkit) , User's Guide, Analogical and Neural Computing Laboratory, Computer and Automation Institute, Hungarian Academy of Sciences (MTA-SzTAKI), Budapest, 1997.

[12] CCPS; (CNN Chip Prototyping System), , User's Guide, Analogical and Neural Computing Laboratory, Computer and Automation Institute, Hungarian Academy of Sciences (MTA-SzTAKI), Budapest, 1997.

[13] „CNN Software Library (Templates and Algorithms) Version 7.1", Edited by T. Roska, L. Kék, L. Nemes, Á. Zarándy and M. Brendel, Computer and Automation Institute of the Hungarian Academy of Sciences, Budapest, 1997.

[14] P.Szolgay, G.Vörös, and Gy. Erôss, „Applications of the Cellular Neural Network paradigm in Mechanical Vibrating Systems", *IEEE Transactions on Circuits and Systems-I*, vol. 40, pp. 222-227, March 1993

[15] T.Roska, L.O.Chua, D.Wolf, T.Kozek, R. Tetzlaff, and F.Puffer, „ Simulating Nonlinear Waves and Partial Differential Equations via CNN - Part I: Basic Techniques", ibid, Vol.42, pp. 807-815, October, 1995

[16] T.Kozek, L.O.Chua, T.Roska, D.Wolf, R.Tetzlaff, F.Putter, and K.Lotz, „„„ Simulating Nonlinear Waves and Partial Differential Equations via CNN - Part II: Typical examples", ibid, Vol.42, pp. 816-821, October, 1995

[17] Cs. Rekeczky, Á.Tahy, Z.Végh, and T. Roska, "CNN-based spatio-temporal nonlinear filtering and endocardial boundary detection in echocardiography", *Proc. ECCTD-97*, Vol. II, pp.667-672, 1997, *Int. J. Circuit Theory and Appl.*, to appear

[18] T. Kozek, Chai Wah Wu, Á. Zarándy, Hua Chen, T. Roska, M.Kunt, and L.O.Chua, "New Results and Measurements Related to Some Tasks in Object-Oriented Dynamic Image Coding Using CNN Universal Chips", *IEEE Trans. on Circuits and Systems for Video Technology*, Vol. 7. No.4. pp.606-614, August 1997

[19] Á. Zarándy, A. Stoffels, T. Roska, and L. O. Chua: „Implementation of Binary and Gray-Scale Mathematical Morphology on the CNN Universal Machine”, *ibid*, Vol. 8, 1998, in print.
[20] A. Schultz, Cs.Rekeczky, I. Szatmári, T. Roska, and L.O.Chua, "Spatio-temporal CNN algorithms for object segmentation and object recognition”, in this Proceedings
[21] T.Roska and L.O.Chua, „On a framework of complexity of computations on flows implemented on the CNN Universal Machine", Technical Report, DNS-15-1995, Computer and Automation Institute, Budapest, 1995
[22] L.Alvarez and J-M.Morel, „Morphological approach to multiscale analysis”, pp. 229-249 in B.M.H.Romeny (ed), Geometry-driven diffusion in computer vision, Kluwer Academic Publishers, 1994
[23] P.Maragos, „Differential morphology and image processing”, *IEEE Trans. Image Processing*, Vol. 5, pp.922-937, 1966
[24] R.Malladi and J.A.Sethian, „A unified approach to noise removal, image enhancement, and shape recovery”, *ibid*, pp.1554-1568, 1996
[25] A. Paasio, A.Dawidziuk, K.Halonen, and V. Porra, „Minimum-size 0.5 micron CMOS Programmable 48 by 48 Test Chip”, Proceeding of the ECCTD-97, pp. 154-156, 1997
[26] A.Rodriguez-Vazquez (personal communication)
[27] R.Carmona, A. Rodriguez-Vázquez, T.Roska, T.Kozek, L.O.Chua, „The first CNN-ARAM Chip: a 0.5 micron CMOS Analog Random Access Memory for Massive Image Processing”, *in this Proceedings*
[28] T. Szirányi, M. Csapodi, „Texture Classification by CNN and Genetic Learning”, *Proc. of IEEE Int. Conf. Pattern Recognition*, Jerusalem, ICPR’94, vol. III, pp. 381-383, 1994.
[29] L.O.Chua, „CNN - a Paradigm for complexity, Part I, Part II, and Part III” , *Int. J. Bifurcation and Chaos*, Vol.7, August, 1997