Brendel Mátyás

Plasticity in the Cellular Neural Networks
CNN-backpropagation, adaptive image sensing and enhancement

Theses of the Ph.D. dissertation

Analogical and Neural Computing Systems Laboratory
Computer and Automation Institute
Hungarian Academy of Sciences

Scientific adviser: Tamás Roska, D.Sc.

Budapest, 2003.

Introduction

"Plasticity", the notion comes from neurobiology and is not commonly used in computer science, however I think, it would be fruitful. Plasticity of the nervous system consists of several aspects: learning, redundancy, damage recovery, robustness under several conditions and against several kind of stimuli, i.e. adaptivity. Since these functions are interweaved, it is plausible to speak about an emergent feature consisting all of them: plasticity.

The first computer hardware and software showed very low plasticity. They worked for only very specific inputs. If there was a faulty input or some internal error, the result could be completely wrong. As information technology develops, the demands, which are required for a computing device became higher and higher. The advance of technology could fulfill several of these demands: hardware and software, which can learn automatically, overcome minor failures, handle several inputs in an intelligent way appear.

"Artificial intelligence" is not yet developed, and I think it would never be completed in that naive way as science-fiction presents it. But it is obvious that several great results were achieved, some of them are hardly believable for people unqualified in this topic. We are eyewitness now breakthroughs in the tasks of artificial intelligence. And the advance in the future will be presumably more compelling. I think, a major aspect of this advance will be related to plasticity.

Another part of development is related to alternative computing devices, parallel computing, neural nets, fuzzy systems and other advanced technologies. Cellular Neural Network (CNN) is one of the most challenging ideas in computer sciences ([9]), which is also related to this line of development. It is basically an analog technology, but the CNN Universal Machine (CNN-UM) additionally integrates analog and logical computing (analogic computing)([10]). CNN can as well be considered as a special kind of neural network, which is part of one of the most genuine result of computer science: the connectionist paradigm. The idea of CNN is also inspired by neurobiology, especially by the low-level part of the visual system. Thus CNN combines several of the advances of science.

Teaching and learning of CNN was a major topic of research since the very start of CNN history ([13]). The theory of learning of CNN has many difficulties to solve. Learning is always also an optimization task, and the difficulty of global optimization is a well-known problem, which is addressed by several kinds of solutions. It is one of the most diversified parts of applied mathematics with several unsolved problems and some presumably unsolvable ones as well. Another aspect, which is always to be considered and to be handled, is the high demand of computing.

CNN as a parallel and analog computing device can realize an exceptional computing power. But this capability has not yet been used for the computing demand of learning of CNN itself. The first part of my dissertation (thesis 1), CNN learning, is based on this idea.

Plasticity is not only related to learning. Another aspect is the intelligent handling of the incoming data: adaptivity. As the physical realizations of CNN is a 2D processor array, the main application area is plausibly image processing (or other 2D-topological signal array-"images"). Incoming data are acquired by some kind of image capturing device. Thus the second important issue is high-quality image capturing.

One of the most important problems of image capturing is that the devices do not operate properly in extreme lighting circumstances. For example the photo and video cameras can not acquire satisfactory images if lighting is too strong or too weak. It is possible to adjust manually the global parameters of the cameras, but this is not a satisfactory solution. An automatic solution has been lately developed, but this is not sufficient if lighting conditions also change extremely in space. In this case spatial adaptation capability is also needed.

The present sate of art is that the devices become more and more intelligent, and they can solve many problems. For example some global parameters, like brightness are adjusted automatically. At the same time there are serious hardware limitation when spatial adaptivity is considered. Applying CNN can provide a solution for these computational demands and the adjustment of the global parameters can also be utilized withal. Another advantage is that it is a general solution, i.e. additional image processing tasks can be programmed, if it is already used in a camera.

The second part of my dissertation (thesis 2) deals with adaptive image sensing, using adaptive CNN-UM. This is an application, which I think could be one of the most promising ones. Moreover the solution I suggest is completely realizable by the current architectures.

Image enhancement is a very important field of image processing, which has already a considerable history. Several digital algorithms have been developed, which are already parts of the best image processing programs. They provide relatively good solutions but also have their limits. The advantage of applying CNN technology, is the speed of computation, which is a crucial problem by digital, serial devices. Integration of sensing and processing - especially when the sensors are tuned according to the processed content and context of the scene - can also be realized better this way.

The goal of my research was to start with simple methods, which can be realized with current technologies or architectures, planed in the near future ([15]). One of the main field of architectural development of CNN technology lies in multilayer/complex cells, where adaptive methods can be exploited.

The two algorithms for adaptive image enhancement I presented in thesis 3 are closely related to these possibilities. They are not theoretically optimal solutions, but a compromise of the complexity of the method and realizability.

My dissertation is closely related to neurobiology ([12]). Specifically the methods are biologically inspired, but they cannot be considered as models of biological functions. The goal of my research was to develop profitable application, not neurobiological simulations. Nevertheless, since the methods are biological inspired, the dissertation also includes a sketch of those areas, which are related to my research.

Methods used in the experiments

As a theoretical background of my research the general theory of CNN is the main topic. In the subject of learning of CNN, the general theory of learning of neural networks was utilized ([14]). Since the methods are biologically inspired the fundamentals of neurobiology are considered. The theory of image enhancement technologies is a background of theses 2 and 3.

At the beginning of research, the methods where tested by simulations. I used the simulation tools developed by the Analogical and Neural Computing Laboratory of the Computer and Automation Institute of the Hungarian Academy of Sciences, ALADDIN. I also had the opportunity to test this methods on the ACE4k CNN-UM chip. For some simulation, I used the general mathematical software MATLAB. The image-capturing devices, which were used, were Sony camera, with programmability in language C included and a commercial digital camera.

For the opto-electronic solution of adaptive sensing an opto-electronic system was used, which was designed in our opto-electronical laboratory.

New scientific results (theses)

1. **Gradient-computation of the template-parameters of the Single-Layer Cellular Neural Network with linear template by using the CNN-UM itself.** ([1],[5] and chapter 4.)

1.1. Gradient-Computing in the case of Discrete-Time CNN

Technical preliminaries: Derivatives of the Discrete-Time CNN

I developed the exact analytical formulas of the derivatives, for each type of the template-parameters.

If x is the state of the original CNN, y is the output and p is a parameter, the following notations are introduced:

$$\mathrm{dpy}_{\mathrm{i,j}}(\mathrm{n}) = \frac{\mathrm{dy}_{\mathrm{i,j}}(\mathrm{n})}{\mathrm{dp}}$$

$$\mathrm{dpx}_{\mathrm{i,j}}(\mathrm{n}) = \frac{\mathrm{dx}_{\mathrm{i,j}}(\mathrm{n})}{\mathrm{dp}}$$

Now, the equations of derivative computing are the following. First the derivative of the output equation is:

$$dpy_{i,j}(n) = f'(x_{i,j}(n))dpx_{i,j}(n) \tag{1}$$

Then, the derivative of the network equation is depending on the type of the parameter.

$$dpx_{i,j}(n+1) = y_{i+\nu,j+\mu}(n) + \sum_{(k,l)\in S_r(i,j)} A(k-i,l-j)dpy_{k,l}(n) \tag{2}$$

if p= A(ν,μ),

$$dpx_{i,j}(n+1) = u_{i+\nu,j+\mu}(n) + \sum_{(k,l)\in S_r(i,j)} A(k-i,l-j)dpy_{k,l}(n) \tag{3}$$

if p= B(ν,μ) and

$$dpx_{i,j}(n+1) = 1 + \sum_{(k,l)\in S_r(i,j)} A(k-i,l-j)dpy_{k,l}(n) \tag{4}$$

if p=z.

1.1.1. Computation of the gradient with CNN

I have shown that the adjoint (reciprocal) network of a single layer DT-CNN with linear templates computing the gradient of its parameter is also a single layer DT-CNN with linear templates, but with different output equation and modified inputs. Thus the gradient of the DT-CNN can be computed with another CNN or with itself.

Equations (1)-(4) are describing 3 types of DT-CNNs with state variable dpx, output variable dpy, input variable y,u or 0 (depending of the type), with special templates, derived from the original ones and special output function, using the state of the original network. These, so called adjoint networks can compute the derivative of the original network.

The mapping from the original network to the adjoint network can be obtained from equations (2),(3) and (4), it is as given in Table 1.

Parameter p	Original network	Adjoint network, CNN_p
$p=A(\nu,\mu)$	A,B,z,x,y,u,f	$A,E_{\nu,\mu},0,dpx,dpy,y,f'(x_{i,j}(n))\cdot$
$p=B(\nu,\mu)$		$A, E_{\nu,\mu},0,dpx,dpy,u,f'(x_{i,j}(n))\cdot$
$p=z$		$A,0,1,dpx,dpy,0,f'(x_{i,j}(n))\cdot$

Table 1. Mapping from the original CNN to the adjoint CNN. $E_{\nu,\mu}$ is a template with 1 on position (ν,μ) and 0 elsewhere. "$f'(x_{i,j}(n))\cdot$" means multiplication with $f'(x_{i,j}(n))$.

The gradient can be computed by a simple formula from the derivatives:

$$\frac{dE(p)}{dp} = -\sum_i \sum_j (d_{i,j} - y_{i,j}(T)) \frac{dy_{i,j}}{dp}(T) = -\sum_i \sum_j (d_{i,j} - y_{i,j}(T)) dpy_{i,j}(T)$$

where d is the desired output, y is the output and T is the time, when the final output was taken.

1.1.2. The learning architecture

Figure 1 shows an architecture capable of gradient-based learning. Note that CNN_A represents the 9 A-type reverse networks, CNN_B represents the 9 B-type reverse networks and CNN_Z represents the z-type reverse network.

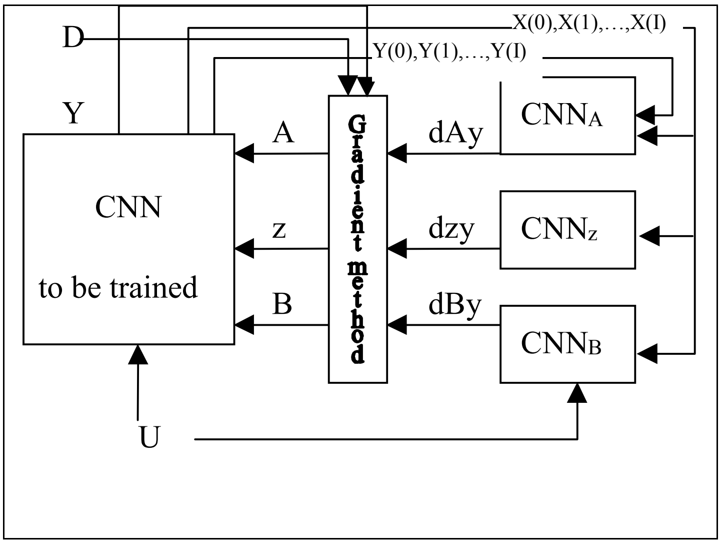

Figure 1 Supervised training of DT-CNN with reciprocal DT-CNNs. D is the desired output, Y(i) is the output at time i, and X(i) the state at time i, U is the input as a matrix, dAy, dBy and dzy are the derivatives, and CNN_A, CNN_B and CNN_z represents the reciprocal networks computing this gradients respectively.

1.2. Gradient computing in the case of Continuous-Time CNN

Technical preliminaries: Exact analytical formulas of the derivatives for each type of parameters

I developed the exact analytical formulas of derivatives, for each type of the template-parameters.

If x is the state of the original CNN, y is the output and p is the parameter, the following notations are introduced:

$$\delta p y_{i,j}(t) = \frac{d y_{i,j}(t)}{dp}$$

$$\delta p x_{i,j}(t) = \frac{d x_{i,j}(t)}{dp}$$

If p=A(m,n), then the derivative of the network equation is as follows:

$$\frac{d}{dt}\delta p x_{i,j}(t) = -\delta p x_{i,j}(t) + \sum_{k,l\in S(i,j)} A(k-i, j-l)\delta p y_{k,l}(t) + y_{i+m,j+n}(t) \tag{5}$$

If p=B(m,n), as follows:

$$\frac{d}{dt}\delta p x_{i,j}(t) = -\delta p x_{i,j}(t) + \sum_{k,l\in S(i,j)} A(k-i, j-l)\delta p y_{k,l}(t) + u_{i+m,j+n}(t) \tag{6}$$

and finally if p=z, as follows:

$$\frac{d}{dt}\delta px_{i,j}(t) = -\delta px_{i,j}(t) + \sum_{k,l\in S(i,j)} A(k-i, j-l)\delta py_{k,l}(t) + 1$$

(7)

The derivative of the output equation is as follows.

$$\delta py_{i,j}(t) = \frac{\delta f}{\delta x}(x_{i,j}(t))\delta px_{i,j}(t)$$

(8)

1.2.1. Computation of the gradient with the CNN itself

I have shown that the adjoint (reciprocal) network of a single layer CT-CNN with linear templates computing the gradient of its parameters is also a single layer CT-CNN with linear template, but with different output equation. Thus the gradient of the CT-CNN can be computed with another CNN.

Equations (5)-(8) are describing 3 types of CT-CNNs with state variable dpx, output variable dpy, input variable y,u or 1 (depending of the type), with special templates, derived from the original ones and special output function, using the state of the original network. These, so called adjoint networks can compute the derivatives of the original network.

The mapping from the original and the adjoint network is the same as in the case of DT-CNN see Table 1.

The gradient can be computed through a simple formula from the derivatives as follows:

$$\frac{dE(p)}{dp} = -\sum_i \sum_j (d_{i,j} - y_{i,j}) \frac{dy_{i,j}}{dp}(T) = -\sum_i \sum_j (d_{i,j} - y_{i,j}) dpy_{i,j}(T)$$

where d is the desired output, y is the output and T is the time, when the final output was taken.

1.2.2. The learning architecture

In typical CNN implementations the output function f(x)=0.5(|x+1|-|x-1|), which is equal to the identity function in the range of [-1,1]. Thus in this so called „linear region", the output function of the original and adjoint (reverse) network are the same, therefore the reverse network can also be implemented with the current technologies.

The method of exact gradient computation requires one additional CNN chip for each gradient (each parameter) and the continuous feeding of the output of the original network as the input of the adjoint (reverse) network (Figure 2).

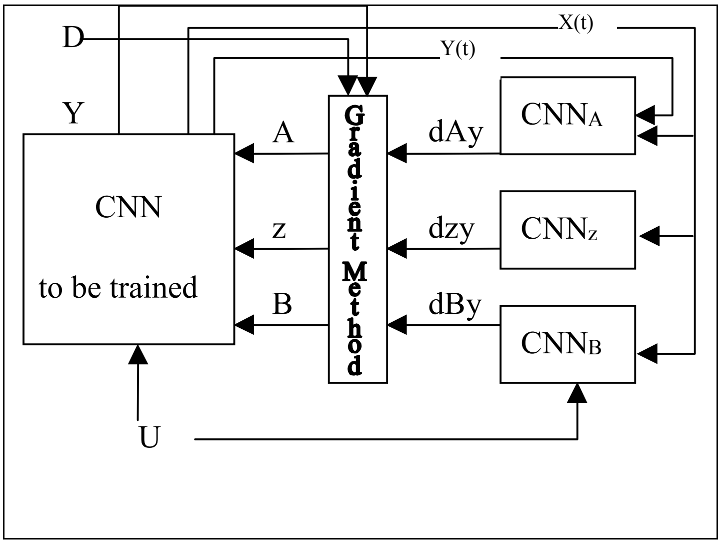

Figure 2 Flowchart of computing the gradients of a CNN template. D is the desired output, X the state and Y the output as a matrix. The CNN_A and CNN_B notations denote several adjoint (reverse)-CNNs computing the gradient belonging to the parameters in the A and B template respectively. The CNN_z notation denotes the adjoint (reverse) CNN computing the gradient for the parameter z.

1.2.3. On-chip approximation of the gradients

Using the CNN Universal-Machine I developed an approximate computation for the gradient computation and tested it on the current chip.

If separate chips are available for the original CNN and the adjoint CNNs, the gradient computation is accurate. However, using the approximation the gradients may be computed with one chip by a CNN-UM algorithm. The continuous data of the original transient is approximated by quantization in time (see Figure 3). The problem in computing the gradients on one CNN-UM is, that the transient in continuous time can not be saved for the gradient computation. It is obvious that the transient can be discretized in time and so saved in memories. I have tested this, and it turned out to be adequate.

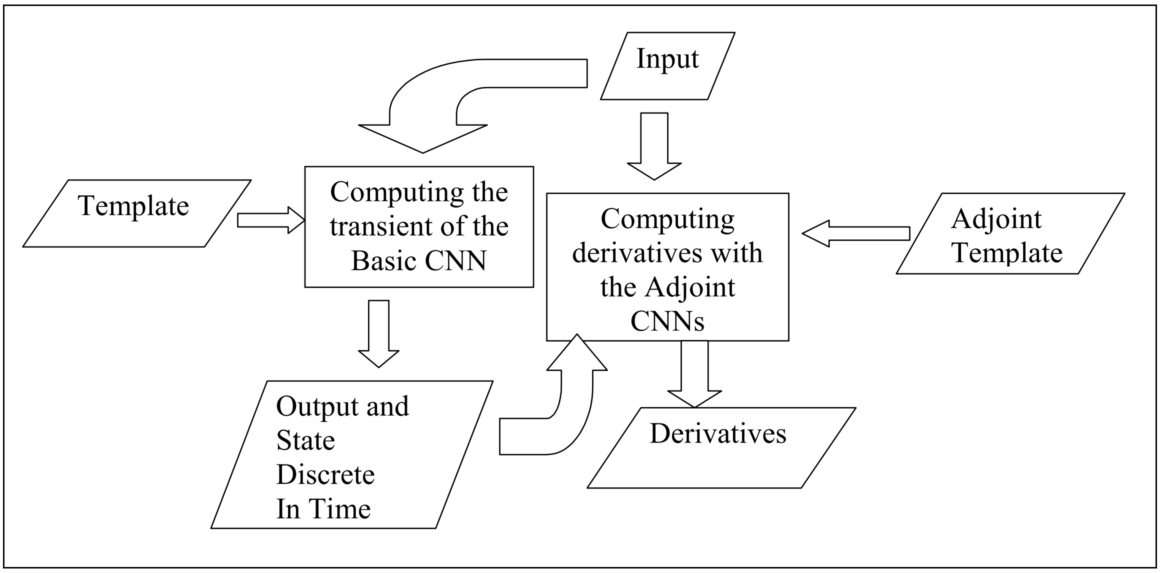

Figure 3 Approximating the gradient in one chip via a CNN-UM algorithm and discretization in time. The output and state of the basic CNN is saved in analog memories (LAM) and used in the adjoint CNN. The adjoint CNNs have an adjoint templates, which depend on the original template.

2. **Interactive Content Dependent (CDA) Adaptive Image Sensing** ([2],[3],[4],[6] and chapter 6.)

I introduced an interactive, content dependent, adaptive image sensing method. The method addresses image sensing, which depends on some image-content. It is assumed that a locally programmable sensing device is available. The method generally describes how to program devices locally using the regional quality of the image. The result of the method is an image, where this quality becomes enhanced.

I developed the image capturing method for scenarios where lighting and contrast is changing in space. The capturing method is adaptive and dependent on regional quality, which is based on both intensity and contrast content. The method equalizes contrast and intensity and avoids over-saturation. Determination of regional quality is based on a CNN algorithm, using contrast and diffusion templates.

The method is interactive, because sensors are locally adjustable (programmable). It is content dependent, since the programming depends on the local image content. Finally, it is adaptive, as programming is carried out in order to make sensing adaptive to regional changes. With this method the specified content is enhanced. The general method can be described as follows:

(i) Capturing with preset parameters

(ii) Content dependent regional quality is computed real-time and pixel-wise

(iii) The global and local sensing parameters are adjusted, using the regional quality information

Depending on the parameter and technical details, the interaction of steps (i)-(iii) may be continuous or discrete. For example if the parameter is local contrast, then a picture must be taken with low but uniform contrast, and the local parameter of the final exposure is computed from this. On the other side, if the parameter is exposure-time, then the exposure can locally and continuously controlled, i.e. locally stppeed, if the pixel is has sufficient quality.

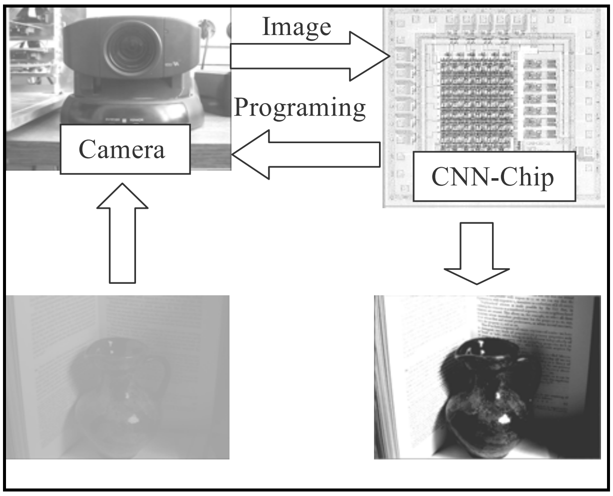

Figure 4 Interactive Content Dependent Adaptive Image Sensing with loally programmable capturing device. The capturing device (camera) is programmed using the information, extracted by the CNN from the image. Thus an enhanced image is captured, which is better in quality.

The application of CNN-UM is justified by the large demand on resources, which is needed for computing regional quality. At present digital technology global and local information can only be used in adaptive devices, which is insufficient. On the other hand regional computation consumes much more computation. CNN-UM as a parallel-computing device is ideal to overcome these obstacles. Thus a CNN-UM based intelligent sensor is the ideal solution for adaptive sensing.

Due to changing lighting condition the contrast and intensity of an image may changing in space. This algorithm computes via a CNN algorithm the local contrast of the image and using diffusion, the regional quality of the image, which is based on regional intensity and contrast.

$$Q(x,y) := c_1 D((I(x,y) - I_a)^2) + c_2 D(C^2(x,y))$$

where I is the input image (intensity) I_a is the mean intensity, C is contrast and D is a diffusion operator.

Using this information, the parameter of the capturing device is determined and programmed locally. The way of adjusting depends on the kind of parameter. One way is to adjust the parameter so that Q is maximized, the other way is that the parameter is adjusted continuously such that a preset limit of Q is reached. This method results in an image, where the contrast and intensity is equalized, enhanced and over-saturation is avoided.

The proposed method is planned for the adaptive CNN-UM framework [15], which will be soon available on silicon. However it is also implementable on the current architecture.

3. **Adaptive image enhancement with CNN** ([2],[3],[4],[6] and chapter 7.)

I developed two adaptive image enhancement methods. Both methods are based on regional intensity and contrast. Both methods address intensifying and equalizing these qualities which avoids over-saturation.

3.1. Static method

I developed a static enhancement method consisting of diffusion and contras templates, and some arithmetic operations. The method intensifies intensity and contrast through using a filter, which inhibits over-saturation. The method can be realized on the current CNN-UM chip, but it exploits the capabilities of the planned adaptive chip.

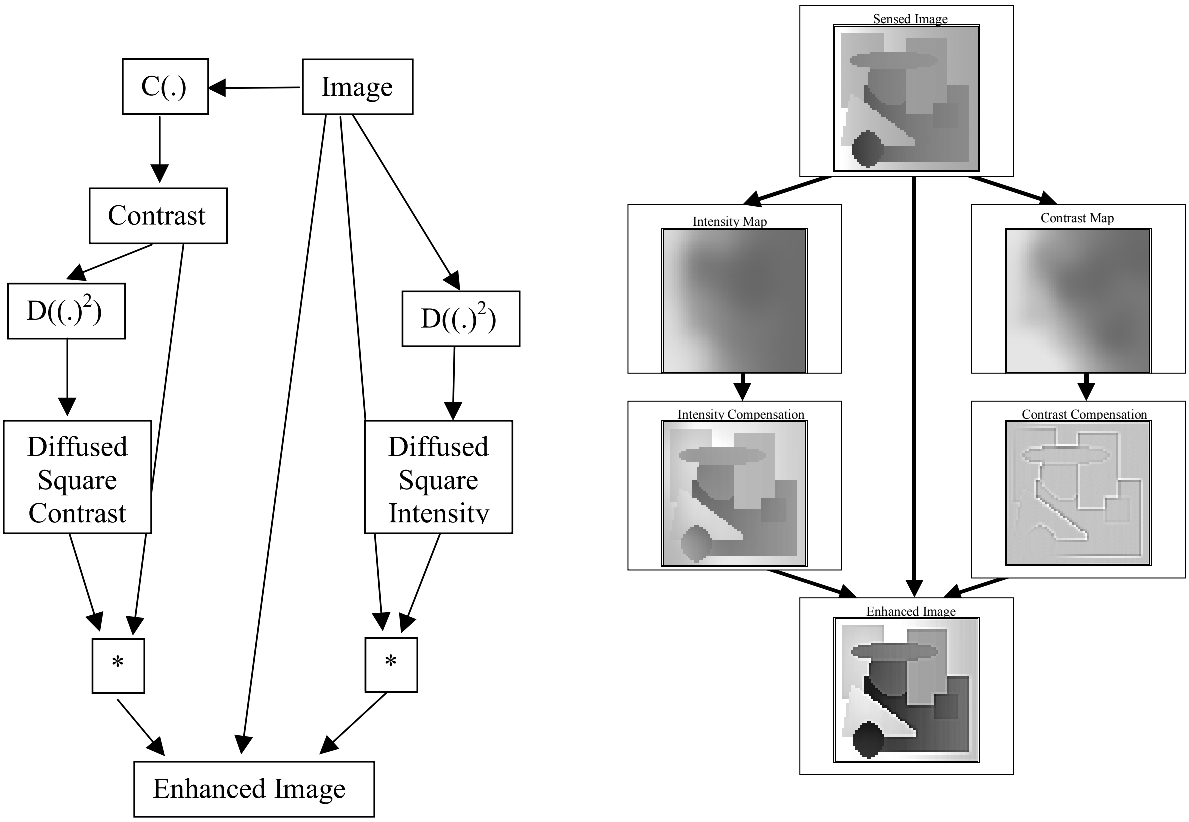

Figure 5 Algorithm and illustration of adaptive image enhancement

Figure 5 shows the static method. The part contrast-enhancement consists of computing the contrast, then diffusion is applied to compute a mask, and last the contrast is enhanced, using this mask, to inhibit over-saturation. Similarly, in the intensity-enhancement part intensity is diffused to compute an intensity-mask and intensity is enhanced, using this mask as a filter to avoid over-saturation.

The single-step enhancement method introduced can easily be realized by using the Adaptive Extended Cell in CNN-UM ([17]). It also can be implemented on the current chip, but several arithmetical operations are needed to replace the space-variant template.

3.2. Dynamic method

I also developed a dynamic method, which can be implemented on a multi-layer/complex-cell architecture. The output is resulting as the equilibrium of four interacting components, representing the intensity the contrast and the regional intensity and contrast The positive and inhibiting impacts result in an image, which is equalized and enhanced in intensity and contrast.

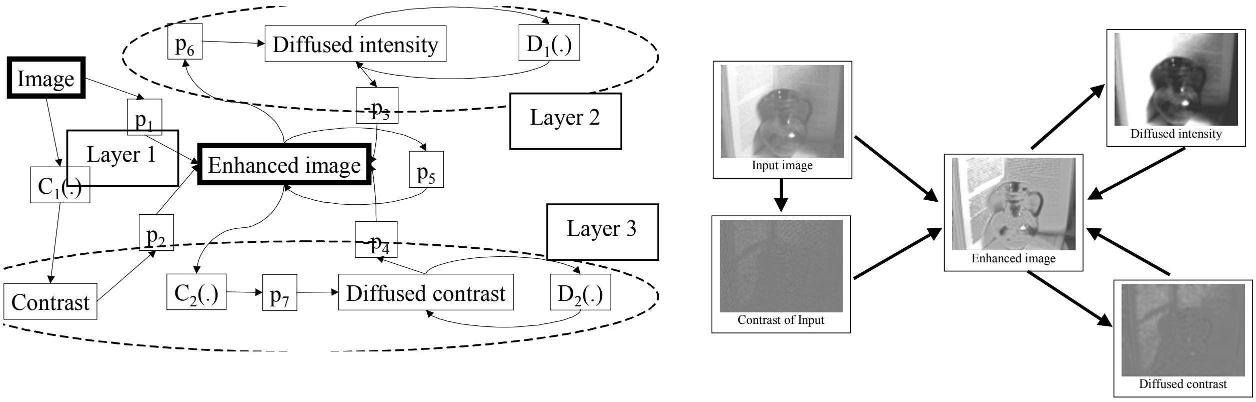

Figure 6 Flowchart and illustration of the multi-layer system of the dynamic method of adaptive image enhancement

Figure 6 shows the dynamic enhancement method. The output is a result of the equilibrium of a multilayer-system. The output is affected by several factors. First, factor is the input (intensity), which affects positively with the weight p_1 as an intensity enhancement. Second, the contrast is computed and it has with the weight p_2 also a positive effect resulting in contrast enhancement. Third, the actual intensity is diffused and fed back with a negative weight $-p_3$ to inhibit intensity over-saturation. Last, the actual contrast is diffused and fed back with a negative weight $-p_4$ resulting in contrast inhibition.

The realization of the dynamic algorithm needs a real-time space variant template. In the near future only the intensity enhancement and equalization part can be implemented, however this is a powerful method by itself. This part of the algorithm is realizable by the 2nd order complex cell CNN-UM described in [16].

4. Application of the results

The result of thesis 1 (chapter 4) is implicitly important in general training of the CNN. Fast computing of the gradient makes it possible that a compact system can be developed, which involves learning and adaptivity real time. Learning and adaptivity has only been possible by using external computing devices (for example digital computers) until now, which is off-line and extremely slower. Real-time adaptation and learning may be possible only, if the needed algorithms are integrated in the hardware. This becomes achievable as the gradients can be computed by the CNN itself.

Real-time learning is important in such cases, where new templates shall be quickly developed from samples or adjusted. On the other side, adaptivity may also be solved by learning: in this case, the template is adjusted quickly according to special conditions. For example, consider an image-processing task, where some secondary but influencing feature of the input image changes. The conventional solution is to create a robust algorithm. Using real-time learning the same problem can be solved applying a simpler algorithm, if the templates are adjusted occasionally.

Typical applications can be for example surveillance and security systems, where the captured image is processed with CNN algorithms. The templates of the algorithm can be adjusted to the special conditions of the site of operation with on-line learning.

Thesis 2 (chapter 5) introduces an adaptive sensing method. Application of this result is naturally to incorporate CNN technology in capturing devices or to create a CNN-UM based intelligent sensor. This sensor can be built in commercial digital video and photo cameras but adaptivity is also important in researching under extreme lighting conditions, like in space-research, under-water research and so on.

Computing speed that the CNN-UM offers makes real-time operation possible by a cheap device. This solution at the same time also involves other possibilities, for example subsequent image processing tasks.

Thesis 3 (chapter 6) is a good example for this, since an image enhancing algorithm can be integrated after sensing, to enhance the image even more for visibility. Another possibility is to use it for already captured images. It is possible that to apply this method, such that the CNN-UM is used as a graphical acceleration card for example in personal computer. Image enhancement for visualization is most important in fields where human decision is based on examining images, for example medical imaging.

5. Acknowledgements

In the professional aspect the first and main acknowledgement shall be entitled to Professor Tamás Roska, my scientific adviser and leader of the Analogic Laboratory, where I made all of the researches. I also have to thank for Him, because He awaked the interest in me to this and several other topics of computer sciences already as one of my under-graduate teacher. Also for the possibility of making undisturbed and absorbed researches, and for the good technical background in the laboratory.

I am also grateful to all of my laboratory co-workers for helping in research and in various kinds of study and for the pleasant feeling in the laboratory. Especially I thank among them Levente Török, László Orzó and Viktor Gál for revision and not only being co-worker but also good friends.

I am beholden for all of my teachers of the study part of my Ph.D., especially to András Lőrincz from whom I learned much about neural networks and related

artificial intelligent methods. I had also the opportunity to work with him on applying a special method for teaching the CNN during a summer.

A Ph.D. does not start at the beginning of the post-graduate studies, but there are also backgrounds in the undergraduate studies. So I have to acknowledge for all of my University Teachers at the University of Veszprém, especially for Ferenc Friedler, with whom I started and learned scientific research.

I also whish to thank for all of my secondary-school teachers, where all the important knowledge were based, and the interest to sciences were awaken. Especially I thank for my math teacher Ilona Jakab for that she taught me mathematics excellently and also for the mathematical and theoretical reason, which I am engaged to since then.

The route to a Ph.D. study and dissertation is a way, on which a man walks in his life among many other happenings and aspects. I thank to all of my friends and girlfriends who helped me on this way and generally in the troubles and pleasures of life. The start of this way springs from the family, from where one brings already very important intentions: interests, goals and mentality. Therefore I thank for my parents and my sister for being on my side.

References to publication of the author

[1] **M. Brendel**, Gusztáv Bártfai, Tamás Roska, "Reciprocal CNN gradient computing for back-propagation-through-time learning of cellular neural networks", Research report of the Analogic (Dual) and Neural Computing Systems Laboratory, (DNS-5-1999), Budapest, MTA SZTAKI, 1999.

[2] **M. Brendel**, "Adaptive image sensing and enhancement using the Cellular Neural Network Universal Machine", Research report of the Analogic (Dual) and Neural Computing Systems Laboratory, (DNS-9-2000), Budapest, MTA SZTAKI, 2000.

[3] **M. Brendel**, T. Roska, "Adaptive Image Sensing and enhancement using Adaptive Cellular Neural Network Universal Machine", Proceedings of IEEE Int. Workshop on Cellular Neural Networks and Their Applications, (CNNA'2000), pp. 93-98, Catania, 0-7803-6344-2, 2000.

[4] **M. Brendel**, T. Roska, "Adaptive image sensing and enhancement using the cellular neural network universal machine", International Journal of Circuit Theory and Applications, vol. 30, pp. 287-312, 2002.

[5] **M. Brendel**, T. Roska and Gusztáv Bártfai, " Gradient computation of continuous-time Cellular Neural/Nonlinear Networks with linear templates via the CNN Universal Machine", Neural Processing Letters, vol. 16, pp. 111-120, 2002.

[6] **M. Brendel**, " Adaptív képérzékelés CNN-UM segítségével ", Research report of the Analogic (Dual) and Neural Computing Systems Laboratory, (DNS--2002), Budapest, MTA SZTAKI, 2002.

[7] T. Roska, L. Kék, L. Nemes, Á. Zarándy, **M. Brendel** and P. Szolgay (eds.), "CNN Software Library (Templates and Algorithms) Version 7.2", Research Report (DNS-1-1998), Analogical and Neural Computing Laboratory, MTA SZTAKI, Budapest 1998.

[8] J. Hámori, and T. Roska (ed.), D. Bálya, Zs. Borostyánkői, **M. Brendel**, V. Gál, J. Hámori, K. Lotz, L. Négyessy, L. Orzó, I. Petrás, Cs. Rekeczky, T. Roska, J. Takács, P. Venetiáner, Z. Vidnyánszky, Á. Zarándy, „Receptive Field Atlas the Retinotopic Visual Pathway and some other Sensory Organs using Dynamic Cellular Neural Network Models”, Research report of the Analogic (Dual) and Neural Computing Systems Laboratory, (DNS-8-2000), Budapest, MTA SZTAKI, 2000.

References to publications used for the introduction of the theses

[9] L. O. Chua and L. Yang, "Cellular Neural Networks: Theory and Applications", IEEE Transactions on Circuits and Systems, Vol. 35, pp. 1257-1290, 1988.

[10] 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, 1993.

[11] W. K. Pratt, "Digital Image Processing", John Wiley & Sons Inc., 1991.

[12] S. M. Smirnakis, M. J. Berry, D. K. Warland, W. Bialek and M. Meister, "Adaptation of retinal processing to image contrast and spatial scale" Nature*, vol. 386, pp. 69-73, 1997.*

[13] J. A. Nossek "Design and learning with cellular neural networks" Proceedings of (CNNA-94), pp. 137-146, 1994.

[14] F. Beaufays and E. Wan, "A Unified Approach to Derive Gradient Algorithms for Arbitrary Neural Network Structures", Proc. ICANN'94, vol. 1, pp. 545-548, Sorrento, Italy, 26-29 May 1994.

[15] T. Roska, "Computer-Sensors: spatial-temporal computers for analog array signals, dynamically integrated with sensors", Journal of VLSI Signal Processing Special Issue: Spatiotemporal Signal Processing with Analogic CNN Visual Microprocessors, (JVSP Special Issue), pp. 221-238, vol. 23, No.2/3, November/December 1999, Kluwer

[16] Cs. Rekeczky, T. Serrano-Gotarredona, T. Roska, and A. Rodriguez-Vazquez "A stored program 2nd order/ 3-layer complex cell CNN-UM", Proceedings of the 6th IEEE International Workshop on Cellular neural Networks and their Applications (CNNA-2000), Catania, pp. 213-217, 2000.

[17] T. Roska, "Analogic CNN computing: Architectural, Implementational and Algorithmic Advances – a Review", Proceedings of the 4th IEEE International Workshop on Cellular neural Networks and their Applications (CNNA-1996), 14-17 April, London, pp. 3-11, 1998.