

CMOL CrossNets as Pattern Classifiers

Jung Hoon Lee and Konstantin K. Likharev

Stony Brook University, Stony Brook, NY 11794-3800, U.S.A
{jlee@grad.physics, klikharev@notes.cc}sunysb.edu

Abstract. This presentation has two goals: (i) to review the recently suggested concept of bio-inspired CrossNet architectures for future hybrid CMOL VLSI circuits and (ii) to present new results concerning the prospects and problems of using these neuromorphic networks as classifiers of very large patterns, in particular of high-resolution optical images. We show that the unparalleled density and speed of CMOL circuits may enable to perform such important and challenging tasks as, for example, online recognition of a face in a high-resolution image of a large crowd.

1 CrossNets

There is a growing consensus that the forthcoming problems of the Moore Law [1] may be only resolved by the transfer from a purely semiconductor-transistor (CMOS) technology to hybrid (“CMOL”) integrated circuits [2, 3]. Such circuit would complement a CMOS chip with a nanowire crossbar (Fig. 1) with nanodevices (e.g., specially designed functional molecules) formed between the nanowires at each crosspoint.

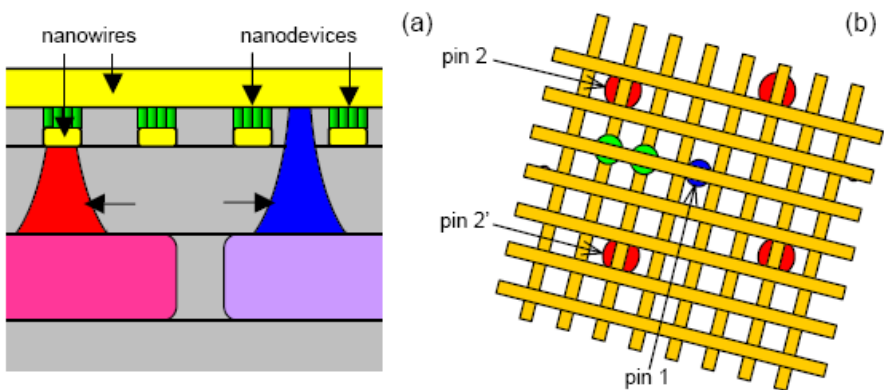


Fig. 1. CMOL circuit (schematically): (a) side view and (b) top view showing several adjacent pins. The latter view shows that the specific angle between the interface pin lattice and nanodevice crossbar allows each nanodevice to be addressed via the appropriate pin pair (e.g., pins 1 and 2 for the left of the two shown devices, and pins 1 and 2' for the right device)

The basic idea behind such hybrid circuits is that its minimum features are not defined by lithography (which below ~ 10 nm will become prohibitively expensive) but by a Nature-given standard such as the size of a certain molecule. Estimates show [3] that CMOL circuits may feature unprecedented density (up to $\sim 10^{12}$ active devices per cm^2), at acceptable fabrication costs.

Realistically, nanodevices will hardly ever be formed (e.g., chemically self-assembled) with 100% yield. This is why CMOL circuit architectures should ensure their high defect tolerance. Recently we have shown that such high tolerance may be achieved in cell-based FPGA-type reconfigurable logic circuits [3, 4] and (to a less extent) in hybrid memories [3, 5]. However, the most natural application of the CMOL technology is in bio-inspired neuromorphic networks which are generically defect-tolerant.

We have proposed [6-8] a family of such architectures, called Distributed Crossbar Networks (“CrossNets”), whose topology uniquely maps on CMOL circuits. Figure 2a shows the generic architecture of our CrossNets. Relatively sparse neural cell bodies (“somas”) are implemented in the CMOS subsystem. In the simplest firing rate model, each soma is just a differential amplifier with a nonlinear saturation (“activation”) function $V_{\text{out}} = f(V_{\text{in}})$. Axons and dendrites are implemented as physically similar, straight segments of nanowires, while nanodevices (latching switches), formed at the nanowire crosspoints, play the role of elementary synapses. Axonic voltage V_a , developed by the somatic amplifier, is transferred by each axonic nanowire to synapses, so that if the synaptic latch of a particular synapse is in the ON state, a current proportional to V_a is flowing into the corresponding dendritic nanowire, contributing to the input signal of the post-synaptic cell.

In the generic CrossNets (Fig. 2a), any pair of cells is interconnected by two synapses leading to the opposite inputs of the somatic amplifier, so that the net synaptic weight w_{jk} may take any of three values which may be normalized to -1, 0, and +1. In other CrossNets versions the number of synapses is larger. In particular, in

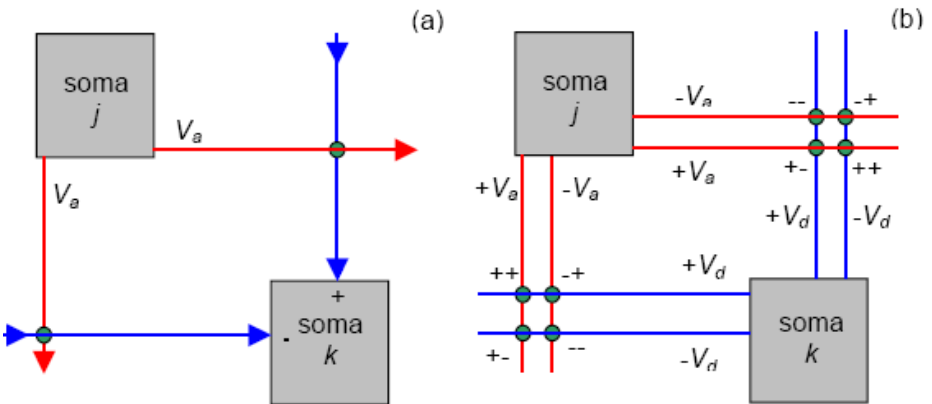


Fig. 2. Cell coupling is the (a) generic feedforward and (b) Hebbian feedforward CrossNets. Here and below, axonic nanowires are shown in red, synaptic nanowires in blue. Each gray square show the interface pin area of a somatic cell. (The cells as such may be much larger, since they are implemented in the underlying CMOS subsystem)

recurrent CrossNets, the number of nanowires and synapses per cell is doubled to carry feedback signals. (Generally, CrossNets are asymmetric: $w_{kj} \neq w_{jk}$.) In order to enable quasi-Hebbian learning rule, the number of nanowires and synapses per cell may be increased even further (Fig. 2b).

In the simplest cases (e.g., quasi-Hopfield networks with finite connectivity),¹ the tri-level synaptic weights of the generic CrossNets are quite satisfactory, leading to just a very modest network capacity loss [7]. However, some applications (in particular, pattern classification) may require a larger number of weight quantization levels L (e.g., $L \sim 30$ for a 1% fidelity [8]). This can be achieved by using compact square arrays (e.g., 4×4) of latching switches (Fig. 3).

In CrossNets the CMOS-implemented somatic cells are large and sparse. Because of this, the each axonic signal is passed to dendritic wires of many (M) other cells, and each dendritic signal is contributed by M pre-synaptic cells. The distribution of somatic cells may vary, creating several CrossNet species (Fig. 4).

CrossNet training faces several hardware-imposed challenges:

- (i) The synaptic weight contribution provided by the elementary latching switch is binary.
- (ii) The only way to adjust any particular synaptic weight is to turn ON or OFF the corresponding latching switch(es). This is only possible to do by applying certain voltage $V = V_a - V_d$ between the two corresponding nanowires. At this procedure, other nanodevices attached to the same wires should not be disturbed.

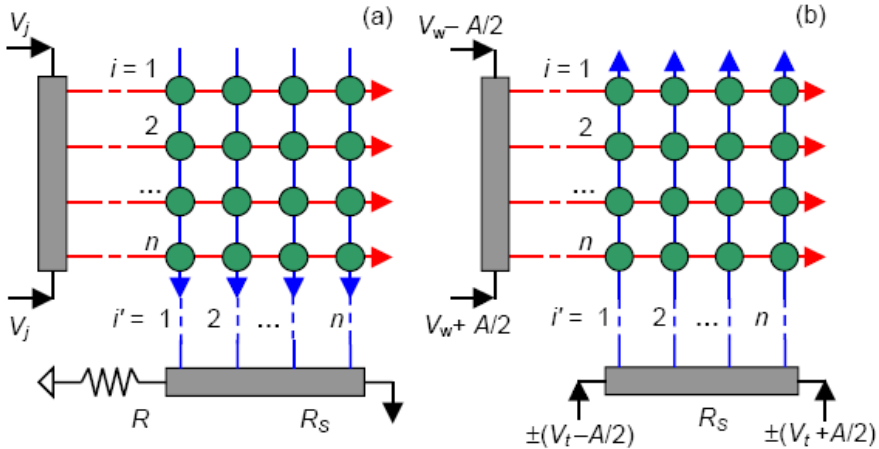


Fig. 3. A half of a composite synapse for providing $L = 2n^2 + 1$ discrete levels of the weight in (a) operation and (b) weight import modes. The dark-gray rectangles are resistive metallic strips at soma/nanowire interfaces

¹ So far, this operation mode is the only one for which the CrossNet defect tolerance has been analyzed in detail [8]. The results are very encouraging: for example, the network may have a 99% fidelity, with a 50% capacity loss, at a fraction of bad nanodevices above 80% (!).

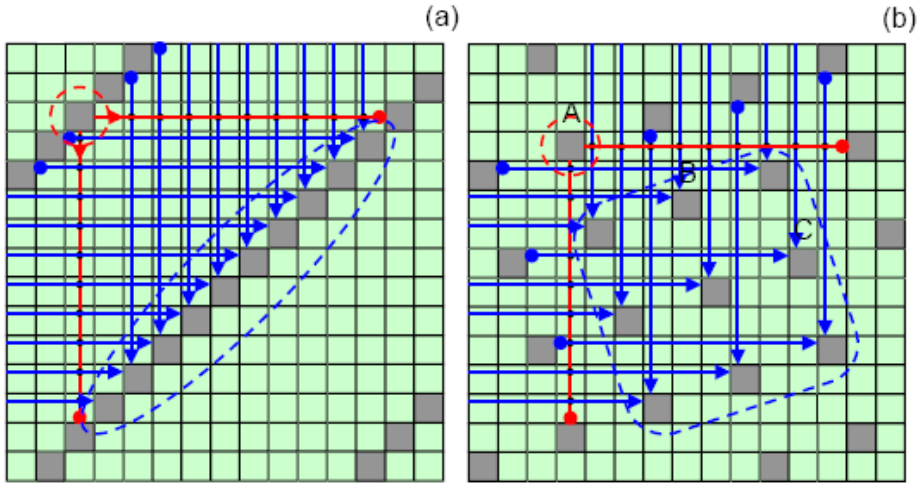


Fig. 4. Two main CrossBar species: (a) FlossBar and (b) InBar, in the generic (feedforward, non-Hebbian, ternary-weight) case for the connectivity parameter $M = 9$. Only the nanowires and nanodevices coupling one cell (indicated with red dashed lines) to M post-synaptic cells (blue dashed lines) are shown; actually all the cells (e.g., B and C on panel (b)) are similarly coupled. Bold points show open-circuit terminations of the axonic and dendritic nanowires, which prevent cell interaction in bypass of synapses

- (iii) Processes of turning single-electron latches ON and OFF are statistical rather than dynamical [2], so that the applied voltage V can only control probability rates of these, generally random events. (This problem is least significant, because the randomness may be confined by an appropriate design of the nanodevices [6].)

We have shown that these challenges may be met using (at least) the following training methods [8]:

(i) *Synaptic weight import.* This procedure is started with training of a homomorphic “precursor” artificial neural network with continuous synaptic weights w_{jk} , implemented in software, using one of established methods (e.g., error backpropagation). Then the synaptic weights w_{jk} are transferred to the CrossNet, with some “clipping” (rounding) due to the binary nature of elementary synaptic weights. To accomplish the transfer, pairs of somatic cells are sequentially selected via CMOS-level wiring. Using the flexibility of CMOS circuitry, these cells are reconfigured to apply external voltages $\pm V_W$ to the axonic and dendritic nanowires leading to a particular synapse, while all other nanowires are grounded. The voltage level V_W is selected so that it does not switch the synapses attached to only one of the selected nanowires, while voltage $2V_W$ applied to the synapse at the crosspoint of the selected wires is sufficient for its reliable switching. (In the composite synapses with quasi-continuous weights (Fig. 4), only a part of the corresponding switches is turned ON or OFF.)

(ii) *Error backpropagation.* The synaptic weight import procedure is straightforward when w_{jk} may be simply calculated, e.g., for the Hopfield networks.

However, for very large CrossNets used as classifiers the precursor network training may take an impracticably long time. In this case the direct training of a CrossNet may become necessary. We have developed two methods of such training, both based on “Hebbian” synapses (Fig. 2b). In CrossNets with such synapses, each axonic and dendritic signal is now passed, in the dual-rail format, through two nanowires to groups of four latching switches. Starting from the Arrhenius law describing the probability of switching of each single-electron latch, it is straightforward to show [8] that the average synaptic weight of the 4-latch group obeys the following equation:

$$\frac{d}{dt}\langle w \rangle = -4\Gamma_0 \sinh(\gamma S) \sinh(\gamma V_a) \sinh(\gamma V_d), \quad (1)$$

where Γ_0 and γ are parameters of the latching switch, while S is a global, externally-controllable shift voltage which may be applied to all switches via a special gate. The quasi-Hebbian rule (1) may be used to implement the backpropagation algorithm either using a periodic time-multiplexing [8] or in a continuous fashion, using the simultaneous propagation of signals and errors along the same dual-rail channels

2 Pattern Classification

It could seem that both feedforward CrossNet species shown in Fig. 4 may be used as multilayered perceptrons (MLPs) for pattern classification [9]. However, in both cases there are some problems. FlossBars (Fig. 4a) are indeed the MLPs, but with limited connectivity between the layers, while InBars (Fig. 4b) do not have a layered structure at all. (For example, cell C gets input not only from the “first-layer” cell A, but also from the “second-layer” cell B.)

Figure 5 shows typical results of our study of an impact of these features on the performance of these networks as classifiers. The results show that, unfortunately, InBars cannot be trained to work well as classifiers, at least by error backpropagation. (At this stage, we still cannot offer a simple explanation for this fact.) On the other hand, finite connectivity of FlossBars does not affect their fidelity too substantially. Moreover, some connectivity restriction turns out to be useful to achieve the best training results. Unfortunately, Fig. 4a shows that FlossBars are not very convenient for CMOS implementation: interconnect pin areas are too close (at the distance of the order of nanowire pitch), so that CMOS layout of somatic cells may be difficult. (This would not be a problem for InBar – Fig. 4b.)

Figure 6 shows a possible way to overcome these difficulties: an MLP with a global synaptic crossbars providing full connectivity between layers implemented in InBar style. The only evident negative feature of this network is a certain loss of chip real estate, since nanodevices over the CMOS somatic cell areas are not used. (In contrast, CMOS circuits under the synaptic crossbars may be utilized – see below.) This loss, however, may be very limited, because the somatic area width W (in terms of the cell number) is only determined by the pitch ratio of CMOS and nanowiring, and may be of the order of 10, i.e. substantially smaller than the data vector length $D \sim 10^3$ for the most challenging classifier applications. In this case the relative area loss is of the order of $(LW)/L^2 = W^2/D \sim 10\%$, i.e. negligible.

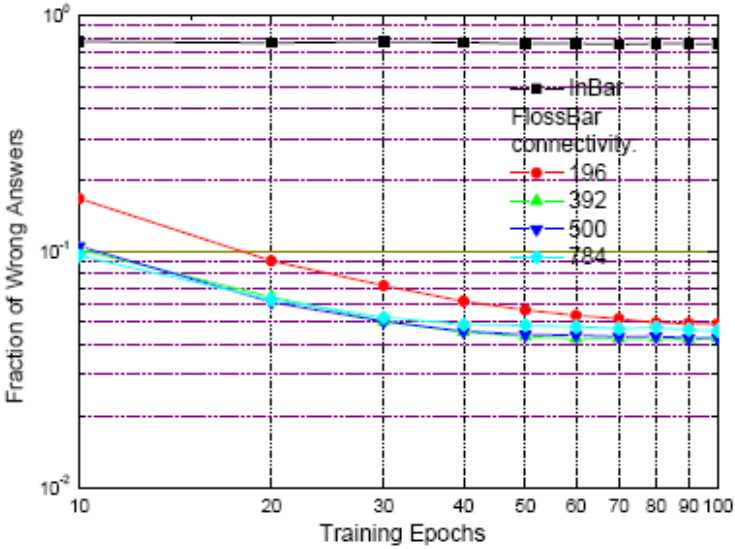


Fig. 5. Test set fidelity of an InBar (with $N = 784+784+10$ cells and connectivity $M = 784$) and FlossBar MLPs (also with $784+784+10$ cells with limited connectivity between the input and first hidden layer) after their training as classifiers of the MNIST data set of handwritten characters [10]. In both cases, the training set size was 60,000 images, test set 10,000 patterns, with $28 \times 28 = 784$ pixels (with 256 shades of gray) each

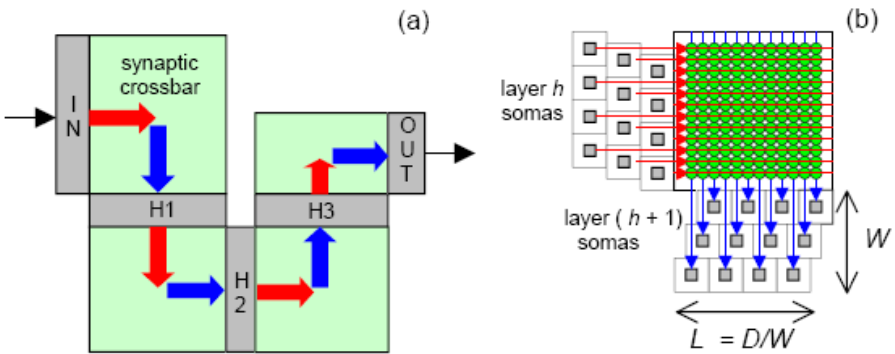


Fig. 6. Multi-layered perceptron based on InBar somatic areas and global synaptic crossbars: (a) general structure and (b) CMOL implementation of two adjacent layers (schematically). In this case, each green circle denotes a composite synapse consisting of two arrays shown in Fig. 3, with quasi-continuous synaptic weight

Let us give an approximate estimate of performance of such CMOL classifiers for such a challenging and important task as a search of a particular person’s face on a high-resolution picture of a large crowd. The most difficult feature of such recognition is the search for face location, i.e. optimal placement of a face on the

image relative to the panel providing input for the processing network. The enormous density and speed of CMOL hardware gives a possibility to time-and-space multiplex this task (Fig. 7). In this approach, the full image (say, formed by CMOS photodetectors on the same chip) is divided into P rectangular panels of $h \times w$ pixels, corresponding to the expected size and approximate shape of a single face. A CMOS-implemented communication channel passes input data from each panel to the corresponding CMOL neural network, providing its shift in time, say using the TV scanning pattern (red line in Fig. 7). The standard methods of image classification require the network to have just a few hidden layers, so that the time interval Δt necessary for each mapping position may be so short that the total pattern recognition time $T = hw\Delta t$ may be acceptable even for online face recognition.

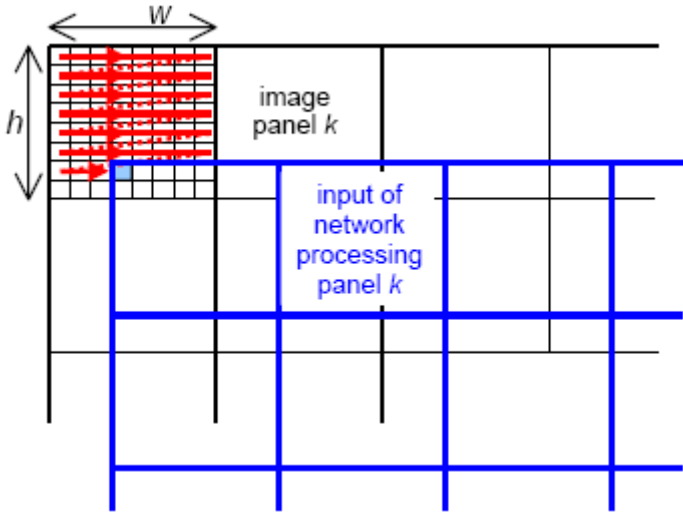


Fig. 7. Scan mapping of the input image on CMOL neural network inputs. Red lines show the possible time sequence of image pixels sent to a certain input of the network processing image from the upper-left panel of the pattern

Indeed, let us consider a 4-Megapixel image partitioned into 4K 32×32 -pixel panels ($h = w = 32$). This panel will require a neural network with several (say, four) layers with 1K cells each in order to compare the panel image with $\sim 10^3$ stored faces. With the feasible 45-nm CMOS technology [1], 4-nm nanowire half-pitch [3, 4], and 65-level synapses (sufficient for better than 99% fidelity [8]), each interlayer crossbar would require chip area about $(4K \times 64 \text{ nm})^2 = 64 \times 64 \mu\text{m}^2$, fitting $4 \times 4K$ of them on a $\sim 0.6 \text{ cm}^2$ chip.² With the typical nanowire capacitance of $2 \times 10^{-10} \text{ F/m}$ and latching

² The CMOS somatic-layer and communication-system overheads are negligible. Also small is the chip area required for the output signal pins, because the search result may be demultiplexed into just a few bits (say, the recognized face’s ID number and its position on the image).

switch ON resistance $\sim 10^{10} \Omega$ (providing acceptable power consumption of the order of 10 W/cm^2 [8]), the input-to-output signal propagation in such a network will take only $\sim 4 \times (8 \times 10^{-9} \text{ m}) \times (2 \times 10^{-10} \text{ F/m}) \times (10^{10} \Omega) \approx 50 \text{ ns}$, so that Δt may be of the order of 100 ns and the total time $T = hw\Delta t$ of processing one frame of the order of 100 microseconds, much shorter than the typical TV frame time of ~ 10 milliseconds. The remaining two-order-of magnitude gap may be used, for example, for double-checking the results via stopping the scan mapping (Fig. 7) at the most promising position. (For this, a simple feedback from the recognition output to the mapping communication system is necessary.)³

It is instructive to compare the estimated CMOL chip speed with that of the implementation of a similar parallel network ensemble on a CMOS signal processor (say, also combined on the same chip with an array of CMOS photodetectors). Even assuming an extremely high performance of 30 billion additions/multiplications per second, we would need $\sim 4 \times 4\text{K} \times 1\text{K} \times (4\text{K})^2 / (30 \times 10^9) \approx 10^4$ seconds ~ 3 hours per frame, evidently incompatible with the online image stream processing.

3 Conclusions

Our preliminary estimates show that even such a brute-force approach as a parallel use of a few thousand similar neural networks on a single silicon chip, with time-multiplexing analysis of each of $\sim 10^3$ possible mappings of a high-resolution image on the network input, may enable CMOL CrossNets to perform tasks clearly impossible for the usual implementation of neural networks on serial digital computers. Our nearest goals are to verify and quantify these predictions via explicit numerical simulation of such systems, and to work on the development of more elaborate schemes for CMOL CrossNet applications as classifiers of complex patterns.

Useful discussions with X. Ma and Ö. Türel are gratefully acknowledged. The work has been supported by AFOSR, NSF, and MARCO FENA Center.

References

1. International Technology Roadmap for Semiconductors. 2003 Edition, 2004 Update, available online at <http://public.itrs.net/>
2. Likharev, K.K.: Electronics Below 10 nm. In: Greer, J. *et al.* (eds.) Nano and Giga Challenges in Microelectronics. Elsevier, Amsterdam (2003) 27-68
3. Likharev, K.K., Strukov, D.B.: CMOL: Devices, Circuits, and Architectures. In: Cuniberti, G. *et al.* (eds.) Introduction to Molecular Electronics. Springer, Berlin (2005)
4. Strukov D.B., Likharev, K.K.: A Reconfigurable Architecture for Hybrid CMOS/Nanoscale Circuits" (with D. B. Strukov). Report at FCCM'05, to be published (2005)

³ Reportedly, hierarchical networks, e.g., the so-called LeNets which have been successfully used for handwriting recognition [10], may be more defect tolerant. Our estimates have shown that a similar set of parallel networks of this type would require just a little bit larger chips – about 1 cm^2 for the parameters cited above.

5. Strukov D.B., Likharev, K.K.: Prospects for Terabit-scale Nanoelectronic Memories. *Nanotechnology* 16 (2005) 137-148
6. Fölling, S., Türel, Ö, Likharev, K.K.: Single-Electron Latching Switches as Nanoscale Synapses. In: *Proceedings of the IJCNN (2001)* 216-221
7. Türel, Ö., Likharev, K.K.: CrossNets: Possible Neuromorphic Networks Based on Nanoscale Components. *Int. J. of Circ. Theor. Appl.* 31 (2003) 37-53
8. Türel, Ö., Lee, J.H., Ma, X., Likharev, K.K.: Neuromorphic Architectures for Nanoelectronic Circuits. *Int. J. of Circ. Theor. Appl.* 32 (2004) 277-302
9. See, e.g., Hertz, J., Krogh, A. and Palmer, R.G.: *Introduction to the Theory of Neural Computation*. Perseus, Cambridge, MA (1991)
10. LeCun, Y. *et al.*: Gradient-based Learning Applied to Document Recognition. *Proc. IEEE* 86 (1998) 2278-2324