On Scalable Spiking ConvNet Hardware for Cortex-Like Visual Sensory Processing Systems

L. Camuñas-Mesa, J. A. Pérez-Carrasco, C. Zamarreño-Ramos, T. Serrano-Gotarredona, and B. Linares-Barranco

Instituto de Microelectrónica de Sevilla (IMSE-CNM-CSIC), Américo Vespucio s/n, 41092 Sevilla, Spain.

E-mail: *bernabe@imse-cnm.csic.es*

***Abstract*– This paper summarizes how Convolutional Neural Networks (ConvNets) can be implemented in hardware using Spiking neural network Address-Event-Representation (AER) technology, for sophisticated pattern and object recognition tasks operating at mili second delay throughputs. Although such hardware would require hundreds of individual convolutional modules and thus is presently not yet available, we discuss methods and technologies for implementing it in the near future. On the other hand, we provide precise behavioral simulations of large scale spiking AER convolutional hardware and evaluate its performance, by using peformance figures of already available AER convolution chips fed with real sensory data obtained from physically available AER motion retina chips. We provide simulation results of systems trained for people recognition, showing recognition delays of a few miliseconds from stimulus onset. ConvNets show good up scaling behavior and possibilities for being implemented efficiently with new nano scale hybrid CMOS/nonCMOS technologies.**

I. Introduction

In 1959 Hubel and Wiesel reported their findings on projection field processing in early stages of visual cortex, receiving the 1981 Nobel prize. Based on this finding, Convolutional Neural Networks (ConvNets) were originally proposed by Fukushima in 1969 [1]-[2] and further developed by Yann LeCun [3] and other groups, as a type of continuous-time gradient-based learning neural paradigm, with great success in a variety of (industrial) applications as well as research. Examples of industrial applications and developments are, to mention a few: (1) NEC with products for face/person detection, age and gender recognition for vending machines, as well as prototypes for cancer cell detection or mobile phone imaging applications, (2) France Telecom/Orange with face detection and recognition, text detection and recognition, various mobile phone applications, (3) Vidient Technologies with products for video surveillance, human detection and tracking, (4) Canon with cameras with embedded video surveillance, (5) Microsoft with handwriting recognition, (6) AT&T/Lucent-Technologies/ NCR with products for check recognition. Examples of state-of-the-art research exploiting ConvNets are (1) Poggio at MIT with object recognition and scene analysis [5], (2) Seung at MIT with image segmentation, and biological image analysis (brain circuit reconstruction) [6], (3) NEC Labs with natural language processing and understanding [7], (4) NYU with biological image analysis, object recognition and visual navigation for robots [8].

On the other hand, in 1996, Thorpe demonstrated that the human visual system is capable of performing object recognition tasks at such speeds that any neuron involved only had time to fire one spike [11]. Based on this finding, he developed a Framework for spiking ConvNets, which is presently being exploited commercially for high speed object recognition software [12].

In the field of VLSI circuit design we have witnessed, during the past years, important developments in the field of spiking neural hardware, and specifically spiking hardware for ConvNet processing [13]-[15] of visual information sensed by highly efficient spiking sensors [16]. It is now becoming apparent that the combination of ConvNets theories and knowledge, the framework of spiking information sensing and processing, with state-of-the-art hardware technologies such as Networks-On-Chip [17] and emergent nano scale CMOS and hybrid CMOS/non-CMOS nanotechnologies [18], will result in highly efficient systems for sophisticated cognitive tasks, similar to the human brain. In this paper we review scaling properties for ConvNets hardware, discuss spike signal coding, explain spiking convolution chips and how to use them for assembling large scale modular cortex-like structures for object recognition. We present some behavioral simulation results of such structures for human detection and tracking. At the end we discuss the potential of spiking ConvNet systems for being implemented with new coming nanotechnology devices.

II. Scaling Properties of ConvNets Hardware

Fig. 1 shows a typical ConvNet architecture. It usually contains a reduced number of sequential layers (4-10), each of which performs several 2D filtering operations in parallel. Early stages extract simple features (such as edge orientation and scale), which are progressively combined into more complex shapes and figures at later stages. Early stages usually operate with small but dense kernels, while later stages use longer range but sparser ones [5]. To increase the knowledge (dictionary of shapes and figures) of the system one simply adds more 2D filters in later layers. Example ConvNet systems for face and character recognition applications may have several tens to hundreds filters per layer. What is interesting about ConvNets, compared to other neural networks, is their graceful scaling capability. To increase knowledge one simply has to increase the number of convolutional modules in a layer. Thus, number of neurons (pixels) scales linearly with the number of modules. There is a fixed number of synapses per filter (the convolutional kernel weights) and a fixed number of filters per convolutional module. Consequently, number of synapses also scales linearly with the number of modules. On the other hand, the latency of the computing structure (if implemented as parallel hardware) is determined mainly by the number of sequential layers, which is a reduced number and does not change for a given application. Therefore, speed does not degrade by adding more modules per layer (more knowledge), at least in first order. In other neural network architectures the number of synapses scales quadratically with the number of neurons. Consequently, ConvNets seem very appealing for configurable, modular and scalable spiking hardware implementations.

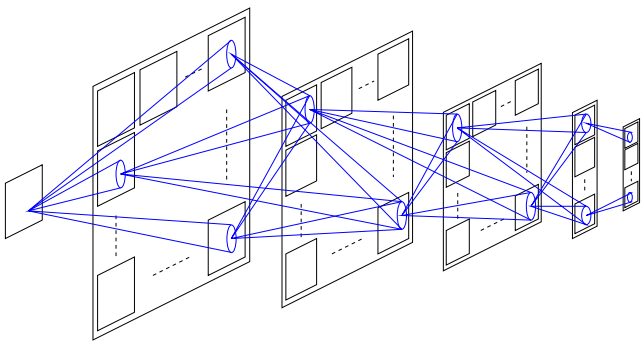


Fig. 1: Typical structure of a Feed Forward ConvNet

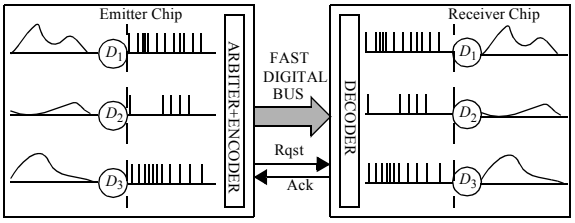


Fig. 2: Rate-coded point-to-point AER inter-chip communication link

III. Spike Signal Coding

Biological brains code and transmit information as spiking signals. This is because spike coding can be highly energy efficient as well as information processing efficient. In biology, a typical 2*ms* "action potential" neural spike travelling about 1*m* from the brain to a finger muscle needs about 20-40*ms*. In space, such spike only charges about 5*cm* of nerve at a time, but not the whole line. Consequently, the spike sender only needs to provide charge for the travelling 5*cm* line segment. This contrasts with present day electronic systems where digital information (bits) are transmitted by fully charging and later discharging wires. Furthermore, when using transmission lines for high speed, there is usually a 50Ω resistive component, besides the capacitive load, which is permanently dissipating energy (even in the absence of information transmission). Spike encoding can use highly energy efficient means such as soliton technology [9].

Regarding information encoding, spikes allow for very efficient schemes. Thorpe demonstrated in 1996 [11] that, in the human brain, fast recognition of sophisticated figures (such as animal detection in a photograph) is performed in a feed forward manner in such a way that the neurons involved only generate one spike. Thorpe later developed spike-processing convolutional architectures and rank-order coding schemes capable of performing this type of recognition efficiently in software [12]. This shows that visual information can be encoded as waves of spikes crossing the cortical structures using a relatively small number of spikes.

IV. AER Convolution Chips

Address-Event-Representation (AER) is a promising emergent hardware technology that shows potential for providing the computing requirements of large projection-field based multi-layer systems. AER was first proposed in 1991 in one of the Caltech research labs [19]-[20], and has been used since then by a wide community of neuromorphic hardware engineers. AER has been used fundamentally in image sensors, for simple light intensity to frequency transformations [21], time-to-first-spike coding [22]-[23], foveated sensors [24], spatial contrast [25]-[26] and more elaborate transient detectors [27]. But AER has also been used for auditory systems [28], competition and winner-takes-all networks [29]-[30], and even for systems distributed over wireless networks [31]. Some AER convolution processing chips with hardwired kernels (slightly tunable) have also been proposed [32]-[33]. However, it was not until arbitrary-shape-kernel convolution chips became available (with [34] or without kernel symmetry restrictions [13]) that their potential for building large scale AER ConvNets for arbitrary pattern and object recognition applications became apparent [14]-[15]. Several AER fully-programmable-kernel convolution chips have been reported. Either mixed-mode based on pixel-level charge packet integration [13]-[14], or fully digital with in-pixel accumulator and adder to emulate leaky integrate-and-fire neurons [35].

Fig. 2 illustrates event communication in a point-to-point AER link [36], where pixel intensity is coded directly as pixel event frequency[1]. The continuous-time states of pixels D_i in an emitter chip are transformed into sequences of fast digital pulses

1. Other more efficient coding schemes have been proposed, such as rank-order coding [37] where the order of the events carries the information, instead of pixel frequency.

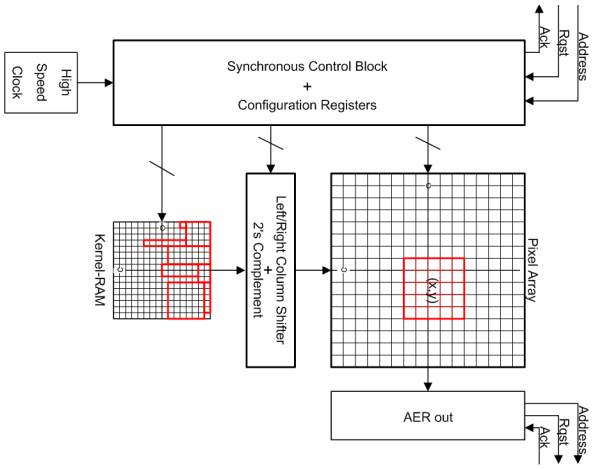


Fig. 3: AER multi-kernel Convolution Chip Diagram

(spikes or events) of minimal width (in the order of *ns*) but with much longer inter-spike intervals (typically in the order of *ms*). Each time a pixel generates a spike, its *x,y* address is written on the inter-chip digital bus, after proper arbitration. This is called an "Address Event". The receiver chip reads and decodes the addresses of the incoming events and sends spikes to the corresponding receiving pixels for reconstruction or further processing. In an AER convolution receiver chip, incoming events are sent to a neighborhood of pixel *x,y* onto which the 2D kernel is added. Fig. 3 shows the conceptual diagram of a fully digital AER Convolution chip. It contains a pixel array, where each pixel includes an adder/accumulator where incoming events are accumulated. When the accumulator reaches a positive (negative) threshold, the pixel is reset and generates a positive (negative) output event. The convolution kernels are stored in the kernel RAM. The controller copies the kernel line by line from the kernel RAM to the pixel array. Kernels are shifted left/right depending on the incoming event coordinate. In parallel, the controller subtracts a fixed number from each pixel accumulator at a fixed rate, to emulate a leak. This way, leaky integrate-and-fire neurons [13] are implemented with fully digital circuitry.

V. Modular Systems

Reported (software) ConvNets need tens, hundreds, or even thousands of convolutional filters to perform properly on real-world pattern recognition tasks. Consequently, if we want to provide a realistic hardware infrastructure for real-world applications, it will be essential to assemble hundreds or thousands of AER convolutional modules like the one shown in Fig. 3. Furthermore, the infrastructure needs to offer a good degree of reconfigurability and programmability, so that arbitrary ConvNet architectures could be implemented and tested easily.

Fig. 4 shows a conceptual solution to achieve this. It consists of a 2D array of modular convolutional units. Each unit includes a programmable-kernel convolution module (like the one in Fig. 3) plus a local router. Each module can receive input events from any four neighbors, and sends its own output events to one or more of

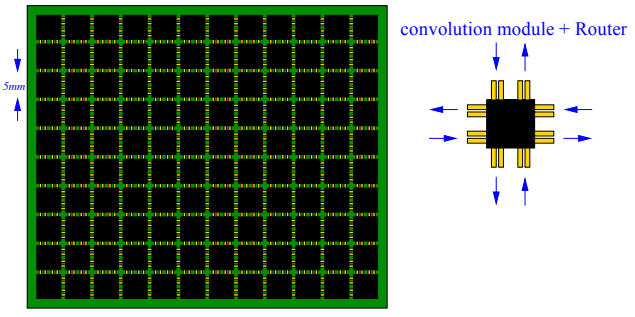
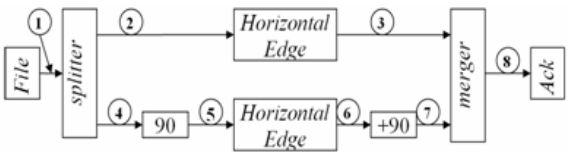


Fig. 4: Concept of modular ConvNet Structure




```
%First, we declare sources to the system
%         SOURCES   SOURCES DATA
sources   (1)       (data1)

%Next, we declare priorities
priorities (0.9,0.8,0.7,0.6,0.5,0.4,0.3,0.2)

%Next, we declare blocks
%NAME          IN-CHANNELS    OUT-CHANNELS   PARAMS          STATES
splitter       (1)            (2,4)          (params1)       (state1)
h_sobel        (2)            (3)            (params2)       (state2)
imrotate90     (4)            (5)            (params3)       (state3)
h_sobel        (5)            (6)            (params4)       (state4)
imrotate90     (6)            (7)            (params5)       (state5)
merger         (3,7)          (8)            (params6)       (state6)
ack            (8)            ()             (params7)       (state7)
```

Fig. 5: Example circuit (top) and its netlist description (bottom) of a 2-convolution (horizontal edge) ConvNet System.

its four neighbors. Each local router is programmed with a local routing table. The router detects, for each incoming event, the input port and decides to either process it by the local convolution module, or copy it to one or more of its output ports. The events produced by the local convolution module are also processed by the local router who sends them out through the proper output port(s). It is fairly easy to show that any arbitrary multi-filter architecture netlist can be implemented in this 2D structure by generating proper local routing tables for each module in the 2D array. Furthermore, such process can be automated by some kind of compiling software which, given a ConvNet netlist, would generate automatically all local routing tables. The modular 2D structure in Fig. 4 can be implemented physically using surface mount PCBs with miniature individual convolution chips with local router, or could be implemented with large chips of the type called Network-on-Chip (NoC), capable of hosting tens to hundreds of such modules per chip [17]. Multiple boards (or NoCs) can be further assembled hierarchically to scale up such systems.

VI. System-Level Behavioral Simulations

So far, it seems feasible to provide a hardware technology for spiking ConvNets based on AER. However, it is true that at present such large scale hardware systems have not been reported yet. Probably the largest AER system reported so far is the CAVIAR system [15], which uses four custom made AER chips (motion retina, convolution chip, winner-take-all chip, and learning chip) plus a set of FPGA based AER interfacing and mapping modules. The CAVIAR system includes 45k neurons, emulates up to 5 million synapses, performs an equivalent of 9 giga-connects-per-second, and can sense, identify and track objects with a 3*ms* delay. However, this system only has 4 convolution modules. Obviously, present-day AER hardware state-of-the-art is still not at the level of what is shown in Fig. 4 (with about 10^7 neurons emulating about 10^{11} synapses). In order to estimate the performance and evaluate the limitations one may encounter when assembling larger scale ConvNet with AER hardware, we have developed an event-based AER system simulator [38]. Fig. 5 shows an example circuit and netlist description used in this simulator. AER links are represented as "channels". At the end of the simulation, each channel would contain a list of all the events that have travelled through this channel including event information (such as its *x,y* address) and timing information (time at which the event was generated inside the emitter module, and time of physical use of channel). The input to the system (like "File" in Fig. 5) can be real physical events captured from a real AER retina and recorded as a file. Individual blocks in the netlist are behaviorally modelled, including all timing delays of handshaking signals, parasitic effects, finite precision effects, etc. This way, we can use real performance figures of already physically available AER chips/modules to model them in the simulator. The combination of real sensory event-format data

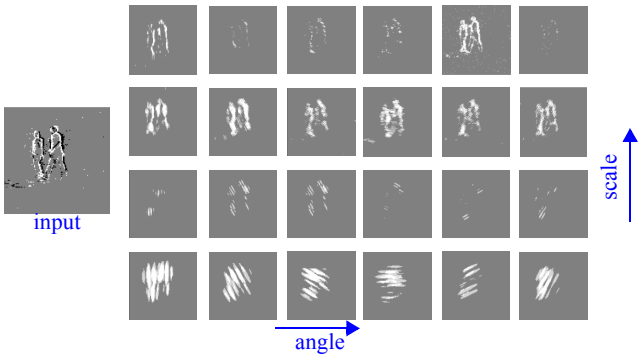


Fig. 6: Behavioral simulation results of a bank of AER Gabor filters of different scales and orientations over a physical sensory input obtained with an AER motion retina.

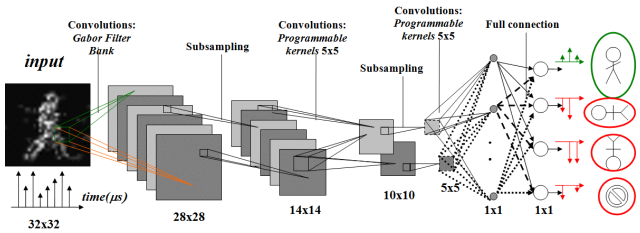


Fig. 7: Structure of spiking ConvNets trained for recognizing people from visual data captured from an AER motion retina

with performance figures of physical AER hardware allows to estimate reasonably well the performance of scaled-up systems.

As an illustrative example, Fig. 6 shows the simulation results of a bank of 24 Gabor filters with 4 scales and 6 orientations. The input is a 4 second 128x128 pixel motion retina recording of two persons walking from right to left, totaling about 130k events. Convolution modules compute their outputs as the events flow in. Each convolution module/chip needs about 100-200*ns* of computation time per input event [13]. After a few input events (about 10, depending on kernel) a convolution module provides its own output events. Consequently, the delay between input and output flow of events in a convolution module is of the order of microseconds (or fraction), making both flows in practice simultaneous. Fig. 6 shows for each convolution module and retina, the events captured during the same 80*ms*.

A bank of Gabor type filtering is usually the first stage of visual processing, like in the human brain [5]. For pattern and object recognition more stages are required. Fig. 7 shows an example ConvNet trained to recognize humans recorded with an AER motion retina. The input visual flow was captured with a physical temporal contrast (motion) AER retina [16] when observing people walking. A person walking produces about 3keps (kilo events per second). Visual pixel array was down sampled to 32x32. The spiking convolutional network has 7 layers. The first layer is a Gabor filter bank, second layer is subsampling, third layer is a trainable 5x5 kernels filter bank, fourth layer is subsampling, fifth layer is again a trainable 5x5 kernel filter bank, and sixth and seventh layers are fully connected trainable perceptrons. The system was trained off-line (using a frame-based representation) through back propagation learning to categorize inputs as vertical humans, up side down humans, horizontal humans, or other objects. After training, the learned weights were mapped from the frame-constraint system to a spiking event based system. It was tested with new spiking retina recordings, showing a correct recognition rate of above 86%.

The system was trained to categorize inputs as vertical humans, up side down humans, horizontal humans, or other objects, and was capable of performing correct recognition using retina recordings not used for training. Correct recognition was performed after receiving only between 50-80 retina events

(18-27ms of input stimulus) with a negligible throughput delay of a few microseconds.

VII. NanoTechnology Implementation Potential

Present-day arbitrary-kernel AER convolution chips compute by copying row-wise the kernel over its array of pixels [13]-[15],[35]. This is a sequential operation which introduces delays in the order of hundreds of nano-seconds, depending on kernel size. In present days we are witnessing a new trend micro and nano technologies, and new nano scale devices are being reported with high potential to be used as compact and trainable synapses [39]. A very promising option is to combine a CMOS chip underneath with a nano scale device fabric on top of it, which is known as a CMOL arrangement [40]. If each pixel has an input pin and an output pin, CMOL allows to physically connect any pixel output pin to any other pixel input pin, through an independent synapse made of a nano device. Therefore, full interconnectivity is possible. A convolutional (projection field) connectivity is a particular case of the full connectivity. Consequently, such a structure would be capable of computing event based convolutions without sweeping row-wise the kernels, thus achieving much higher speeds and event throughputs. Furthermore, by making the neurons send forward and backward electrical spike pulse, it is possible to equip this type of synapses with STDP (Spike-time-dependent-plasticity) learning [41]-[42], but this is beyond the scope of this paper. More details can be found in

VIII. Conclusions

We have shown how to implement ConvNets with spiking hardware to perform sophisticated pattern recognition task. Large scale systems have been emulated using a behavioral simulator, but using performance figures of already available hardware, together with real stimuli obtained with physical AER retina chips.

IX. Acknowledgements

This work was supported by EU grant 216777 (NABAB), Spanish grants TEC2006-11730-C03-01 (SAMANTA2) and TEC2009-10639-C04-01 (VULCANO), and Andalucian grant P06TIC01417 (Brain System). JAPC was supported by an Andalucian scholarship and CZR by an FPU scholarship.

X. References

[1] K. Fukushima, "Visual Feature Extraction by a Multilayered Network of Analog threshold Elements," *IEEE Trans. Syst. Sci. and Cybernetics*, vol. SSC-5, No. 4, pp. 322-333, Oct. 1969.

[2] K. Fukushima and N. Wake, "Handwritten Alphanumeric Character Recognition by the Neocognitron," *IEEE Trans. Neural Networks*, vol. 2, No. 3, pp. 355-365, May 1991.

[3] Y. LeCun and Y. Bengio, "Convolutional Networks for Images, Speech, and Time Series," in *The Handbook of Brain Science and Neural Networks*, M. Arbib (Ed.), Cambridge, MA: MIT Press, pp. 255-258, 1995.

[4] L. Bottou and Y. LeCun, "Graph Transformer Networks for Image Recognition," *Proc. of ISI.*

[5] T. Serre, L. wolf, S. Bileschi, M. Riesenhuber, and T. Poggio, "Robust Object Recognition with Cortex-Like Mechanisms," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 29, No. 3, pp. 411-426, March 2007.

[6] Jain V, et al., "Supervised learning of image restoration with convolutional networks," *IEEE Int. Conf. Comp.t Vis.* (ICCV), 2007.

[7] R. Collobert and J. Weston, "A unified architecture for natural language processing: deep neural networks with multitask learning," *Proc. Int. Conf. on Machine Learning* (ICML 08), pp. 160-167, 2008.

[8] R. Hadsell, P. Sermanet, M. Scoffier, A. Erkan, K. Kavackuoglu, U. Muller and Y. LeCun, "Learning Long-Range Vision for Autonomous Off-Road Driving," *J. Field Robotics*, 26(2):120-144, February 2009.

[9] D. S. Ricket, et al., "On the Self-Generation of Electrical Soliton Pulses," *IEEE J. Solid-State Circ.,* vol. 42, No. 8, pp. 1657-1668, Aug. 2007.

[10] H. P. Graf, E. Cosatto, L. Bottou, I. Durdanovic, V. Vapnik, "Parallel Support Vector Machines: The Cascade SVM,"

[11] S. Thorpe, D. Fize, C. Marlot, "Speed of processing in the human visual system," *Nature* 381: 520-2, 1996.

[12] S. Thorpe, et al., "SpikeNet: Real-time visual processing with one spike per neuron," *Neurocomputing* 58-60: 857-64, 2004.

[13] R. Serrano-Gotarredona, et al., "A Neuromorphic Cortical-Layer Microchip for Spike-Based Event Processing Vision Systems," *IEEE Trans. Circ. Syst. I*, vol. 53, No. 12, pp. 2548-2566, December 2006.

[14] R. Serrano-Gotarredona, et al., "On Real-Time AER 2D Convolutions Hardware for Neuromorphic Spike Based Cortical Processing," *IEEE Trans. on Neural Networks*, vol. 19, No. 7, pp. 1196-1219, July 2008.

[15] R. Serrano-Gotarredona, et al., "CAVIAR: A 45k-Neuron, 5M-Synapse, 12G-connects/sec AER Hardware Sensory-Processing-Learning-Actuating System for High Speed Visual Object Recognition and Tracking," *IEEE Trans. on Neural Networks*, vol. 20, No. 9, pp. 1417-1438, September 2009.

[16] P. Lichtsteiner, C. Posch, and T. Delbrück, "A 128x128 120dB 30mW Asynchronous Vision Sensor that Responds to Relative Intensity Change," *IEEE J. Solid-State Circ.*, vol. 43, No. 2, pp. 566-576, Feb. 2008.

[17] S. Vangal et al, "An 80-Tile 1.28 TFLOPS Network-on-Chip in 65nm CMOS," *IEEE* ISSCC07, pp. 98-99, Feb. 2007.

[18] D. B. Strukov and K. K. Likharev, "CMOL FPGA: a configurable architecture for hybrid digital circuits with two-terminal nanodevices," *Nanotechnology* 16, pp. 888-900, 2005.

[19] M. Sivilotti, *Wiring Considerations in Analog VLSI Systems with Application to Field-Programmable Networks*, Ph.D. Thesis, California Institute of Technology, Pasadena CA, 1991.

[20] J. Lazzaro, et al., "Silicon Auditory Processors as Computer Peripherals," *IEEE Trans. on Neural Networks*, vol. 4, pp. 523-528, May 1993.

[21] E. Culurciello, et al., "A biomorphic digital image sensor," *IEEE J. of Solid-State Circuits*, vol. 38, pp. 281-294, 2003.

[22] P. F. Ruedi, et al., "A 128x128, pixel 120-dB dynamic-range vision-sensor chip for image contrast and orientation extraction," *IEEE J. of Solid-State Circuits*, vol. 38, pp. 2325-2333, 2003.

[23] S. Chen, and A. Bermak, "Arbitrated Time-To-First Spike CMOS Image Sensor with On-Chip Histogram Equalization," IEEE Trans. VLSI Systems, vol. 15, No. 3, pp. 346-357, March 2007.

[24] M. Azadmehr, et al."A foveated AER Imager Chip," *Proc. of the IEEE Int. Symp. on Circ. and Syst.* (ISCAS2005), pp. 2751-2754, Kobe, Japan, 2005.

[25] K. A. Zaghloul and K. Boahen, "Optic nerve signals in a neuromorphic chip," *IEEE Trans. Biomed. Eng.,* vol. 51, no. 4, pp. 657-675, Apr. 2004.

[26] J. Costas-Santos et al., "A contrast retina with on-chip calibration for neuromorphic spike-based AER vision systems," *IEEE Trans. Circ. Syst. I, Reg. Papers,* vol. 54, no. 7, pp. 1444-1458, July 2007.

[27] P. Lichtsteiner, C. Posch, and T. Delbruck, "A 128x128 120dB 30mW Asynchronous Vision Sensor that Responds to Relative Intensity Change," *IEEE J. Solid-State Circuits*, vol. 43, No. 2, pp. 566-576, Feb. 2008.

[28] G. Cauwenberghs, et al., "An analog VLSI Chip with Asynchronous Interface for Auditory Feature Extraction," *IEEE Trans. Circ. Syst.*, vol. 45, pp. 600-606, May 1998.

[29] Chicca et al., "A Multichip Pulse-Based Neuromorphic Infrastructure and Its Application to a Model of Orientation Selectivity," *IEEE Trans. Circ. Syst. Part I*, vol. 54, No. 5, pp. 981–993, May 2007.

[30] M. Oster, et al., "Quantification of a spike-based winner-take-all VLSI network," *IEEE Trans. Circ. Syst. 1*, vol. 55, No. 10, pp. 3160-69, 2008.

[31] T. Teixeira, et al., "Event-Based Imaging with Active Illumination in Sensor Networks," *Proc. IEEE Int. Symp. Circ. Syst.*, pp. 644-647, ISCAS 2005.

[32] P. Vernier, et al., "An integrated cortical layer for orientation enhancement," *IEEE J. solid-State Circ.,* vol. 32, No. 2, pp. 177-186, Feb. 1997.

[33] T. Y. W. Choi, et al., "Neuromorphic implementation of orientation hypercolumns," *IEEE Trans. Circ. Syst. I,* vol. 52, No. 6, pp. 1049-60, 2005.

[34] T. Serrano-Gotarredona, et al., "AER image filtering architecture for vision processing systems," *IEEE Trans. Circ. Syst. II, Analog Dig. Signal Proc.,* vol. 46, No. 9, pp. 1064-1071, Sep. 1999.

[35] L. Camuñas-Mesa, et al., "Fully digital AER convolution chip for vision processing," Proc. 2008 IEEE Int. symp. Circ. & Syst. (ISCAS08), pp. 652-655, May 2008.

[36] K. Boahen, "Point-to-Point Connectivity Between Neuromorphic Chips Using Address Events," *IEEE Trans. Circ. Sys. II*, vol. 47, No. 5, pp. 416-434, 2000.

[37] Delorme A, et al., "Networks of integrate-and-fire neurons using Rank Order Coding B: Spike timing dependent plasticity and emergence of orientation selectivity," *Neurocomputing* 38-40, pp. 539-45, 2001.

[38] J. A. Pérez-Carrasco, et al., "On the Computational Power of AER Vision Processing Hardware," *Proc. Int. Conf. Circ. Design & Integ. Syst.*, 2007.

[39] D. B. Strukov, et al., "The Missing Memristor Found," *Nature,* vol. 453, pp. 80-83, 1 May 2008.

[40] D. B. Strukov and K. K. Likharev, "CMOL FPGA: a configurable architecture for hybrid digital circuits with two-terminal nanodevices," *Nanotechnology* 16, pp. 888-900, 2005.

[41] B. Linares-Barranco and T. Serrano-Gotarredona, "Memristance can explain Spike-Time-Dependent-Plasticity in Neural Synapses," *Nature Precedings* <http://hdl.handle.net/10101/npre.2009.3010.1> 31st March, 2009.

[42] B. Linares-Barranco and T. Serrano-Gotarredona, "Exploiting Memristance in Adaptive Asynchronous Spiking Neuromorphic Nanotechnology Systems," *Proc. IEEE NANO*, July 2009.

[43] J. A. Pérez-Carrasco, et al. , "On Neuromorphic Spiking Architectures for Asynchronous STDP Memristive Systems," *IEEE ISCAS* 2010.