Spike Events Processing for Vision Systems

R. Serrano-Gotarredona[1], T. Serrano-Gotarredona[1], A. Acosta-Jiménez[1], A. Linares-Barranco[2], G. Jiménez-Moreno[2], A. Civit-Balcells[2], and B. Linares-Barranco[1]

1 Instituto de Microelectrónica de Sevilla (IMSE-CSIC), Ed. CICA, Av. Reina Mercedes s/n, 41012 Sevilla, Spain.
2 Dpto. Arquitectura de Computadores, University of Sevilla, Sevilla, Spain

Abstract

In this paper we briefly summarize the fundamental properties of spike events processing applied to artificial vision systems. This sensing and processing technology is capable of very high speed throughput, because it does not rely on sensing and processing sequences of frames, and because it allows for complex hierarchically structured cortical-like layers for sophisticated processing. The paper includes a few examples that have demonstrated the potential of this technology for high-speed vision processing, such as a multilayer event processing network of 5 sequential cortical-like layers, and a recognition system capable of discriminating propellers of different shape rotating at 5000 revolutions per second (300000 revolutions per minute).

I. Introduction

Artificial man-made machine vision systems operate in a quite different way to biological brains. Machine vision systems usually operate by capturing and processing sequences of frames. For example, a video camera captures images at about 25-30 frames per second, which are then processed frame by frame to extract, enhance and combine features, and perform operations in feature spaces, until a desired recognition is achieved. Biological brains do not operate on a frame by frame basis. In the retina, each pixel sends spikes (also called events) to the cortex when its activity level reaches a threshold. This activity level may respond to different image properties like intensity, contrast, color, motion, etc. - properties which have been pre-computed within the retina before generating the spikes to be sent to the visual cortex. Very active pixels will send more spikes than less active pixels. When the retina responds to a stimulus, for example a moving profile, then those pixels sensing the profile will elicit a simultaneous collection of spikes which are strongly space-time correlated. The visual cortex receiving these spikes is sensitive to the space location where the spikes were originated and to the relative timing between them. This way, it can recognize and follow this moving profile. All these spikes are transmitted as they are being produced, and do not wait for an artificial "frame time" before sending them to the next processing layer. This way, in biological brains, strong features are propagated and processed from layer to layer as soon as they are produced, without waiting to finish collecting and processing data of whole image frames.

As an illustration, consider the setup in Fig. 1. On the left, a circular solid object (a ball) is observed by a motion sensing retina in the center. The pixels in this retina are sensitive to motion (changes in intensity). Consequently, at a given instant in time only the pixels on a circumference will become active. This means that the pixels on the same circumference will simultaneously fire spikes. Let us assume each pixel fires just one single spike. We may state that, at a given instant (or short time interval), the spikes produced by the retina are highly space-time correlated: in time because they are simultaneous and in space because they form a circumference of a certain radius. In Fig. 1, the output spikes of the retina are sent, through projection fields, onto the next processing layer. Suppose the projection fields are tuned to detect circumferences of a given radius range $R \pm \varepsilon$. Then, each spike produced by a pixel in the retina will be sent to a circumference (of radius R) of pixels in the projection-field layer in Fig. 1. This way, pixel '1' in the retina sends a spike to all pixels in circumference '1' of the projection-field layer. The same for pixels '2', '3', '4', and all others in the retina circumference. If the circumference sensed in the retina is of the same radius R than the projection-fields, as is the case in Fig. 1, then the pixel in the projection field layer that has the same coordinates as the central pixel of the retina circumference (pixel 'A'), will receive spikes from all active projection-fields. Consequently, this pixel will receive the strongest stimulus. The pixels in the projection-field layer can be made to fire a spike if their stimulus reaches a certain threshold. If this threshold is sufficiently high, only the central pixel 'A' in the projection field layer will generate an output, signaling that this is the center of the moving ball of radius R sensed by the retina. In general, projection-fields in biological neuro-cortical layers perform feature

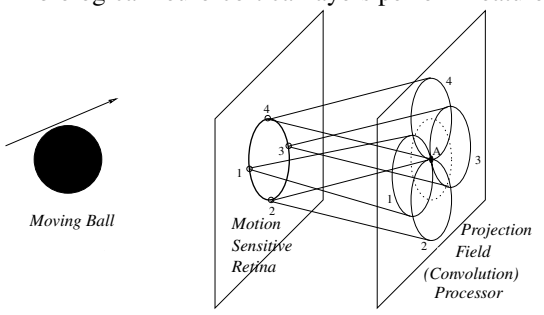

Fig. 1: Example of high-speed projection-field spike-based image processing for detecting a moving ball of a specific radius

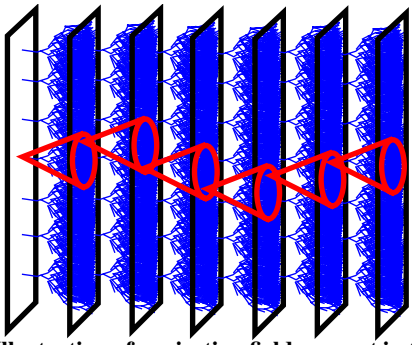

Fig. 2: Illustration of projection field concept in the brain. Each neuron in one layer connects to a projection field of neurons in the following layer. The weights of the connections follow a pattern which is independent of neuron position within a sending layer. Consequently, this is like applying a convolution from layer to layer.

extraction operations, which are dependent on the "shape" (weights) of the projection-field connections. Note that projection-field processing is equivalent to convolution processing, where the kernel of the convolution is the projection-field shape. In the case of Fig. 1, the feature to be detected is a circumference of radius R. In biological neuro-cortical structures there are several (8-10) sequential projection-field layers (see Fig. 2) that extract features [1]-[4], group them, extract more elaborate features, and so on, until in the end they perform complicated recognition tasks, such as handwritten character recognition [5]-[8] or face recognition [9]-[13].

A very interesting and powerful property of the projection-field processing illustrated in Fig. 1 and Fig. 2, is its high speed. In Fig. 1, note that the spikes produced at the retina are sent simultaneously to the projection field layer. The central pixel 'A' produces its output spike almost instantly. Consequently, this spike based projection-field processing approach is structurally much faster than a conventional frame-based processing approach. In a frame-based approach all pixels in a retina (or camera) are sensed and transmitted to the next layer (or processing stage), where all pixels of the frame are processed, usually with convolution operations, and so on. This frame convolution processing is slow, specially if several convolutions need to be computed in sequence for each input image frame. Artificial spike based processing hardware systems usually exploit the AER (Address-Event-Representation) technology.

II. Address-Event-Representation

AER was originally proposed by Mahowald and Sivilotti [14]-[16] as an inter-chip communication protocol to reproduce the state of a 2D array of neurons from one emitter chip onto another receiver chip, continuously and in real time. A growing community of researchers is using the scheme for bio-inspired vision [18]-[23] and audition [24] systems. Since then, the

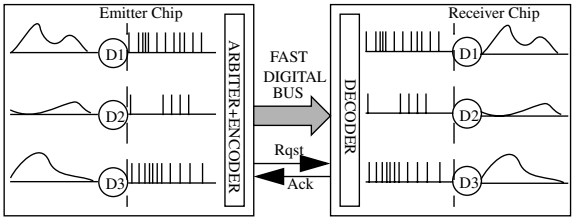

Fig. 3: Illustration of Address-Event-Representation (AER) Point-to-Point Communication

scheme has been evolving in efficiency and processing power.

Fig. 3 shows the essence behind the AER protocol. The emitter chip contains an array of cells or pixels whose intensity or activity changes in time with slow time constants. This happens, for example, in commercial cameras or artificial retinae where the bandwidth of the signal sensed by an individual pixel is in the order of hundreds of Hertz at the most. Each pixel contains an oscillator whose frequency is proportional to pixel intensity. The oscillator produces spikes of very short duration (in the order of *nano seconds*, for example 15*ns* [17]) but with much longer spike intervals (in the order of *mili seconds*). These spikes are called "events". Every time a pixel sends a spike, its (x, y) coordinate is written on an inter-chip high speed digital bus and sent to one or more receiver chips. Events are generated asynchronously. Therefore, additional handshaking signals are required for the proper transmission of events from chip to chip. Also, since events are generated asynchronously, "collisions" of events generated simultaneously by different pixels may occur. Several ways of handling collisions have been reported in the literature. One way is to detect and discard events that collide [18]-[20], while another is to introduce arbitration [26], [28], [29] and enforce sequencing of colliding events. The latter is more sophisticated but can handle much higher event traffic loads, although it will introduce small event delays (in the order of *nano* or maybe *micro seconds*). A channel will saturate when it has to handle a sustained event rate close to its physical bandwidth, or above. At that point, one can either (a) put more channels in parallel, or (b) decrease pixel maximum frequency and readjust system level parameters at subsequent stages to adapt to this. Solution (b) will slow down overall system speed.

In Fig. 3 each event produced by the emitter chip is received by one receiver chip. The receiver chip decodes the address of the event and sends it to the pixel with the same (x, y) coordinate. This pixel contains some type of integration mechanism that reconstructs the original low frequency time waveform of the same coordinate pixel in the transmitter chip. The delay between events produced in the emitter pixel until they are received by the receiver pixel is in the order of *nano seconds*. One can say the signals at the receiver pixels are identical and simultaneous to those in the emitter pixels, as if there were wires between pixels of the same coordinate.

However, the only physical wires between chips are the ones forming the high-speed digital bus, which has a relatively small number of pins compared to the number of pixels of the images[1].

III. Event Coding Schemes

When AER was first proposed, the information coding scheme considered was '*rate coding*'. This means that pixel activity level was represented directly as pixel event frequency. Therefore, to recover pixel activity one would need to integrate the events during a certain time interval. This '*rate coding*' principle has been also considered as biologically realistic by many neuroscientists during many years. However, because of recent discoveries, there are reasons to believe that biology not only codes information by '*rate coding*', but there might be also other plausible schemes that allow for more rapid information processing. For example, '*rank order coding*' [25] is an alternative scheme that exploits the ordering in time of set of simultaneous events. Or simply coding the '*synchronicity*' of pixels could be meaningful [27]. More simple solutions could be just to code the derivative of pixel activity [26] to detect changes. In principle, AER is not restricted to '*rate coding*', since AER consists only of sending events that code pixel addresses. The way those events are processed at the receivers (for example, by integration) is what puts restrictions on the coding schemes.

IV. AER Processing Capabilities

The AER protocol not only allows for a "virtual wiring" between pixels of emitter and receiver chips, but allows for extra processing on the addresses while they travel between chips. For example, image translation can be performed by inserting digital adders between chips, that would add fixed offsets to the travelling (x, y) coordinates. Image rotations could be performed by inserting properly coded look-up tables, as well as any arbitrary transformations and distortions. Even sophisticated micro-controller based approaches have been reported that generate "bubbles" of events for each original event [30]. In 1999 Serrano et al. introduced an architecture for performing AER based real-time programmable convolutions [23]. However, these convolution operations were limited to kernels $k(x, y)$ which are decomposable into x and y components $k(x, y) = h(x)v(y)$. More recent versions [31] do not suffer from this restriction and can be programmed to perform convolutions with arbitrary kernels. Other researchers presented in the past AER circuits for convolution processing. For example, Vernier et al. presented a chip with a fixed hardwired kernel, whose spatial shape could be slightly fine tuned through analog biases [20].

1. If there are N^2 pixels, only $n_w = \log_2(N^2)$ physical wires are required. If $N^2 = 128 \times 128 = 16384$ then $n_w = 14$.

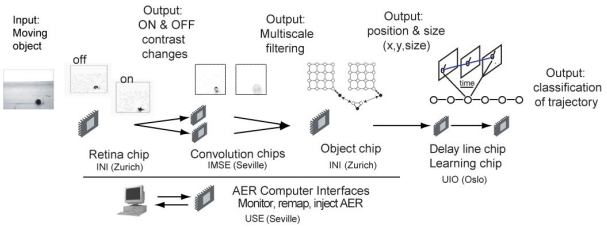

Fig. 4: Demonstration AER vision system

V. Multi-Chip-AER

A potentially huge advantage of AER systems is its capability for assembling many modules, while keeping its high speed processing. Fig. 4 describes a set of AER building blocks and how they were assembled into a prototype vision system that learns to classify trajectories of a moving object [32]. All modules communicate asynchronously using AER. The building blocks consist of: (1) a retina loosely modeled on the magnocellular pathway that responds to brightness changes [33], (2) a convolution chip with programmable convolution kernel of arbitrary shape and size [31], (3) a multi-neuron 2D competition chip [34], (4) a spatio-temporal pattern classification learning module [35], and (5) a set of FPGA-based PCBs for address remapping and computer interfaces [36]-[37].

Using these AER building blocks and tools we built the demonstration vision system shown schematically in Fig. 4, that detects a moving object and learns to classify its trajectories. It has a front end retina, followed by an array of convolution chips, each programmed to detect a specific feature with a given spatial scale. The competition or 'object' chip selects the most salient feature and scale. A spatio-temporal pattern classification module categorizes trajectories of the object chip outputs [32].

VI. High-Speed Convolutions

Recently, important advances of high speed processing for convolution based recognition have been reported [31]. For example, an experiment that demonstrates the high speed processing capabilities of AER based systems is the recognition of high speed rotating propellers. For this, a convolution chip is fed with a stimulus consisting of two rotating propellers. Each propeller has a different shape, as shown in Fig. 5. One is rectilinear, and the other has an S-like shape. When the propellers rotate at high speed one only sees a solid circle that moves slowly across the screen. Therefore, a human observer would not be able to discriminate between the two propellers. In this experiment an artificial sequence of events representing the rotating propellers is generated. This sequence of events was generated numerically and physically provided in real-time by an AER emitter PCB controlled by a host computer [36]-[37]. This PCB connects to the input AER port of a convolution chip[2] [31]. The input and output AER ports of the convolution chip were recorded simultaneously using a monitor PCB. All input

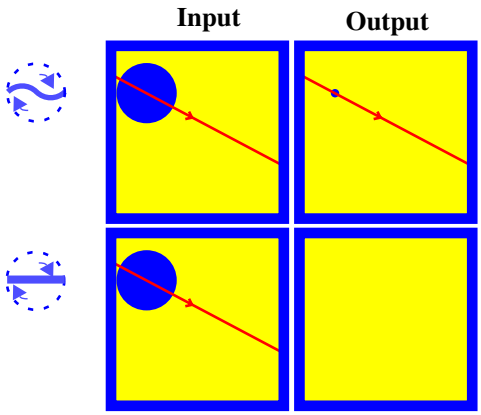

Fig. 5: Response of convolution Chip to two rotating propellers of different shape. Top row corresponds to an S-shaped propeller, while bottom row corresponds to a rectilinear propeller. The kernel programmed onto the chip is for recognizing the S-shaped propeller when it is in horizontal position. The left-hand columns show the input stimuli. Since the propellers are rotating at high speeds, only solid circles moving across the screen are seen. The right-hand columns show the output of the convolution processing. The top output detects the center of the S-shaped propeller. Consequently, in the top output a dot would be seen moving along the screen, which means that the S-shaped propeller is being followed. The bottom output is empty, since the convolution chip input is not an S-shape propeller.

and output events, conveniently time-stamped, can be recorded in computer memory.

In this experiment it was possible to provide propellers rotating at up to 5000 revolutions per second (300K revolutions per minute). The input stimulus is either the rectilinear propeller or the S-shape propeller. As shown in Fig. 5, the chip correctly discriminates between the two propellers.

VII. References

[1] H. Fujii, H. Ito, K. Aihara, N. Ichinose, and M. Tsukada, "Dynamical Cell Assembly Hypothesis - Theoretical Possibility of Spatio-Temporal Coding in the Cortex," *Neural Networks*, vol. 9, pp. 1303-1350, 1996.

[2] G. A. Orban, *Neural Operations in the Visual Cortex*, Springer-Verlag, Berlin, 1984.

[3] M. Shadlen and W. T. Newsome, "Noise, Neural Codes and Cortical Organization," *Current Opinion in Neurobiology*, vol. 4, pp. 569-579, 1994.

[4] G. M. Shepherd, *The Synaptic Organization of the Brain*, Oxford University Press, 3rd Edition, 1990.

[5] K. Fukushima: "Visual feature extraction by a multilayered network of analog threshold elements", *IEEE Transactions on Systems Science and Cybernetics*, SSC-5 (4), pp. 322-333 (Oct. 1969).

[6] K. Fukushima, S. Miyake, "Neocognitron: A new algorithm for pattern recognition tolerant of deformations and shifts in position," *Pattern Recognition*, vol. 15, pp. 455-469, 1982.

[7] K. Fukushima, "Neocognitron: A hierarchical neural network capable of visual pattern recognition," *Neural Networks*, vol. 1, pp. 119-130, 1988.

[8] K. Fukushima, "Analysis of the process of visual pattern recognition by the neocognitron," *Neural Networks,* vol. 2, pp. 413-420, 1989.

[9] Y. Le Cun and Y. Bengio, "Convolutional networks for images, speech, and time series," *Handbook of Brain Theory and Neural Networks*, M. A. Arbib (Ed.), pp. 255-258. MIT Press, Cambridge, MA, 1995.

[10] C. Neubauer, "Evaluation of Convolution Neural Networks for Visual Recognition," *IEEE Trans. on Neural Networks*, vol. 9, No. 4, pp. 685-696, July 1998.

[11] M. Matsugu, K. Mori, M. Ishi, and Y. Mitarai, "Convolutional spiking neural network model for robust face detection," *Proc. of the 9th Int. Conf. on Neural Information Processing* (ICONIP'02), vol. 2, pp. 660-664, 2002.

[12] B. Fasel, "Robust face analysis using Convolutional Neural Networks," *Proc. Int. Conf. on Pattern Recognition* (ICPR'02), pp. 40-43, 2002.

[13] M. Browne and S. S. Ghidary, "Convolutional Neural Networks for Image Processing: An Application in Robot Vision," *Advances in Artificial Intelligence: 16th Australian Conf. on AI*, pp. 641-652, November 2003.

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

[15] M. Mahowald, *VLSI Analogs of Neural Visual Processing: A Synthesis of Form and Function,* Ph.D. Thesis, California Institute of Technology, Pasadena CA, 1992.

[16] M. Mahowald, *An Analog VLSI Stereoscopic Vision System,* Kluwer Academic Publishers, 1994.

[17] J. Costas-Santos, T. Serrano-Gotarredona, R. Serrano-Gotarredona, and B. Linares-Barranco, "An AER Contrast Retina with On-Chip Calibration," *Proc. of the 2007 IEEE Int. Symp. on circ. and Syst.* (ISCAS07), 2007.

[18] A. Mortara and E. A. Vittoz, "A Communication Architecture Tailored for Analog VLSI Artificial Neural Networks: Intrinsic Performance and Limitations," *IEEE Trans. Neural Networks*, vol. 5, No. 3, pp. 459-466, 1994.

[19] A. Mortara, E. A. Vittoz, and P. Venier, "A Communication Scheme for Analog VLSI Perceptive Systems," *IEEE Journal of Solid-State Circuits*, vol. 30, No. 6, pp. 660-669, June 1995.

[20] P. Vernier, A. Mortara, X. Arreguit, and E. A. Vittoz, "An Integrated Cortical Layer for Orientation Enhancement," *IEEE Journal of Solid-State Cicuits*, vol. 32, pp. 177-186, February 1997.

[21] J. Kramer, R. Sarpeshkar, and C. Koch, "Pulse-Based Analog Velocity Sensors," *IEEE Trans. on Circuits and Systems, Part II*, vol. 44, pp. 86-101, 1997.

[22] E. Culurciello, R. Etienne-Cummings, and K. Boahen, "A Biomorphic Digital Image Sensor," *IEEE Journal of Solid-State Circuits,* vol. 38. No. 2, pp. 281-294, February 2003.

[23] T. Serrano-Gotarredona, A. G. Andreou, B. Linares-Barranco, "AER Image Filtering Architecture for Vision-Processing Systems," *IEEE Transactions on Circuits and Systems, Part I*, vol. 46, No. 9, pp. 1064-1071, September 1999.

[24] J. Lazzaro, J. Wawrzynek, M. Mahowald, M. Sivilotti, and D. Gillespie, "Silicon Auditory Processors as Computer Peripherals," *IEEE Transactions on Neural Networks,* vol.4, No 3, pp. 523-528, May, 1993.

[25] S. Thorpe and J. Gautrais, "Rank Order Coding," *Proc. 6th Annual Conf. on Comp. Neuroscience: Trends in Research*, Plenum Press, NY, USA, pp. 113-118, 1998.

[26] K. Boahen, "Retinomorphic Vision Systems," *Proc. of the 5th Int. Conf. on Microelectronics for Neural Networks and Fuzzy Systems,* Lausanne, pp. 2-14, February 1996.

[27] W. Maass and C. M. Bishop, *Pulsed Neural Networks*, The MIT Press, 1999.

[28] K. Boahen, "Point-to-Point Connectivity Between Neuromorphic Chips Using Address Events," *IEEE Trans. on Circuits and Systems Part-II*, vol. 47, No. 5, pp. 416-434, May 2000.

[29] K. Boahen, "A Thoughput-On-Demand Address-Event Transmitter for Neuromorphic Chips," in *Proc. 20th Aniversary Conf. Advanced Research in VLSI,* D. S. Wills and S. P. DeWeerth, Eds., pp. 72-86, 1999.

[30] D. H. Goldberg, G. Cauwenberghs, and A. G. Andreou, "Probabilistic Synaptic Weighting in a Reconfigurable Network of VLSI Integrate-and-Fire Neurons," *Neural Networks,* vol. 14, pp. 781-793, 2001.

[31] R. Serrano-Gotarredona, T. Serrano-Gotarredona, A. Acosta-Jiménez, and B. Linares-Barranco, "A Neuromorphic Cortical Layer Microchip for Spike Based Event Processing Vision Systems," to be published in *IEEE Trans. on Circuits and Systems, Part 1*, 2007.

[32] R. Serrano-Gotarredona, M. Oster, P. Lichtsteiner, A. Linares-Barranco, R. Paz- Vicente, F. Gómez-Rodríguez, H. Kolle Riis, T. Delbrück, S. C. Liu, S. Zahnd, A. M. Whatley, R. Douglas, P. Häfliger, G. Jimenez-Moreno, A. Civit, T. Serrano-Gotarredona, A. Acosta-Jiménez, B. Linares-Barranco, "AER Building Blocks for Multi-Layers Multi-Chips Neuromorphic Vision Systems" *Advances in Neural Information Processing Systems*, vol. 18, Y. Weiss and B. Schölkopf and J. Platt (Eds.), (NIPS'06), MIT Press, Cambridge, MA, pp. 1217-1224, 2006.

[33] P. Lichtsteiner, C. Posch, and T. Delbruck, "A 128x128 120dB 30mW Asynchronous Vision Sensor that Responds to Relative Intensity Change," *2006 IEEE ISSCC Digest of Technical Papers*, pp. 508-509, San Francisco, 2006.

[34] S-C. Liu and M. Öster, "Feature Competition in a Spike Based Winner-Take-All VLSI Network," *Proc. of the 2006 IEEE Int. Symp. Circ. and Syst.*, (ISCAS'06), pp. 3634-363637, May 2006.

[35] P. Häfliger, "Adaptive WTA with an analog VLSI neuromorphic learning chip," under review.

[36] F. Gomez-Rodriguez, R. Paz-Vicente, et al, "AER Tools for Communications and Debugging," *Proc. of the 2006 IEEE Int. Symp. Circ. and Syst.*, (ISCAS'06), pp. 3253-3256, May 2006.

[37] R. Paz-Vicente, A. Linares-Barranco, et al "PCI-AER Interface for Neuro-Inspired Spiking Systems," *Proc. of the 2006 IEEE Int. Symp. Circ. and Syst.*, (ISCAS'06), pp. 3161-3164, May 2006.

2. Note that this stimulus was generated artificially because presently, there is no AER motion retina capable of correctly sensing propellers rotating at up to 5000 revolutions per second. For example, the AER motion retina in [33] is able to sense rotations of up to 400-500 revolutions per second.