

THE NEUROMORPHIC ENGINEER

VOLUME 1, NUMBER 1

SPRING 2004

Beyond address-event communication: dynamically-reconfigurable spiking neural systems

Large-scale artificial sensory information-processing systems that emulate biological intelligence when interfacing with their surroundings have been the holy grail of neuromorphic engineering. The effort of our community has concentrated on modeling neural structures and adaptive mechanisms in biology as much as it has on efficient implementation in real-time micropower hardware.

However, despite steady advances in the scaling of VLSI technology, which promises to deliver more transistors on a single chip than neurons in our brain, it is neither feasible nor advisable to integrate the full functionality of a complete nervous system on a single chip. Early experimentation with neuromorphic systems revealed the need for a multi-chip approach and a communication protocol between chips to implement large systems in a modular and scalable fashion. Thus, the address-event representation (AER) protocol was developed over a decade ago and quickly became a universal 'language' for neuro-morphic engineering systems to communicate neural spikes between chips.¹⁻⁷ However, AER is now used for func-

tions in addition to inter-chip communication. The *Silicon Cortex* project proposed using AER to connect detailed compartmental models of neurons and synapses on multiple chips,^{8,9} and a few different groups have used AER to implement synaptic connectivity.^{4,7}

Here we will concentrate on our integrate-and-fire array transceiver (IFAT) chips, which can be used to implement large-scale neural networks in silicon with both synaptic connectivity and synaptic plasticity in the address-domain.^{10,11} The newest IFAT chip¹² implements 2,400 silicon neurons, each with a single dynamically-programmable conductance-like synapse: both the synaptic 'conductance' and the synaptic driving potential can vary for each incoming event. Rather than hardwiring connections between cells, the network architecture and synaptic parameters are stored off-chip in a RAM-based look-up table (LUT). An external digital microcontroller (MCU) provides the appropriate signals to configure synapses and route spikes to their respective targets via an asynchronous AER bus. A block diagram of the system is shown in Figure 1.

During normal operation, the event-driven microcontroller is activated when a presynaptic neuron generates a spike. The cell's address is used as an index into the LUT and the data stored at that location in RAM specifies one or more postsynaptic targets with their associated synaptic weights and driving potentials. The MCU then provides signals to the IFAT to configure each synapse and sends the events serially. Any postsynaptic spikes generated by this process can either be sent back to

In this issue:

- *Telluride*: Call for applications
- Smart building research at ETH/Caltech
- Wide dynamic range imaging from Ben Gurion
- University of Oslo on using the time domain
- Johns Hopkins research on using the time domain in analog circuits
- Caltech and MBARI collaborate to see jellies
- Imperial College vision chips for biomedical imaging
- CNRS robotic flight guidance
- *Laboratory Notes*:
The iLab Neuromorphic Vision C++ Toolkit from USC
- *Library Essentials*:
Analog VLSI & Neural Systems
- Mark Tilden and WowWee Toys introduce the Robosapien

the IFAT (in recurrent mode) or sent off-chip (in feed-forward mode), depending on the data stored in RAM. Additionally, updates to the network can be implemented by modifying the LUT according to a spike-based learning rule computed by the MCU.¹¹

A printed circuit board (Figure 2) inte-

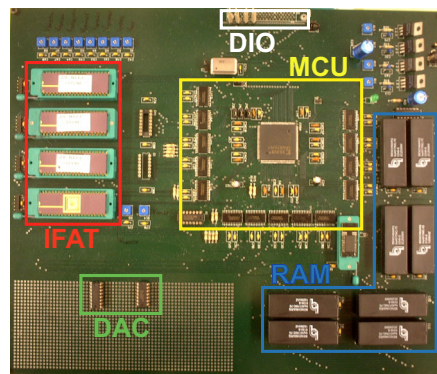
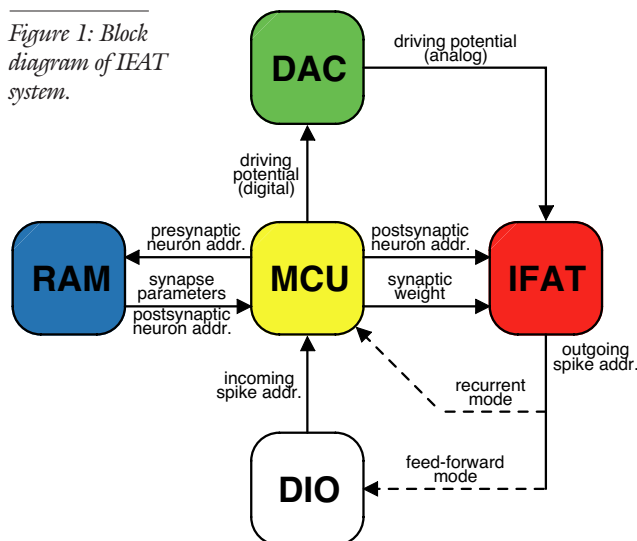


Figure 2. Printed circuit board integrating all components of the IFAT system.

grates the components of the IFAT system, including 9,600 neurons on four IFAT chips, 128MB of non-volatile SRAM, a high-speed 8bit voltage DAC, a 200MHz FPGA, and a 32bit digital I/O (DIO) interface. The DAC is used to control synaptic driving potentials, while synaptic weights are specified by three separate fields in the LUT: one each for the size of the postsynaptic potential, the number of events to send, and the probability of event transmission. External AER-compliant hardware or a peripheral computer interface

Figure 1: Block diagram of IFAT system.



Neuromorphic Engineering Workshop: call for applications

We invite applications for a three-week summer workshop that will be held in Telluride, Colorado, from Sunday, June 27 to Saturday, July 17, 2004. The application deadline is Friday, March 19 (see full application in-

structions on the web¹). The 2003 workshop and summer school, sponsored by the National Science Foundation, the Whitaker Foundation, the Office of Naval Research, the Defense Advanced Research Projects Agency, and the Center for Neuromorphic Systems Engineering at the California Institute of Technology, was an exciting event and a great success. We strongly encourage interested parties to browse through the previous workshop web pages.²

Goals

Carver Mead introduced the term 'neuromorphic engineering' for a new field based on the design and fabrication of artificial neural systems—such as vision systems, head-eye systems, and roving robots—whose architecture and design principles are based on those of biological nervous systems. The goal of this workshop is to bring together both young investigators and more established researchers from academia, as well as their counterparts in industry and national laboratories, to work on both neurobiological and engineering aspects of sensory systems and sensory-motor integration. The focus of the workshop will be on active participation, with demonstration systems and hands-on experience for all who take part.

Neuromorphic engineering has a wide range of applications: from nonlinear adaptive control of complex systems to the design of smart sensors. Many of the fundamental principles in this field, such as the use of learning methods and the design of parallel hardware (with an emphasis on analog and asynchronous digital VLSI), are inspired by biological systems. However, existing applications are modest and the challenge of scaling up from small artificial neural networks and designing completely autonomous systems at the levels achieved by biological systems lies ahead. The assumption underlying this three-week workshop is that the next generation of neuromorphic systems would benefit from closer attention to the principles found through experimental and theoretical studies of real biological nervous systems as whole systems.

Format

The three-week summer workshop will include background lectures on systems neuroscience (in particular learning, oculomotor and other motor systems and attention), practical tutorials on analog VLSI design, small mobile robots (Koalas, Kheperas, LEGO robots), hands-on projects, and spe-

cial interest groups. Participants are required to take part and possibly complete at least one of the projects proposed. They are furthermore encouraged to become involved in as many of the other activities proposed as interest and time allow. There will be two lectures in the morning that cover issues that are important to the community in general. Because of the diverse range of backgrounds among the participants, the majority of these lectures will be tutorials, rather than detailed reports of current research. These lectures will be given by invited speakers.

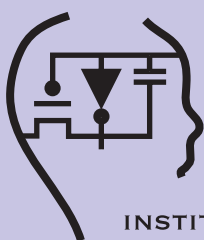
Participants will be free to explore and play with whatever they choose in the afternoon. Project/interest groups meet in the late afternoons and after dinner. Early afternoon there will be tutorial on a wide spectrum of topics including analog VLSI, mobile robotics, auditory systems, central pattern generators, and selective-attention mechanisms.

Projects that are carried out during the workshop will be centered in a number of working groups, including: active vision, audition, motor control, central-pattern generator, robotics, multichip communication, analog VLSI, and learning. The active perception project group will emphasize vision and human sensory-motor coordination. Issues to be covered will include spatial localization and constancy, attention, motor planning, eye movements, and the use of visual motion information for motor control. The central pattern generator group will focus on small walking and undulating robots. It will look at characteristics and sources of parts for building robots, play with working examples of legged and segmented robots, and discuss CPGs and theories of nonlinear oscillators for locomotion. It will also explore the use of simple analog VLSI sensors for autonomous robots. The robotics group will use rovers and working digital vision boards as well as other possible sensors to investigate issues of sensorimotor integration, navigation and learning. The audition group aims to develop biologically-plausible algorithms and aVLSI implementations of specific auditory tasks such as source localization and tracking and sound pattern recognition. Projects will be integrated with visual and motor tasks in the context of a robot platform. The multichip communication project group will use existing interchip communication interfaces to program small networks of artificial neurons to exhibit particular behaviors such as amplification, oscillation, and

Telluride, continued p. 11

THE NEUROMORPHIC ENGINEER

is published by the



INSTITUTE OF
NEUROMORPHIC
ENGINEERING

Editor

Sunny Bains
Imperial College London
sunny@sunmybains.com

Editorial Assistant

Stuart Barr
newsletters@sunmybains.com

Editorial Board

Avis Cohen
Ralph Etienne-Cummings
Rodney Douglas
Timothy Horvach
Giacomo Indiveri
Christof Koch
Shih-Chii Liu
Shihab Shamma
Chris Toumazou
André van Schaik

Available on the web at:

<http://www.ine-web.org>

The Institute of
Neuromorphic Engineering
Institute for Systems Research
AV Williams Bldg
University of Maryland
College Park, MD 20742
<http://www.ine-web.org>

Unsupervised learning and control provide ambient intelligence to smart buildings

Buildings are changing their nature from static structures of bricks and mortar to dynamic work and living environments that

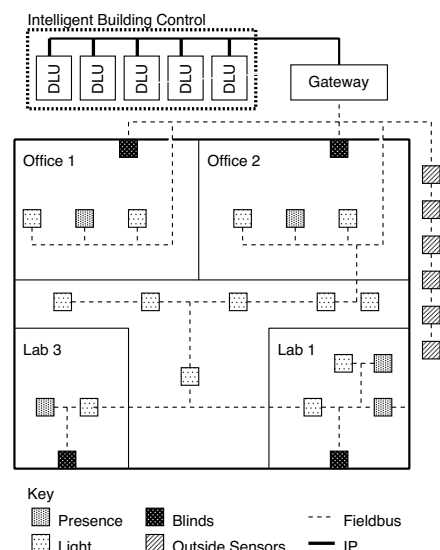


Figure 1. The floor of a typical building structured into rooms. All sensors and effectors are wired to a common fieldbus network. A gateway allows the agents to access the fieldbus network.

actively support and assist their inhabitants. These new buildings are expected to behave intelligently. In addition, to satisfying the needs of its inhabitants, a building is an active, autonomous entity that pursues its own goals (energy consumption, security). To fulfill this goal, a building must continually take decisions: but specifying rules that describe which actions to take, at which time, and because of which conditions, is complex and time consuming. In addition, these rules have to be changed constantly, because preferences and needs of users change. The building thus needs to learn its own rules of behavior, and continually adapt them, based on feedback from its occupants.

Engineering such a system poses a number of challenges. Decisions must be made in near-real time. The system must have a way to interact with its users to obtain feedback. On the other hand, it

should not intrude on the user. In this article we particularly highlight the problem¹ from a machine-learning perspective.

Unsupervised learning and control

We tackle the learning problem in several stages. All knowledge that the system has about its goals and users is stored as fuzzy logic² rules (in a rulebase). These rules are continually adapted based on feedback from the environment. A number of agents,³ each responsible for a small part of the whole decision space, use individual rulebases for decision-making. Using fuzzy rules has the advantage that the rulebase, although completely automatically constructed by the learning algorithm, is easily read and modifiable by humans.

Two-stage memory and decision process

The inherent non-stationarity and noisiness of user interactions with the building make it difficult to acquire stable, long-term knowledge. On one hand, the building must be able to retain and recognize long-term overall patterns of behavior; but on the other hand it must also react to short-term changes in requirements without destroying such long-term knowledge. Motivated by biological systems, we use a two-stage memory process: short-term (STM) and long-term (LTM) memory.⁴ Newly-acquired knowledge remains in STM and does not become part of LTM before it is confirmed and generalized. STM changes rapidly based on user demand and always gets precedence over LTM during day-to-day operation. However, if the former is discarded after a short time if it does not get incorporated into latter. Knowledge transformation from

STM to LTM, as in natural systems, takes place transparently: in parallel to decision making and learning. For efficient generalization of knowledge, a genetic algorithm⁵ is used (see Figure 2). This serves as an efficient mechanism to retain long-term knowledge in spite of inherent contradictions in feedback that occur in the short-term. This is a two-way process: knowledge in LTM, as in STM, is discarded if it is not needed anymore, or if it contradicts new facts. This property is essential because of the non-stationarity of the environment.

We have been running various versions of this system in a real building to evaluate its performance. Initial results are promising and show that the system is able to improve the inhabitants' comfort¹ and minimize energy consumption.

Ueli Rutishauser*†, Jonas Trindler† and Rodney Douglas†

*Computation and Neural Systems
California Institute of Technology

†Institute of Neuroinformatics
ETH/University of Zurich

E-mail: urut@caltech.edu

and trindler@ini.phys.ethz.ch

<http://www.ini.unizh.ch/~trindler/abi>

References

1. U. Rutishauser, J. Joller, R.J. Douglas, *Control and learning of ambience by an intelligent building*, *IEEE T. on Systems, Man and Cybernetics*, submitted, 2004.
2. R. Fuller, *Neural fuzzy systems*, Springer, Berlin, 2000.
3. G. Weiss, *Multiaagent systems*, The MIT Press, Cambridge, 2001.
4. H. Zhou, *CSM: A Computational Model of Cumulative Learning*, *Machine Learning*, 5, pp 383-406, 1990.
5. M. Mitchell, *An introduction to Genetic Algorithms*, The MIT Press, Cambridge, 1998.

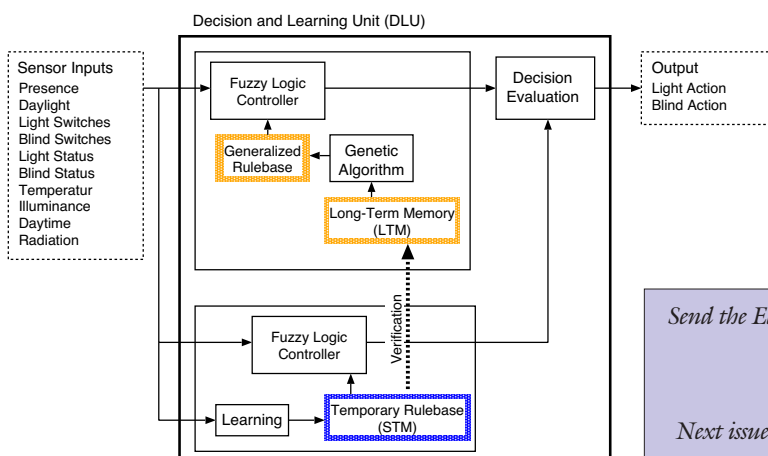


Figure 2. Decisions are taken based on fuzzy logic rules, which are stored in either long- or short-term memory. Feedback from the environment is used to continually adapt the rules.

Send the Editor ideas for the next issue by:

19 July 2004

Next issue copy due 17 September 2004.

Wide dynamic range (WDR) imaging

Interest in technologies that can help improve image sensors—in terms of cost, noise and dynamic range—is growing as quickly as the market for them. Dynamic range (DR) quantifies the ability of a sensor to image high-lights and shadows. If we define the dynamic range of the sensor as $20\log(S/N)$, where S is

for various conditions by changing its overall sensitivity. Thus, it still has a high dynamic range even at a single adaptation level. This is achieved with the assistance of three major mechanisms:¹ change in pupil diameter, neural adaptation, and photochemical adaptation.

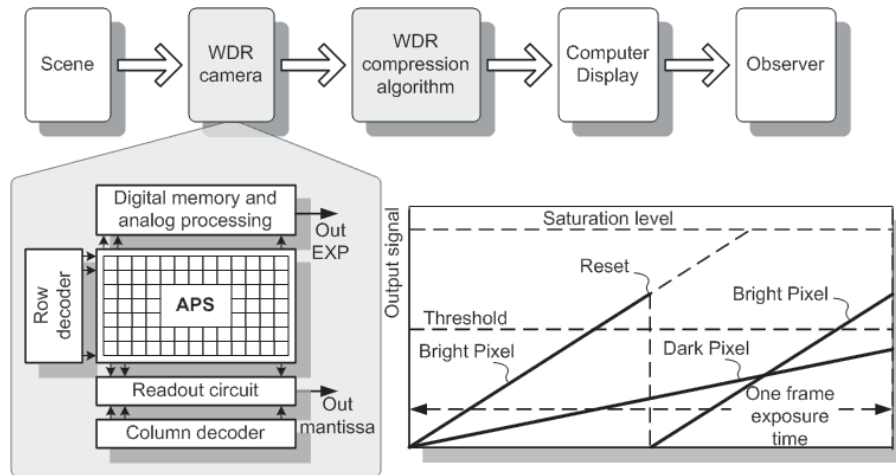


Figure 1. Imaging pipeline, image sensor architecture, and work principle.

the maximal signal value and N is the sensor noise, the typical image sensors will have a very limited dynamic range, about 65-75dB. Thus, dynamic range can be increased in two ways: noise reduction can expand dynamic range for darker regions, and raising the incident-light saturation level can improve it for brighter regions.

It is obvious that these HVS mechanisms are very complicated and cannot be exactly emulated. Nevertheless, we can try to approach some of them by designing smart image sensors with algorithms that allow an adaptive widening of the dynamic range. The overall task for WDR imaging can be divided into two distinguished stages: image capture, preventing the loss of scene details; and image compression, allowing image representation on conventional computer screens. The first stage is a challenge mainly for image-sensor designers, while the second stage can be performed also in software.

WDR imaging

Herein, we present one of the possible solutions for dynamic range extension in CMOS active pixel sensors (APS).²

As in a traditional CMOS APS, this imager is constructed of a two-dimensional pixel array, with random pixel-access capability and a row-by-row readout rolling-shutter method. Each pixel contains an optical sensor to receive light, a reset input, and an electrical output representing the illumination received. This imager implements a simple function for saturation detec-

tion, and is able to control the light exposure time on a pixel-by-pixel basis, resulting in no saturation. The pixel value can then be determined as a floating-point representation, similar to brightness adaptation in the HVS. To do so, the outputs of a selected row are read out through the column-parallel signal chain and, at certain points in time, are also compared with an appropriate threshold value, as shown in Figure 1.

If a pixel value exceeds the threshold—i.e. the pixel is expected to be saturated at the end of the exposure time—the reset is given at that time to that pixel. The binary information concerning the reset (i.e., if it is applied or not) is saved in a digital storage for later calculation of the scaling factor. Thus, in order to emulate the HVS, we can represent the pixel output in the following floating-point format: $M \cdot 2^{EXP}$. Here, the mantissa (M) represents the digitized pixel value, and the exponent (EXP) represents the scaling factor. This way, a customized, linear, large increase in the dynamic range is achieved.

Figure 2 compares an image captured by a traditional CMOS imager and by the auto-exposure system presented here. In Figure 2(a), a scene is imaged with a strong light hitting the object: hence, some of the pixels are saturated. At the bottom of Figure 2(b), the capability of the auto-exposure sensor for imaging the details of the saturated area in real time is shown. Since the display device is limited to eight bits, only the most relevant eight-bit part (i.e., the mantissa) of the thirteen-bit range of each pixel is displayed here. The exponent value, which is different for different areas, is not displayed here.

WDR compression

To allow image representation on conventional computer screens, compression of the acquired dynamic range is required. The resulting image should be as natural as possible, and should contain most of the details (both in bright and in shadowed areas) that a human observe would see if watching at the scene. Following space limitations, please see the reference for more details.³

Alexander Belenky, Evgeny Artyomov, Alexander Fish, and Orly Yadid-Pecht
VLSI Systems Center
Ben-Gurion University
http://www.ee.bgu.ac.il/~Orly_lab/

References

1. A. C. Guyton, *Textbook of Medical Physiology*, W. B. Saunders Co, 8th edition, 1991.
2. O. Yadid-Pecht, A. Belenky, Autoscaling CMOS APS with customized increase of dynamic range, *Proc. IEEE ISSCC*, San Francisco, California, February 4-7, 2001.
3. M. Herscovitz, E. Artyomov, O. Yadid-Pecht, Improving the global impression of brightness of the Multi Scale Retinex algorithm for wide dynamic range pictures, *Proc. SPIE 5017*, pp. 393-405, 2003.



Figure 2. (a) Scene observed with a traditional CMOS APS sensor. (b) Scene observed with our in-pixel auto exposure CMOS APS sensor

The human visual system (HVS) has overall dynamic range of about 200dB, but it cannot operate over the entire (enormous) range at the same time. Using the phenomenon called brightness adaptation, it adapts

On using the time domain for analog signal representation in electronic circuits

There is an ever-growing interest within the fields of neurophysiology—as well as within coding theory and signal processing—into the particular information transmission method employed by most neurons: spikes. Many models used today regard those volt-

with an interval by integrating that current on a capacitor during that interval. Thus, the output can be read as the voltage on that capacitor (second stage in Figure 1). These two stages in sequence can be described as an amplifier with voltage input

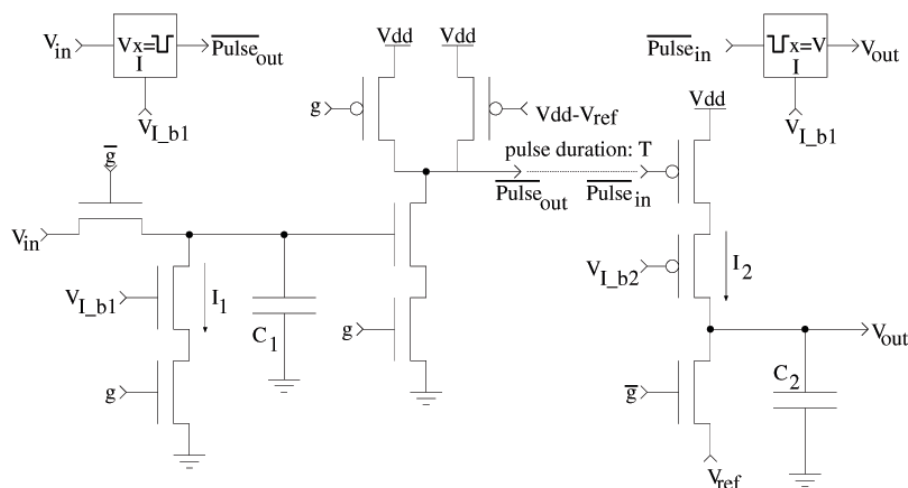


Figure 1. A voltage amplifier with internal time-domain representation.

age pulses as uniform events where only the occurrence of an event from a particular neuron at a particular moment in time matters. This signal representation offers many powerful encoding schemes, where information can be encoded into spike-event patterns over time and space. An analog value, for instance, can be encoded as an interval between two spikes.

Time-domain encoding and processing can also be used to advantage in electronic devices. So, besides representing signals with currents or voltages, they can just as well be represented with time intervals. This is, of course, not unknown in analog electronics, but since time-domain signals also *consume* time, they are not widely popular. They do, however, start to be increasingly important in neuronal models and, thus, in neuromorphic electronics.

The concept is not just limited to the imitation of biological functions, but can also be applied to more classical computations. Let us consider two simple examples for an analog division and multiplication circuit. A voltage can be divided by a current by measuring the interval it takes to deplete a capacitor from that voltage down to a zero reference. The output of that circuit is then that time interval, represented (for example) by a pulse width (first stage in Figure 1). Alternatively, a current can be multiplied

and output: it samples the input voltage when triggered (signal g), and its gain A is set by the ratio of two bias currents I_1 and I_2 .

$$A := \frac{V_{out}}{V_{in}} = \frac{C_1 I_2}{C_2 I_1} \quad \text{Equation 1}$$

The label $V_{dd}-V_{ref}$ refers to a voltage that can be derived on chip from V_{ref} and makes a NAND-like comparator switch as the voltage on C_1 becomes smaller than V_{ref} . Both g and its inverse are digital-level signals that start the computation and remain active during it. The duration of g limits the maximal range for the interval T and, thus, gives a lower limit for I_1 .

This amplifier circuit has a wide input range, i.e. between V_{ref} and V_{dd} , and it can be set to a wide range of different gains A , both greater and smaller than 1. It only consumes power when triggered, and internal currents are integrated over time, which reduces the total error. A source for error in the time domain will be the relatively slow switching of the pulse-generating comparator circuit. Because of the internal time-domain representation, it is not suited for fast or continuous computations. It is, however, well suited for event-triggered computations with a limited event frequency. Good examples of these are all kinds of spike-signal-processing neuromorphic circuits.

In a chip layout that is currently in production, two of these amplifier circuits were combined (Figure 2) to compute the weight

updates for a spike-based learning rule.¹⁻⁴ The specific rule implemented here⁵ changes a synaptic weight for every output spike (g is the neuron's output pulse, lasting 1ms) of the neuron according to Equation 2.

$$\Delta w = \alpha c - \beta w \quad \text{Equation 2}$$

There is an amplifier circuit for each multiplicative term. Both a and b are constant parameters and are the gain set by the two bias currents (I_1 and I_2 in Equation 1). I_1 (set by V_{Lb1}) is shared by the two first stages. In practice only the second parameter currents (as set by V_a and V_b) are used to set a and b . An average value of the recent input activity on this synapse is contained in c , and both it and (obviously) w are dynamically-changing variables that are represented as voltages in the circuit. The second stages of the amplifiers have been modified to transform the pulses into charge packages that are added and subtracted from the weight. On the real chip, the storage capacitor is actually a weak multi-level memory cell that is better suited for long-term retention of the weight voltage.⁶

These simple examples illustrate that time-domain signal representation can be used to advantage in electronic circuits, primarily to implement functions of neurophysiological models (like feature based motion-

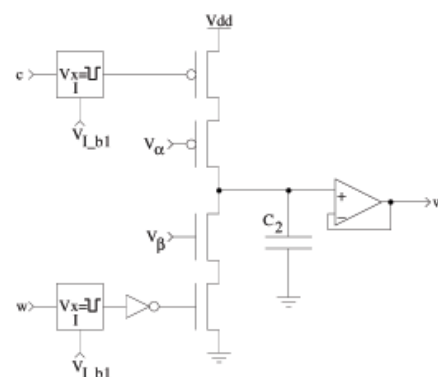


Figure 2. A weight update circuit for a spike-based learning rule.

processing⁷ or sound-localization⁸ systems), but also for all kinds of more basic functions. Other examples that have been published are, for example, a max circuit⁹ and a delay comparator.¹⁰

We do not claim that similar methods have never been applied before. But looking at time consumption as a resource, rather than

Häfliger, continued p. 9

The art of seeing jellies

The oceans contain a wealth of living creatures that account for a large amount of the biomass on our planet. How can we assess the kinds and numbers of animals in the oceanic water column? For more than a century, the traditional approach has been to tow nets behind ships. This method is limited in its spatial resolution, and because of the design of the nets, gelatinous animals (such as jellies, previously known as jelly fish) are destroyed and, hence, under-sampled. Today, remotely-operated underwater vehicles (ROVs) provide an excellent alterna-

to a computer-readable format, and some generic pre-processing is performed for each frame, such as subtracting the background, smoothing scan lines, and global contrast enhancement.

For the crucial detection step, we use an extended version of the Itti & Koch saliency-based attention algorithm³ (see also Itti's article in this issue). For this neuromorphic detection system for salient objects, each input frame is decomposed into seven channels for intensity contrast: red/green and blue/yellow double color opponencies, and four spatial orientations (0° , 45° , 90° , and 135°) at six spatial scales, yielding 42 'feature maps'. To improve the detection of faint, elongated animals, we introduced an additional across-orientation normalization step for the orientation filters, which is inspired by local interactions of orientation-tuned neu-

the most salient objects, however, we obtain a sparse number of objects whose predicted locations are usually separated far enough to avoid ambiguities. If ambiguities occur, we use a measure based on the distance of the objects from the predictions of the trackers and the size ratio of the detected and the tracked objects. Every couple of frames, the scene is again scanned for salient targets that are not already being tracked, and new trackers are initiated for these.

For each tracked object we obtain a binary mask that allows us to extract a number of low-level properties such as the object size, the second moments with respect to the centroid, the maximum luminance intensity, the average luminance intensity over the shape of the object, and its aspect ratio. We use these features to broadly classify the detected objects into those

that are interesting for the scientists, and those that are debris.

Since the occurrence of visible animals in the video footage is typically sparse in space and time, we can identify many frames that do not contain any objects of interest. By omitting these frames and marking candidate objects, we can enhance the productivity of human video annotators and/or cue a subsequent object classification module.

Our attentional selection and tracking system shows very promising results for transects from ROV dives that have been analyzed by human annotators already. This module is only the first step towards an integrated neuromorphic video annotation system that will consist of an object classification module and control mod-

ules for pan/tilt/zoom cameras: these in addition to the attentional module. This integrated system will be able to count the most common animals fully automatically.

Dirk Walther* and Duane Edgington†

*California Institute of Technology

E-mail: walther@caltech.edu

†Monterey Bay Aquarium Research Institute

E-mail: duane@mbari.org

<http://www.mbari.org/aved>

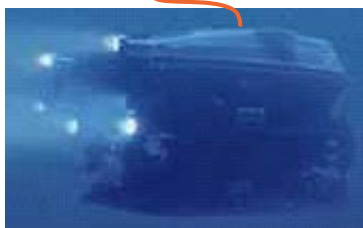
References

1. B.H. Robison, *The coevolution of undersea vehicles and deep-sea research*, *Marine Technology Society J.* **33**, pp. 69-73, 2000.
2. D. Edgington, D. Walther, K.A. Salamy, M. Risi, R.E. Sherlock, and C. Koch, *Automated Event Detection in Underwater Video*, *Proc. MTS/IEEE Oceans Conf.*, San Diego, California, 2003.
3. L. Itti, C. Koch, and E. Niebur, *A model of saliency-based visual attention for rapid scene analysis*, *IEEE Trans. on Pattern Analysis and Machine Intelligence* **20** (11), pp. 1254-1259, 1998.
4. R.E. Kalman and R.S. Bucy, *New Results in Linear Filtering and Prediction Theory*, *J. of Basic Engineering* **83** (3), pp. 95-108, 1961.

at sea



video recorded on digital tapes



ROV with HDTV camera

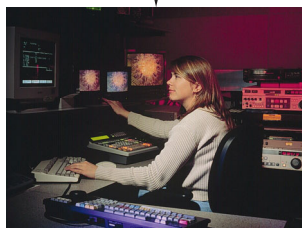
on shore



read digital tapes



video capture



manual annotation



automatic processing

Figure 1. The process flow for recording the video material at sea and processing it on shore either manually or automatically.

tive to nets for obtaining quantitative data on the distribution and abundance of oceanic animals.¹

Using video cameras, it is possible to make quantitative video transects (QVT) through the water, providing high-resolution data at the scale of the individual animals and their natural aggregation patterns. However, the current manual method of analyzing QVT video by trained scientists is very labor intensive and poses a serious limitation to the amount of data that can be obtained from ROV dives. To overcome this bottleneck in analyzing ROV dive videos we have developed an automated system for detecting and tracking animals for subsequent identification, based on neuromorphic vision algorithms.² These tasks are difficult due to the low contrast of many translucent animals and due to debris (known as 'marine snow') cluttering the scene.

Onboard the research vessel, the HDTV video signal from the ROV's broadcast-quality cameras is recorded on a digital BetaCam video deck. Back on shore, the videos are converted

to a computer-readable format, and some generic pre-processing is performed for each frame, such as subtracting the background, smoothing scan lines, and global contrast enhancement.

After iterative spatial competition for saliency within each map, only a sparse number of locations remain active and all maps are combined into a unique 'saliency map'. This is scanned by the focus of attention in order of decreasing saliency, through the interaction between a winner-take-all neural network (which selects the most salient location at any given time) and an inhibition-of-return mechanism (transiently suppressing the currently-attended location from the saliency map).³ Once salient targets have been detected, they are tracked from frame to frame using linear Kalman filters⁴ for the x and the y coordinates of the apparent motion of the objects in the camera plane: this assumes motion with constant acceleration. This is a good assumption for the constant-speed-heading motion of ROVs while obtaining QVTs.

Normally, tracking multiple targets at the same time raises the problem of assigning measurements to the correct tracks. Since our neuromorphic detection algorithm only selects

Neuromorphic electronics for real-time biomedical image processing

Algorithms for determining edges and contours,¹ and for computing object position, size, and centroid, have been well established in software.² However, stand-alone implementations of such systems remain limited by the high power dissipation and modular organization of conventional electronics. Such schemes require an external digital camera streaming the scan to a DSP or embedded PC that runs the image-processing software.

A new breed of neuromorphic³ or biologically-inspired vision chips is emerging that combine the imaging and front-end filtering with higher-order processing. Monolithic systems realized in standard technologies can provide low-cost and power-efficient solutions by using massively-parallel architectures and integrating the processing with the pixel circuitry. This approach prevents the need for the heavy communication bandwidths that are usually essential for real-time processing, as well as reducing power dissipation in the processing core. With use of novel weak-inversion circuits,⁴ real-time operation is now a realistic prospect.

ORASIS is a vision chip based on these principles, inspired by the unparalleled computational efficiency of neurobiology. It implements a novel, hybrid-neuromorphic-and-synthetic distributed algorithm⁵ for the analysis of microscopic cellular populations. The object segmentation part works as follows. First, continuous-time edge-detection provides the input for the contour-determination circuitry. When the contour has formed, a signal is initiated that fills inwards

until a unique centroid pixel is determined. Then, on completion, a localized reset reinitiates the sequence, realizing a pulsating action. By timing the resetting period, the object size is determined whilst the centroid pixel flags up its position. This process is illustrated in Figure 1. In addition to micropower operation, this system provides real-time data for population counts, using size and/or position discrimination (see Figure 2).

This distributed technique provides a robust solution against fabrication defects and process variations: factors that could render conventional algorithms unusable. Implemented on a quad-pixel connectivity scheme (square pixel), the chip has tolerance to fixed-pattern noise (FPN) up to $\pm 20\%$, as well as having multiple defect immunity. This is because—in such a distributed organization—

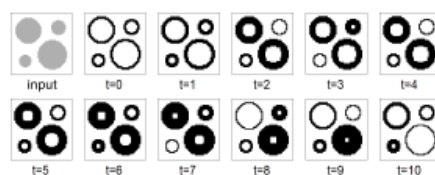


Figure 1. Computer simulation results of the bio-pulsating contour reduction algorithm, with snapshots taken at regular time intervals at the propagation delay of the processing.

the signal propagation can route itself around a defective node: thus permitting error-free operation as normal. Implementing this on a hexagonal-pixel connectivity scheme would further improve both robustness and accuracy.

Biological inspiration comes from the mammalian retina: our front-end circuitry includes continuous-time logarithmic photo detection, in addition to localized smoothing (averaging) and adaptive edge detection⁶ for signal conditioning. Furthermore, signal propagation based on localized interaction works in a similar way to the orientation-selective V1 cells in the primary visual cortex.⁷

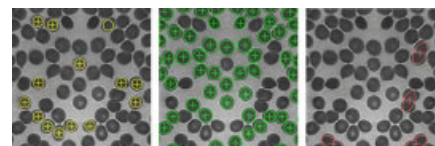


Figure 2. Illustration of image processing in ORASIS. Information such as cellular position and size can be computed on a real-time basis, enabling further discrimination by either size or position (window or proximity). Size discrimination for smaller to larger objects is shown from left to right.

Centroid determination is implemented using delay, propagate, integrate-and-fire-type neuron circuits: these are very similar to ganglion cells in the retina.

This system is currently being fabricated in a standard 0.18 μm CMOS (1P6M MM/Rf) process provided by UMC through IMEC, Belgium. The principal target specifications are given in Table 1.

Future research will be focused on developing ORASIS for dynamic cellular/microbial tracking in advanced biochemical analysis. Related research involves integrating real-time feature extraction and identification into endoscopic surgery.

Timothy G. Constandinou and Chris Toumazou

Department of
Electrical and Electronic Engineering
Imperial College London
E-mail: {t.constandinou, c.toumazou}
@imperial.ac.uk
<http://infoeng.ee.ic.ac.uk/~tc298>

References

1. M. Kass, A. Witkin, and D. Terzopoulos, *Snakes: active contour models*, *Int'l J. of Computer Vision* 1, pp. 321-331, 1988.
2. *Open source Computer Vision Library (OpenCV)*, *Int'l Res.*, <http://www.intel.com/research/mrl/research/opencv>.
3. C. A. Mead, *Neuromorphic Electronic Systems*, *Proc. IEEE* 78 (10), pp. 1629-1636, October 1990.
4. E. A. Vittoz, *Micropower Techniques*, in *Design of Analogue-Digital VLSI Circuits for Telecommunications and Signal Processing*, J. E. Franca and Y. Tsividis (eds.), Prentice Hall, 1994.
5. T. G. Constandinou, T. S. Lande and C. Toumazou, *Bio-pulsating architecture for object-based processing in next generation vision systems*, *IEE Electronics Lett.* 39 (16), pp. 1169-1170, August 2003.
6. T. G. Constandinou, J. Georgiou and C. Toumazou, *Nano-power mixed-signal tunable edge-detection circuit for pixel-level processing in next generation vision systems*, *IEE Electronics Lett.* 39 (25), pp. 1774-1775, December 2003.
7. Z. Li, *Pre-attentive segmentation in the primary visual cortex*, *Spatial Vision* Vol. 13 (1), pp. 25-50, 2000.

Technology		UMC 0.18 μm 1P6M CMOS
Supply voltage		1.8V
Dynamic Range		from 50mWm ⁻² to 5kWm ⁻²
Responsivity		0.2AW ⁻¹ m ⁻² @ λ =500nm
Maximum tolerable FPN		+/- 15%
Cellular	Cell Area	75 μm x 75 μm
	Active fill factor	16%
	Pixel power	18nW (typical)
	Edge power	20nW (maximum)
	Averaging power	95nW (typical)
	Logic power	5nW (maximum)
	Total online power	138nW
System	Chip area	5mm x 5mm
	Array resolution	50 x 50 cells
	Total array power	345 μW (typical)
	Total periphery power	100 μW (maximum)
	Total online power	345 μW
	Duty cycle (online)	10%
Total effective power		44.5 μW

Table 1. Target design specifications for ORASIS cell and system level hardware implementation.

Neuromorphic motion detection for robotic flight guidance

Flying insects use the relative motion of contrasts, or optical flow (OF), to detect and avoid obstacles. When contrasts move past successive *ommatidia* of an insect's compound

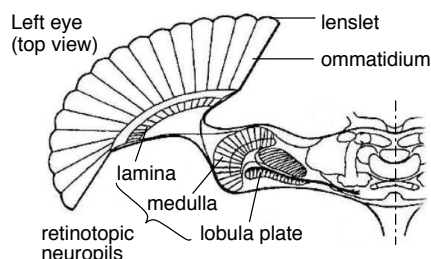


Figure 1. The blowfly's visual system.²

eye, the changing light intensities are converted in five stages (see Figure 1). These are: low-pass spatial filtering by ommatidial optics; light intensity conversion into DC signals by photoreceptors in ommatidia; signal-to-noise-ratio improvement in each *lamina* column; motion detection by elementary-motion-detector (EMD) neurons in the *medulla*; and motion signal aggregation by large-field directional neurons in the *lobula plate*. The reduced visual information contributes to the generation of flight commands. This article describes how an insect's visual-processing chain inspired the design of a flying robot's altitude-control system.¹

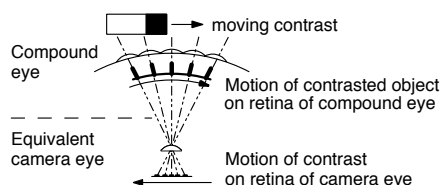


Figure 2. The equivalent camera eye.³

Light intensity sensing

A fly has 700-5000 ommatidia per compound eye. Each is capped by a lenslet whose underlying photoreceptors cover a visual field of a few degrees. Each photoreceptor has a bell-shaped angular sensitivity function (ASF) that prevents aliasing artefacts caused by spatial discretization of the compound eye.

For a robot, one can design a simpler camera eye that is equivalent to a compound eye for OF analysis (see Figure 2). The flying robot's camera assembly contains a one-dimensional, 20-pixel linear photoreceptor array and an aspheric lens (focal length 24mm) set at only 13mm from the array. Defocus is adjusted so that the ASF of each pixel is slightly wider than the interreceptor

angle. The eye is tilted down (-50°) so that its total field of view (FOV= 75°) covers the frontal and ventral region. Each photoreceptor in the array feeds an op-amp signal-conditioning circuit.

Motion detection

The aircraft's photoreceptor array feeds into 19 analog electronic EMDs, the operation of which is derived from electrophysiological experiments³ (see Figure 3a). As in insects, each EMD computes motion in a particular direction across the visual field of two adjacent photoreceptors that define two separate channels. Common to both is a band-pass filtering stage followed by thresholding and pulse-shaping stages. The first channel eventually triggers a long-lived, exponentially-decaying pulse, whereas the second produces a spike. The latter samples the exponential output of the first channel at a time that is inversely related to motion velocity. The EMD therefore outputs a pulse whose amplitude grows monotonically with velocity, while remaining largely invariant with contrast. EMD outputs are digitized by a PC.

Motion signal aggregation

Of the 60 large-field neurons of the lobula plate, ten vertical system (VS) neurons aggregate motion information along elevation planes. Neurophysiological data show that the frontal VS neuron in each eye is most responsive to downward motion directly ahead of the fly² (see Figure 3b). The robot's flight computer digitally aggregates frontal to ventral motion by using a weighted average rule that gives more weight to motion in the frontal than in the ventral FOV. The rule normalizes the OF distribution corresponding to a reference flight condition at a predefined altitude and flight speed over horizontal ground (see Figure 3c).

Flight guidance

Altitude control using OF was demon-

strated on a tethered rotorcraft (rotor diameter 34cm) using the rotor's collective pitch to vary thrust and a blown aerodynamic vane to regulate pitch (see Figure 4). The flight control system (FCS) commands thrust to vary the height above the ground so that in-flight OF is controlled with respect to the reference OF setpoint. If the OF is below this then the aircraft is too high and thrust is decreased, and vice versa. The paradigm supposes a constant flight speed that is regulated using an onboard pitch in-

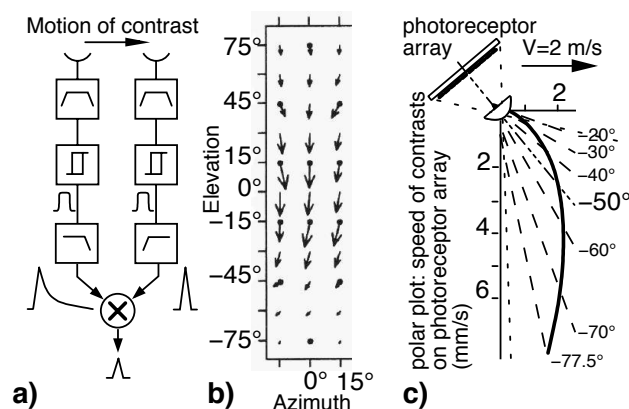


Figure 3. EMD model, vertical system (VS) neuron response,² and retina-motion distribution during horizontal flight at 2m/s and 5m above ground.¹

clinometer. Although neuromorphic motion detection uses discrete analog electronic components, the FCS uses a 20Hz Real-

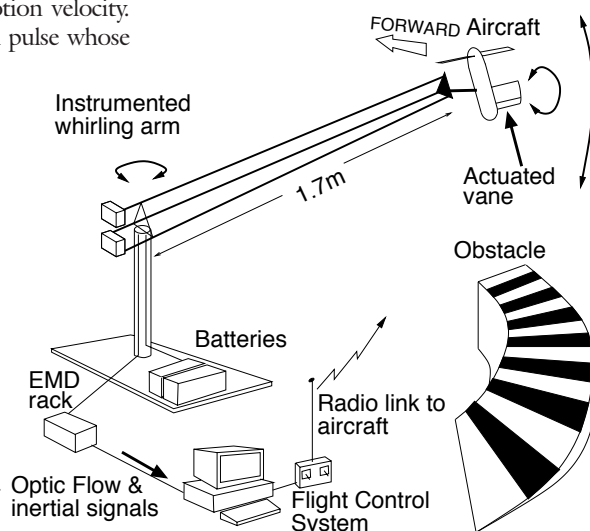


Figure 4. Rotorcraft on whirling arm.

Time Linux loop.

The weighted-average motion-based FCS generated smooth trajectories. The aircraft flew at nearly 3m/s and climbed a 30° ramp extending up to 1.5m (see Figure 5). Landing was simulated by voluntarily de

Netter, continued p. 11

Beyond address-event communication... from cover

can communicate with the IFAT through the DIO. The system is capable of implementing over four million synapses.

Previous generations of the IFAT system¹⁰ have served a variety of applications. For instance, Laplacian filters were implemented to isolate vertical edges on static images:¹⁰ a task that ran two orders of magnitude faster in hardware than in simulation. Similar network architectures can be employed to compute arbitrary filter kernels by varying the pattern of lateral connections between neurons. Even more interesting applications arise by extending address-event synaptic connectivity to address-domain synaptic plasticity. We implemented spike-based learning rules by monitoring the AER bus and dynamically updating the LUT.¹¹ Using this strategy with a form of spike-timing dependent plasticity (STDP), we constructed a network that could

detect correlated inputs and group them together. Subsequent work by other groups has demonstrated that the resulting networks are capable of preserving spike synchrony across multiple levels of neural processing.¹³ Finally, we recently built rudimentary neural spatio-temporal filters and used them to process a spike train produced by an AER retina.¹⁴ By constructing an array of similar elements and combining the appropriate outputs, it is possible to construct velocity-selective cells similar to those found in the medial-temporal cortical area (MT) of the human brain.¹⁵

We believe that by combining analog VLSI hardware with a digital microcontroller and RAM, reconfigurable hardware neural networks provide the 'best of both worlds': analog cells efficiently model sophisticated neural dynamics in real-time, while network architectures can be reconfigured and adapted

On using the time domain... from p. 5

a nuisance, is certainly not common among electronic engineers. This view, inspired by emerging neurophysiological coding models, can definitely give new impulse to circuit designs.

P. Häfliger

University of Oslo, Norway
E-mail: hafliger@ifi.uio.no

References

1. A. Bofill, A. F. Murray, and D.P. Thompson, *Circuits for VLSI implementation of temporally asymmetric Hebbian learning*, *Advances in Neural Information Processing Systems (NIPS)* 14, MIT Press, Cambridge, 2001.
2. S. Fusi, M. Annunziato, D. Badoni, A. Salamon, and D.J. Amit, "Spike-driven synaptic plasticity: theory, simulation, VLSI implementation", *Neural Computation* 12, pp. 2227-2258, 2000.

3. W. Gerstner, R. Kempter, J. L. van Hemmen, and H. Wagner, *A neuronal learning rule for sub-millisecond temporal coding*, *Nature* 383, pp. 76-78, 1996.
4. P. Häfliger, M. Mahowald, and L. Watts, *A spike based learning neuron in analog VLSI*, *Advances in neural information processing systems* 9, pp. 692-698, 1996.
5. P. Häfliger, *A spike based learning rule and its implementation in analog hardware*, Ph.D. thesis, ETH Zurich, Switzerland, 2000.
<http://www.ifi.uio.no/~hafliger>
6. P. Häfliger and H. Kolle Riis, *A multi-level static memory cell*, *Proc. IEEE ISCAS* 1, pp. 22-25, Bangkok, Thailand, May 2003.
7. J. Kramer, R. Sarpeshkar, and C. Koch, *Pulse-based analog VLSI velocity sensors*, *IEEE Trans. on Circ. and Sys.-II* 44 (2), pp. 86-100, February 1997.
8. J. Lazzaro and C. Mead, *A silicon model of auditory localization*, *Neural Computation* 1, pp. 41-70, 1989.
9. P. Häfliger and E. Jørgensen Aasebo, *A rank encoder: Adaptive analog to digital conversion exploiting time domain spike signal processing*, *Analog Integrated Circuits and Signal Processing*, accepted for publication in 2004.
10. P. Häfliger and E. Bergh, *An integrated circuit computing shift in stereopictures using time domain spike signals*, *Proc. 2002 NORCHIP* 20, København, November 2002.

in-site. The newest of these systems provide a number of advantages over previous designs, including more neurons, a richer parameter space, more biologically plausible dynamics, and a higher degree of inter-connectivity and plasticity. We expect them to serve as useful tools for future investigations of learning in large-scale neural networks. We invite the readers to contact us for collaborative opportunities on modeling of large-scale biological neural circuits.

R. Jacob Vogelstein*, Udayan Mallik, and Gert Cauwenberghs

Johns Hopkins University
{jvogelst, udayan, gert}@jhu.edu

*Corresponding author

References

1. M. Sivilotti, *Wiring considerations in Analog VLSI Systems, with application to Field-Programmable Networks*, PhD thesis, Cal. Inst. of Tech., 1991.
2. M. Mahowald, *An analog VLSI system for stereoscopic vision*, Kluwer Academic Publishers, Boston, MA, 1994.
3. J. Lazzaro, J. Wawrzyniec, M. Mahowald, M. Sivilotti, and D. Gillespie, *Silicon auditory processors as computer peripherals*, *IEEE Trans. Neural Networks* 4 (3), pp. 523-528, 1993.
4. K. A. Boahen, *Point-to-point connectivity between neuromorphic chips using address events*, *IEEE Trans. Circuits and Systems II* 47 (5), pp. 416-434, 2000.
5. C. M. Higgins and C. Koch, *Multi-chip neuromorphic motion processing*, in *Proc. 20th Anniversary Conference on Advanced Research in VLSI* (D. Wills and S. DeWeerth, eds.), pp. 309-323, IEEE Computer Society, Los Alamitos, CA, 1999.
6. S. Liu, J. Kramer, G. Indiveri, T. Delbruck, and R. Douglas, *Orientation-selective aVLSI spiking neurons*, in *Advances in Neural Information Processing Systems* 14, MIT Press, Cambridge, MA, 2002.
7. Interchip communication project group, *Report on the 2001 Workshop on Neuromorphic Engineering*, Telluride, CO, 2001.
8. S. R. Deiss, R. J. Douglas, and A. M. Whatley, *A Pulse-Coded Communications Infrastructure for Neuromorphic Systems, in Pulsed Neural Networks* (W. Maass and C. M. Bishop, eds.), pp. 157-178, MIT Press, Cambridge, MA, 1999.
9. M. Mahowald and R. Douglas, *A silicon neuron*, *Nature* 354, pp. 515-518, 1991.
10. D. H. Goldberg, G. Cauwenberghs, and A. G. Andreou, *Probabilistic synaptic weighting in a reconfigurable network of VLSI integrate-and-fire neurons*, *Neural Networks* 14 (6-7), pp. 781-793, 2001.
11. R. J. Vogelstein, F. Tenore, R. Philipp, M. S. Adlerstein, D. H. Goldberg, and G. Cauwenberghs, *Spike timing-dependent plasticity in the address domain*, in *Advances in Neural Information Processing Systems* 15, MIT Press, Cambridge, MA, 2003.
12. R. J. Vogelstein, U. Mallik, and G. Cauwenberghs, *Silicon spike-based synaptic array and address-event transceiver*, in *Proc. 2004 IEEE International Symposium on Circuits and Systems*, Vancouver, Canada, 2004 (forthcoming).
13. A. Bofill-i-Petit, A. Murray, *Synchrony detection by analogue VLSI neurons with bimodal STDP synapses*, in *Advances in Neural Information Processing Systems* 16, MIT Press, Cambridge, MA, 2004.
14. R. J. Vogelstein, R. Philipp, P. Merolla, S. Kalik, and R. Etienne-Cummings, *Constructing spatiotemporal filters with a reconfigurable neural array*, *2003 Workshop on Neuromorphic Engineering*, Telluride, CO, 2003.
15. R. Etienne-Cummings, J. Van der Spiegel, and P. Mueller, *Hardware Implementation of a Visual Motion Pixel using Oriented Spatiotemporal Neural Filters*, *IEEE Trans. Circuits and System II* 46 (9), pp. 1121-1136, 1999.

Biologically Inspired Cognitive Systems (BICS 2004)

Stirling, Scotland, August 29 - September 1 2004

Includes tracks on:

Neuromorphic Systems

Sensor and Sensory Systems - Computer Vision, Audition, Olfaction; High-level Perception; Intelligent Sensor Fusion and Sensor/motor integration; Smart Human-machine Communication; Autonomous Robots; Behavior-based Control; and Hardware and Software Implementations.

Neurophysiologically Inspired Models

Neuro-physiological foundations; Spiking neuron models and neuron assemblies; Models of brain centers and sensory pathways; Sensation, Perception and Attention; Spatio-temporal Orientation; Reactive Behaviour

Paper call deadline extended to 29 February 2004.

<http://www.icsc-naio.org/conferences/bics2004/bics-cfp.html>

LABORATORY NOTES

The iLab Neuromorphic Vision C++ Toolkit: Free tools for the next generation of vision algorithms

Because of its truly interdisciplinary nature—benefiting from the latest advances in experimental and computational neuroscience, electrical engineering, control theory, and signal and image processing—neuromorphic engineering is a very complex field. This has been one of the leading motivations for the development of a Neuromorphic Vision Toolkit at the iLab of the University of Southern California to provide a set of basic tools that can assist newcomers in the field with the development of new models and systems. More generally, the iLab Neuromorphic Vision C++ Toolkit project aims at developing the next generation of vision algorithms, the architecture of which will closely mimic the neurobiology of the primate brain rather than being specifically developed for a given set of environmental conditions or tasks. To this end, it provides a software foundation that is specifically geared towards the development of neuromorphic models and systems.

At the core of the toolkit are a number of neuroscience models, initially developed to provide greater understanding of biological vision processing, but here made ready to be applied to engineering challenges such as visually-guided robotics in outdoor environments. Taken together, these models provide general-purpose vision modules that can be easily reconfigured for, and tuned to, specific tasks. The gross driving architecture for a general vision system, the basis of many of the modules available in the toolkit, is shown in Figure 1.

Input video, whether captured by camera or from other sources, is first processed by a bank of low-level visual feature detectors. These are sensitive to image properties such as local contrast, orientation, or motion energy and mimic the known response properties of early visual neurons in the retina, lateral geniculate nucleus of the thalamus, and primary visual cortex. Subsequent visual processing is then split into two cooperating streams. The first is concerned with the rapid computation of the 'gist' and layout of the scene and provides coarse clues by which the system obtains a sense of the environmental conditions (e.g., indoors vs. outdoors, on a track vs. off-road) and of its position within the environment (e.g., path is turning left, the scene is highly cluttered). The second stream is concerned with directing both attention and the eyes towards the few most visually-conspicuous objects in the scene. This

stage relies on a neural saliency map, which gives a graded measure of 'attractiveness' to every location in the scene, and is modeled on the neural architecture of the posterior parietal cortex in the monkey brain. At any

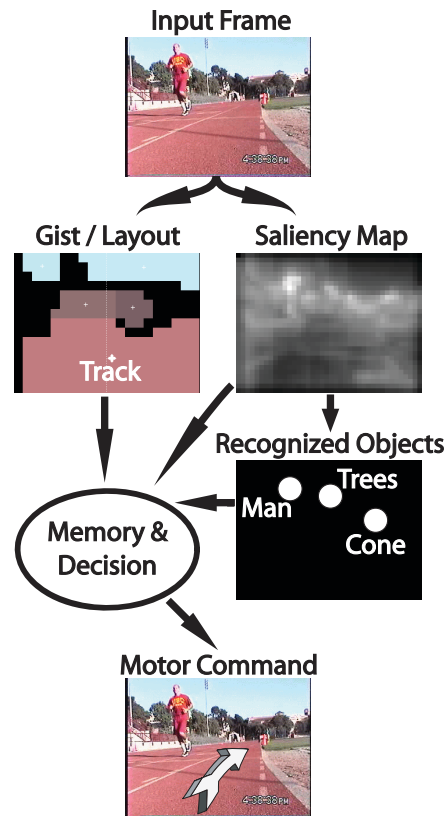


Figure 1. The iLab Neuromorphic Vision C++ Toolkit provides software modules for the implementation of some of the major components of primate vision, as depicted here. These include low-level visual processing algorithms, bottom-up visual attention and a saliency map, object recognition, as well as algorithms for the rapid computation of scene-level gist and rough layout. When used on a robot, the toolkit additionally provides modules for image and video digitization, short- and long-term symbolic and spatial memories, decision processes, and control of actuators. The source code for the toolkit is freely available.²

given point in time, the system uses the gist for basic orientation in the scene and sequentially attends to interesting objects (which could be obstacles, landmarks to aid navigation, or target objects being looked for).

Several neural models are available in the

toolkit for the implementation of the next processing stage—concerned with identifying the object that has drawn the attention and the eyes—and most of these models are inspired by the visual-response properties of neurons in the infero-temporal cortex. Finally, additional modules are available for short-term and long-term memory, cognitive knowledge representation, and modulatory feedback from a high-level task definition (e.g., look for the stop sign) to the low-level visual processing (e.g., emphasize the contribution of red to the saliency map, prime the object recognition module for the 'traffic sign' object class).

Not all of the components shown in the figure have been fully implemented, and many are at a very preliminary stage of development: including some simply not yet existing. The interesting point to note already, however, is how the biologically-inspired visual system architecture proposed here is very different from typical robotic-vision and computer-vision systems, which are usually defined to solve a specific problem (e.g., find a stop sign by looking for its specific shape using an algorithm matched to its exact geometrical properties). This promises to make the systems developed around this architecture particularly capable when dealing with novel complex outdoors scenes and unexpected situations, as has been widely demonstrated by, for example, our model of bottom-up attention.¹

The entire source code for the iLab Neuromorphic Vision C++ Toolkit is distributed freely upon request, in an effort to encourage more researchers to explore the potential of neuromorphic vision for real-world applications.

For more information about the iLab Neuromorphic Vision C++ Toolkit, see our web page.²

Laurent Itti
University of Southern California

References

1. <http://iLab.usc.edu/bu/>
2. <http://iLab.usc.edu/toolkit/>

If have a toolkit or other goodies to share, contact the Editor at sunny@sunnybains.com

LIBRARY ESSENTIALS

1986 in Carverland—as we called Carver Mead’s lab at Caltech—was a golden year, a year of promise, a year of seemingly endless possibilities. This was the birthplace (at least in our minds) of neuromorphic analog VLSI. In fact the word *neuromorphic* had yet to be coined by Carver, and he was still in the process of writing the book. Here we are roughly 15 years later, and where do things stand? Has this book changed the world?

The short answer is a qualified *yes*. It reminds one of the old joke, recast by Rodney Douglas, about the girl who is telling her friend about her most recent date: “He was a neuromorphic engineer,” she said, “which sounded really promising. I like engineers, they are so practical. But in the end it was a pretty pointless date—he spent the evening sitting on the edge of the bed, rubbing his hands together and saying, ‘This is gonna be great, this is gonna be great, this is really gonna be great!’” The community of neuromorphic engineers still thinks it’s gonna be great and are in the process of proving it, but the industry is going on with what it always does best: incremental improvement to existing technology.

But what about the book? Hardcore EEs who are interested in excursions into nontraditional (i.e. neural) realms of circuit design will find this book fascinating from a cultural perspective. Nary a mention of strong inversion (except in Mary Ann Maher’s appendix). Not a single converter, PLL, bandgap reference, switch-cap, or two-stage op-amp design will befuddle them with yet another convoluted evolutionary improvement. Indeed, transistor mismatch is hardly mentioned in the entire book, although this has limited application of subthreshold design more than any other single factor. Instead they are treated

Analog VLSI and Neural Systems by Carver Mead, Addison Wesley, 1986

to beautiful conceptual and quantitative (backed by real data) discussions of analog circuit design in subthreshold using biological neural circuit principles as inspiration. However, they will also not find the book particularly helpful in actually designing functional chips: the kinds that people presently pay real money for. For that lore they would be better off with Holcomb or Sansen or Razavi.

But that was never its purpose to begin with. Its explicit purpose was to lay out—in what turned out to be a series of extremely well-written chapters—an approach to circuit and system design that is explicitly neural, that eschews clocks and linearity, and that embraces signal aggregation, collective computation, and the use of device physics for computation. In this respect it succeeds remarkably in the sense that these systems were actually built and they actually work: at least if you take work to mean that these systems can be beautifully demonstrated. Unfortunately, they have been largely useless for any application (so far). However, the lead time for technology can be very long, especially when fighting an entrenched approach. Remember steam? Vacuum tubes? The same could apply to synchronous logic nowadays.

For people not in the business of actually building chips themselves, the book offers the clearest introduction to the basic device physics of transistors of any book that I have seen. The first couple of chapters show how semiconductors work, how subthreshold conduction mechanisms work, and how subthreshold transistor physics is intimately related to the device physics of voltage-activated nerve channels. I recall how I, as a neophyte in the business (although supposedly trained in physics), had my eyes opened as to what was

really going on. It helped me grok* what was happening, but I needed Andy Grove’s famous book on semiconductors to actually do some things I wanted to do.

The later chapters of the book clearly explain some basic circuits traditionally used in CMOS design, but with the distinction that the treatment is strictly subthreshold and quantitative without losing intuition. A number of unique and useful circuits not found in traditional EE books are also taught: a hysteretic differentiator, a horizontal resistor circuit, and a generic second-order section among them. The beautiful chapter on small-signal analysis of second-order systems using the canonical form is one of the most useful tricks that I have ever learned (I still use it routinely today).

Finally, several appendices round out the book with descriptions of early versions of complete systems, including Misha Mahowald’s silicon retina and Mary Ann Maher’s charge-based device model. Better ways of doing many of the circuits used in these systems are now known, but they have never since been so clearly explained.

The book is thoughtfully and carefully written and is beautifully designed and produced.

Tobi Delbrück

Institute for Neuroinformatics
ETH/University of Zurich, Switzerland
<http://www.ini.unizh.ch/~tobi/>

(*For the uninitiated, the word ‘grok’ is from Robert A. Heinlein’s novel *Stranger in a Strange Land*, and means to understand something deeply and exhaustively. Definition paraphrased from *The New Hacker’s Dictionary*—Ed.)

Telluride... from p. 2

associative memory. Issues in multichip communication will be discussed.

This year we will also have some new robots kindly donated by the WowWee Toys division of Hasbro in Hong Kong. This will permit us to carry out experiments with WowWee/Hasbro hardware through Mark Tilden (see article on page 12).

Location and arrangements:

The summer school will take place in the small town of Telluride, 9000 feet high in Southwest Colorado, about six hours drive away from Denver (350 miles). All facilities within the beautifully-renovated public school building are fully accessible to participants with

disabilities. Participants will be housed in ski condominiums within walking distance of the school. Participants are expected to share condominiums. The workshop is intended to be very informal and hands-on.

Participants are not required to have had previous experience in analog VLSI circuit design, computational or machine vision, systems level neurophysiology or modeling the brain at the systems level. However, we strongly encourage active researchers with relevant backgrounds from academia, industry, and national laboratories to apply, in particular if they are prepared to work on specific projects, talk about their own work, or bring demonstrations to Telluride (e.g. robots, chips, software). Unless otherwise arranged with one of the organizers, we expect

participants to stay for the entire duration of this three week workshop.

Avis Cohen, University of Maryland,
Rodney Douglas, Institute of Neuroinformatics, Uni/ETH Zurich,
Ralph Etienne-Cummings, Johns Hopkins University, **Timmer Horiuchi**, University of Maryland, **Giacomo Indiveri**, Institute of Neuroinformatics, Uni/ETH Zurich, **Christof Koch**, California Institute of Technology, **Terrence Sejnowski**, Salk Institute and UCSD, **Shihab Shamma**, University of Maryland

References

1. <http://www.ini.unizh.ch/telluride/announcement.html>
2. <http://www.ini.unizh.ch/telluride/>

Neuromorphic robot humanoid to step into the market

WowWee Toys and I have teamed up to develop what we hope will be the first entertainment humanoid robot useable by and for everyone. The RoboSapien, developed using neuromorphic principles, is a 14in- (36cm-) high, remote-controlled, programmable, fast-reflex robot with no boot time, no special batteries, no plugs, no setup, and no com-



puter degree required. Best yet, for budding roboticists, it's also designed to be hacked.

The technology, called Applied Bio-morphic Robotics (a body-side neuromorphic approach started in 1988) looks to optimize performance-to-silicon ratios for real-world mechanisms. Exhausting classical approaches to adaptive machine control as too expensive, CPU intensive, or power extensive, we found the solution in analog non-linear necklace functions, or Nervous Networks (Nv). While classical neural nets are parallel arrays of integrators adept at retaining patterns, nervous networks are series loops of differentiators excellent at generating them.¹ Using only handfuls of transistors we found that symmetric Nv extended oscillators—when mapped into limbed robot bodies—automatically exhibited efficient locomotion competence. We further found that the loads generated by the motors directly modified the oscillator processes so that each motor became its own inertia-damped sensor.

This reduces robot control electronics to almost nothing, and as Nv systems are automatically self-optimizing, no programming is required either. This approach meant extremely minimal, lightweight machines that are more leg than brain: this allows for auto-

adapting mechanics supporting minimal digital control. Like a smart horse carrying a dumb cowboy, the combination implies competence more than the sum of the parts, and we think this comes across fairly obviously in our humanform robot.

The RoboSapien is quick, funny, walks, crawls, picks up, throws, pushes, pulls, dances, kicks, listens, burps, farts, whistles, yells, snores, plays soccer, speaks international 'caveman', and does six types of karate chop. It is programmable up to eighty-four steps, uses three callable, programmable 'reflex' macros, has conditional sensors for touch and sound, and even features 'fast response' infrared programming up to twenty baud for wireless computer control. It uses just two chips to run seven motors and seven sensors over sixty built-in functions and three extended demonstration modes. Its 'brain' is an enhanced 8bit-2mHz sound controller common in many toys, and its 'peripheral nervous system' a custom motor driver based on Nv technology. It also features a twelve-hour battery life so there's plenty of time for experimentation.

The original prototype (built during an extended solder session at a Telluride Neuromorphic workshop many moons ago) used a twenty-four-transistor Nv analog controller, and is the only working survivor after dozens of 'knock-off' digital versions (most subsequent vendor prototypes had to drag their digital brains behind them on a cart). We can't use raw analog controllers in com-

mercial products because they're too variable to pass reliability tests. Fortunately, it's always easier to increase complexity from minimal solutions than the other way around. In the end we managed a hybrid digital-analog device allowing the robot good play range with only a minor loss in adaptability.

Originally the robot was to have sight, but we have yet to succeed in building an inexpensive digital vision system from the analog original (requires more translation from continuous to the discrete). However, the robot still has remnants of the Nv vision system, including object-tracking palm LEDs and a large empty chest for the pan-tilt motors. Also, though the design is single-screw-driver hackable—with color-coded wires and well-labelled electronics—the RoboSapien is less about education research and more about feature and function. Though scientists may love details, the average user just wants a cool gadget. Still, I hope someone might consider it a vehicle to carry around some cool VLSI. If you do, drop me a picture. Mutations are the only way this species can evolve.

So there you have a little detail behind my 'toy example': an inexpensive body into which I hope many might ply their Neuromorphic brains. It's cheap, shaky, inconsistent, stiff, and doesn't even have the brains of a pocket calculator.

Other than that though, it hardly resembles grad students at all.

Mark W. Tilden

Consultant to WowWee Toys
<http://www.wowwee.com>

References

1. Edward A. Rietman, E. A., Mark W. Tilden, M. W., Manor Askenazic, A., *Analog computation with rings of quasiperiodic oscillators: the microdynamics of cognition in living machines*, **Robotics and Autonomous Systems**, 45 (3-4), 31 December 2003, pp. 249-263.

Neuromorphic motion detection... from p. 8

creasing flight speed while retaining the reference OF. Hence, similarly to flying insects, the aircraft uses visual motion to control its altitude. Flight guidance using OF has also been demonstrated qualitatively on free-flying model aircraft.⁴

Thomas Netter and Nicolas Franceschini
CNRS Biorobotics

E-mail: tnetter@ini.unizh.ch and
franceschini@laps.univ-mrs.fr

References

1. T. Netter and N. Franceschini, *A robotic aircraft that follows terrain using a neuromorphic eye*, **Proc. IEEE/RSJ IROS'02**, pp. 129-134, 2002.
2. H. G. Krapp, B. Hengstenberg, and R. Hengstenberg, *Dendritic structure and receptive-field organization of optic flow processing interneurons in the fly*, **J. Neurophys** 79, pp. 1902-1917, 1998.
3. N. Franceschini, *Early processing of colour and motion in a mosaic visual system*, **Neurosci. Res. Sup. 2**, pp. 17-49, 1985.
4. G. L. Barrows, J. S. Chahl, and M. V. Srinivasan, *Biomimetic visual sensing and flight control*, **Proc. Bristol UAV Conf.**, 2002.

Figure 5. Obstacle avoidance: two laps over a 30° ramp.¹ Altitude and azimuth were monitored by encoders on arm axes.

