



## Short communication

## A FPGA real-time model of single and multiple visual cortex neurons

Guangxing Li, Vargha Talebi, Ahmad Yoonessi, Curtis L. Baker Jr. \*

McGill Vision Research Unit, Department of Ophthalmology, McGill University, Montreal, QC, Canada

## ARTICLE INFO

## Article history:

Received 1 March 2010

Received in revised form 23 June 2010

Accepted 26 July 2010

## Keywords:

Spiking model

Visual cortex

FPGA

Tetrode

Cortical neuron

Simple cell

## ABSTRACT

Using a biologically realistic model of a single neuron can be very beneficial for visual physiologists to test their electrophysiology setups, train students in the laboratory, or conduct classroom-teaching demonstrations. Here we present a Field Programmable Gate Array (FPGA)-based spiking model of visual cortex neurons, which has the ability to simulate three independent neurons and output analog spike waveform signals in four channels. To realistically simulate multi-electrode (tetrode) recordings, the independently generated spikes of each simulated neuron has a distinct waveform, and each channel outputs a differentially weighted sum of these waveforms. The model can be easily constructed from a small number of inexpensive commercially available parts, and is straightforward to operate. In response to sine wave grating stimuli, the neurons exhibit biologically realistic simple-cell-like response properties, including highly modulated Poisson spike trains, orientation selectivity, spatial/temporal frequency selectivity, and space-time receptive fields. Users can customize their model neurons by downloading modifications to the FPGA with varying parameter values, particularly desired features, or qualitatively different models of their own design. The source code and documentation are provided to enable users to modify or extend the model's functionality according to their individual needs.

© 2010 Elsevier B.V. All rights reserved.

## 1. Introduction

A realistically behaving physical model of a primary visual cortex neuron can be useful for classroom-teaching and for training students in the research laboratory. Such a model would also be beneficial to visual physiologists by helping to ensure smooth operation of experiments through proper evaluation of electrophysiological recording setups (including data acquisition, data analysis, and stimulus presentation). Incorporating the ability to simulate multi-electrode recordings of multiple neurons would further enhance the power of such a model.

Earlier neuron models that met some of these objectives were based on hardware, software, or custom integrated circuits. Most hardware-implemented models require numerous resistors, capacitors, and transistors (French and Stein, 1970) or integrated circuit chips (Rebrik and Miller, 2004; Schweitzertong, 1983) on a breadboard or homemade printed circuit board, making the device difficult to build, awkward to modify, and potentially unreliable. Software models such as the Visual Neuron Simulator (Peterson and Ohzawa, 1999; Saito et al., 2008) have been well-suited to classroom demonstrations, but did not provide electrical signals

for testing electrophysiological recording systems. Even with the addition of hardware extensions, a software simulator might still have difficulty providing real-time responses from complex models or multi-unit simulations.

The “Physiologist’s Friend” (Delbruck and Liu, 2004) is a technologically impressive model neuron for visual neurophysiologists, based on a custom-designed analog VLSI (Very-Large-Scale Integration) circuit. This model was designed as a biologically realistic simulation of early retinal processing, including automatic gain control to provide robustness for background lighting conditions. The output simulates spiking responses of retinal ganglion cells or visual cortex neurons to visual stimuli presented in its receptive field. However the use of a VLSI chip makes it impossible for the user to modify the model, and availability is limited due to the complex design process and special facilities required for fabrication.

To overcome the limitations of the models mentioned above and to incorporate new model features, for example multi-unit responses and multi-channel signal outputs, a higher performance simulator would be useful. It should be easily constructed from a small number of inexpensive commercially available parts, and be user-extensible without elaborate or specialized circuit fabrication.

A Field Programmable Gate Array (FPGA) addresses these issues by offering a high performance, configurable platform for building hardware model neurons. A FPGA is a programmable device consisting of a reconfigurable array of logic elements and a customizable hierarchy of interconnects that allows the logic elements to be tailored to a particular application and to be modified repeat-

\* Corresponding author at: McGill Vision Research, 687 Pine Ave W, H4-14, Montreal, Quebec, Canada H3A 1A1. Tel.: +1 514 934 1934x34819; fax: +1 514 843 1691.

E-mail address: [curtis.baker@mcgill.ca](mailto:curtis.baker@mcgill.ca) (C.L. Baker Jr.).

edly, somewhat like a one-chip programmable breadboard. The FPGA is reconfigured using hardware description languages (HDLs), which are not specific to a particular FPGA chip – thus code developed for one chip can be easily moved to another with only minor adjustments. Because of broad commercial development and support for FPGA technologies as well as the transportability of HDL code, this approach to neural modeling should be considerably cheaper and less vulnerable to obsolescence of hardware or operating systems. Finally, the inherent parallelism of the logic elements on a FPGA allows for a fully parallel approach to designing a multi-neuron model, providing the potential to run simulations in real-time and much faster than conventional software models. Previous FPGA neural models (Guerrero-Rivera et al., 2006; Pearson et al., 2007; Weinstein et al., 2007) demonstrated the power of this approach, but did not include visual inputs, simulations of visual cortex receptive fields, or outputs suitable for testing electrophysiology setups.

Here we present a FPGA-based model of one or more visual cortex neurons. The model is not intended to be biologically realistic in its internal computations, but only to simulate an “LNP” (Linear-Nonlinear-Poisson) model (Schwartz et al., 2006) of simple-type V1 cells’ responses to optically presented visual stimuli. We expect this simple model will be satisfactory for most users; however using the FPGA platform, users can extend response properties according to their individual needs by adding new code to the model definition, and/or hardware extensions. This model is constructed from an optical front-end (camera lens, filter, optical sensors) and a FPGA development board. It can simulate a tetrode (Gray et al., 1995) recording of up to three independent neurons on four channels of analog outputs. The simulated neurons produce spikes with distinct wave shapes and output differentially weighted sums of these waveforms on each channel. The model can be positioned to observe a physical stimulus screen and be directly connected to a multi-channel experimental recording setup. In response to sinewave grating stimuli, the model exhibits biologically realistic selectivity for orientation and spatial/temporal frequency, with a realistic simple-cell-like space-time receptive field structure. Since the analog output signals can be routed through the entire recording setup, stimulus presentation, data acquisition, and analysis can be easily tested prior to an animal experiment. This model is also suitable for training students on experimental protocols and setup usage, or demonstrating visual cortex neuron behavior in the classroom. The source code and necessary documentation of the

model are provided in the [Supplementary Materials](#). Using these resources, users will be able to build their own FPGA model from commercially available parts, with the option to modify or extend it according to their individual needs.

## 2. Methods

### 2.1. Hardware and FPGA programming

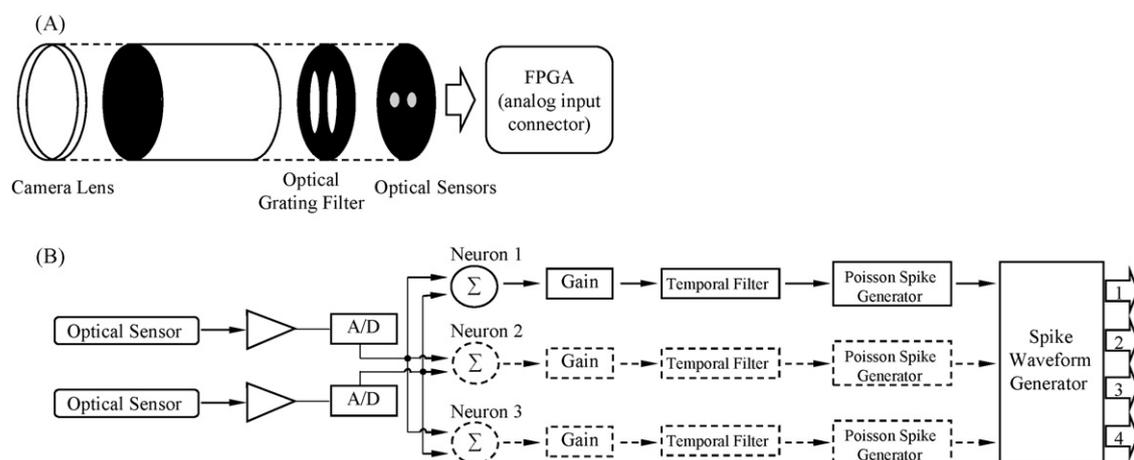
The model is composed of two main components: a FPGA development board and an optical front-end. The development board is a Xilinx Spartan-3AN Starter Kit (Xilinx, CA, USA), which includes a Xilinx Spartan-3AN FPGA (XC3S700AN-FG484) chip, a two-channel analog capture circuit, a four-channel analog output circuit, and a variety of connectivity and display options such as a LCD screen, LEDs, slide switches, RS-232 serial ports, and differential I/O connectors. The analog capture circuit consists of a dual-channel programmable scaling pre-amplifier and a 14-bit, 3Msps dual-channel analog-to-digital converter. The analog output circuit consists of a four-channel, 12-bit digital-to-analog converter.

The optical front-end (Fig. 1A) includes a 50 mm SLR camera lens, an optical grating filter (cardboard with two slits) to provide orientation and spatial frequency selectivity, and two optical sensors (T2L12S, TAOS, Texas, USA), one positioned behind each slit. Each optical sensor combines a photodiode and a transimpedance amplifier to provide an analog voltage output that is linearly proportional to input light intensity. All components of the optical front-end are enclosed in a matte black box to prevent stray light reaching the optical sensors.

The Xilinx ISE® WebPack™, free software that runs on a PC (Windows® or Linux), was used to create and compile hardware description code written in VHDL (Very-High-Speed Integrated Circuit Hardware Description Language). The compiled hardware description is then downloaded via a USB port to the FPGA, which can subsequently run “untethered” from the PC.

### 2.2. Architecture

A functional block diagram of the FPGA model is shown in Fig. 1B. The voltage outputs of the optical sensors are amplified and converted to digital signals by an analog capture circuit on the development board. The digitized sensor signals are differen-



**Fig. 1.** Architecture for the FPGA model. (A) The optical front-end. It includes a 50 mm SLR camera lens, an optical grating filter to produce spatial frequency and orientation selectivity, and two optical sensors positioned in the image plane of the lens. (B) Functional block diagram of the FPGA model. The model can run in either a one-neuron or a three-neuron mode. For each neuron, the FPGA uses a differential sum of the optical sensor signals to produce orientation and spatial frequency selective membrane potentials. The membrane potential is adjusted by a luminance gain controller to suit different average stimulus luminances, and digitally filtered to provide temporal selectivity. The filtered signal controls the rate parameter of a Poisson Spike Generator, which triggers a spike waveform generator to produce four-channel spike trains that are simulated to mimic signals from a tetrode recording. Dashed lines indicate parts relevant to the three-neuron mode.

tially summed to create a model ON–OFF simple cell receptive field and produce orientation and spatial frequency selective membrane potentials. Each membrane potential is adjusted by a gain controller to ensure robust operation for different average stimulus luminances, and then fed into a Butterworth–FIR (Finite Impulse Response) filter (designed with WinFilter Ver. 0.8) to produce low pass temporal frequency selectivity (cutoff 4 Hz).

Each filtered signal drives the rate parameter  $r[i]$  of a Poisson Spike Generator. For each spike generator, a sequence of random numbers  $x[i]$  is generated by a 16-bit linear feedback shift register (LFSR). As long as  $r[i]$  varies slowly in each sample  $i$  (sample interval = 1 ms), a spike is generated when  $x[i] \leq r[i]$ , and the spike does not fall in a refractory period (5 ms) since the preceding spike. A programmable switch on the development board specifies whether the model runs in three-neuron or one-neuron mode. In three-neuron mode, each neuron has its own independent sensor signal summation, FIR filter, and Poisson rate generator.

When a neuron fires, its spike waveform (from a pre-specified set of stored wave shape templates) is selected by the spike wave generator and sent to different channels on the DAC chip, where it is converted into an analog voltage signal. If the FPGA model is running in three-neuron mode, the independent neurons' spikes will have distinct wave shapes, and will appear at different amplitudes on each channel as in a tetrode recording (Fig. 2A and B). If the model is running in one-neuron mode, then one wave shape will appear with different amplitudes on the four channels (again, simulating a tetrode recording of only one neuron). The output signals can be connected directly to an audio monitor, a data acquisition system, and/or the head-stage of an extracellular amplifier. In the latter case, the spike amplitudes (maximum 3.3 V) may require attenuation appropriate to the gain of the recording system.

### 3. Results

To illustrate the model's capability, we recorded responses of the simulated neurons to drifting sinusoidal grating stimuli using a Plexon® data acquisition system (Recorder/16), and analyzed the data with in-house software. In the two channels of recorded raw data shown in Fig. 2A, it is apparent that there are spikes from multiple neurons in each trace, and the spike times in the different "recording sites" are highly correlated in a way that mimics neuronal spikes collected from a multi-electrode. In this manner, the FPGA model simulates the fact that neurons with different distances from two electrode tips will have different spike-amplitude ratios when recorded on different channels (McNaughton et al., 1983). By using spike-sorting methods (Offline Sorter, Plexon Inc.), three simulated neurons with different wave shapes can be clearly isolated (Fig. 2B).

Since each of the simulated neurons has its own differential summation, luminance gain, and temporal filter, the neurons can each exhibit different spontaneous activity, temporal frequency selectivity, and spatial receptive fields (relative strengths of ON- and OFF-zones). To demonstrate the model's response properties, we show results from one of the simulated neurons. The spikes from the simulated neuron fire randomly, with an average rate that is highly modulated by sinewave grating stimuli (Fig. 2C), as expected for a simple cell response. The simulated neuron also exhibits V1-like biologically realistic selectivity for orientation (Fig. 2C and D), spatial frequency (Fig. 2E), and temporal frequency (Fig. 2F). We also mapped the spatial receptive field of the simulated neuron by applying system identification methods (Wu et al., 2006) to its response to short bar stimuli (Fig. 2G). Specifically, we employed a gradient decent, iterative regression algorithm with regularization (Theunissen et al., 2001) to estimate the spatio-temporal receptive field of the FPGA neuron. Fig. 2G shows the spatial receptive

field estimate of the neuron at a temporal latency of approximately 26 ms. Like a real V1 simple cell, the receptive field has spatially segregated ON and OFF subfields.

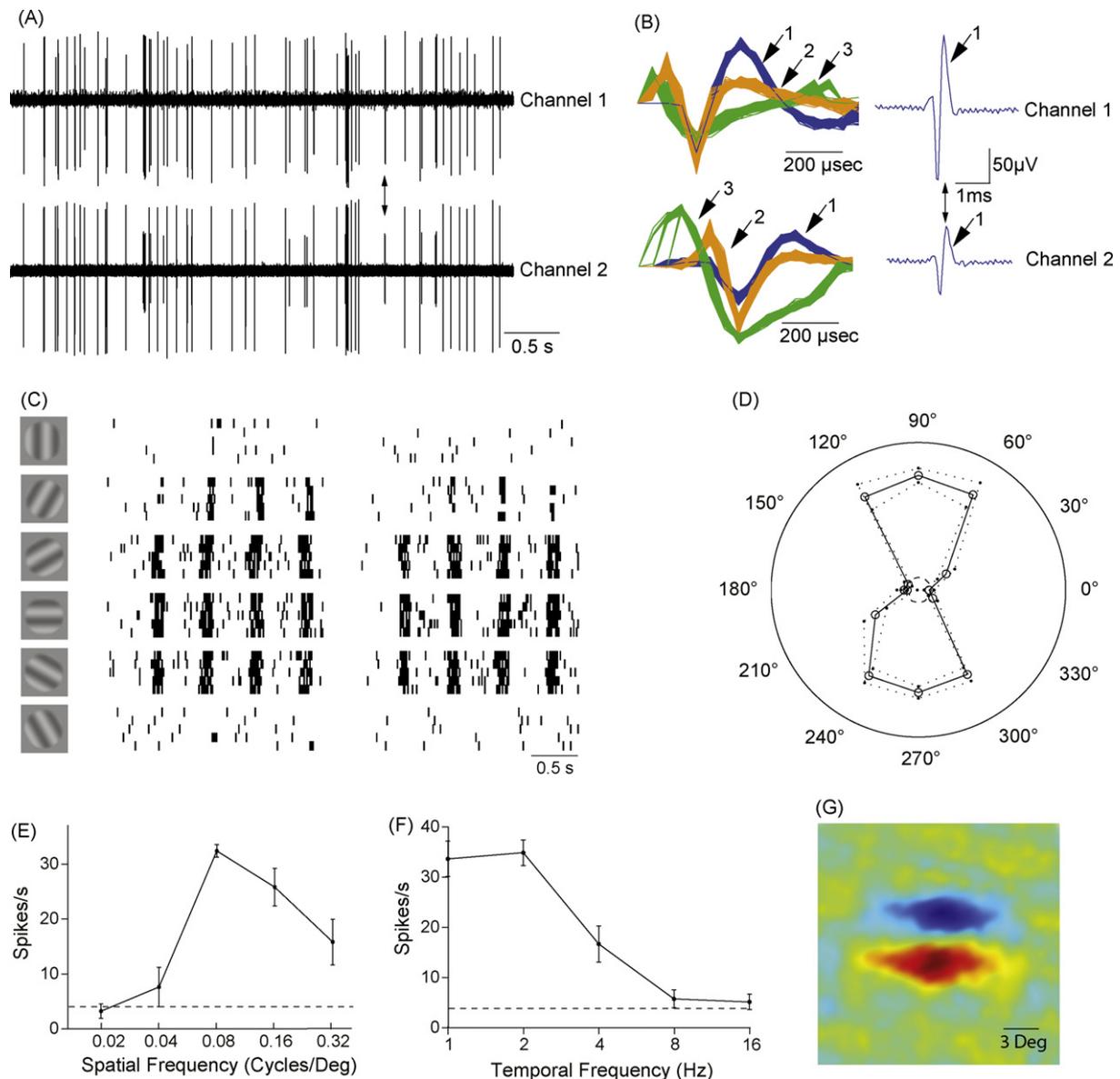
The FPGA model can also run in one-neuron mode, which simulates a single unit recording. With an attached audio monitor, the model produces irregular spontaneous activity when no stimulus was presented. It also exhibits biologically realistic orientation selectivity in response to a hand-held black bar against a bright background and gives modulated responses when using computer-produced drifting grating stimuli presented with an LCD projector in a small seminar room.

### 4. Discussion

The FPGA-based approach combines advantages of both hardware and software models. Like other hardware models, it is very easy to use, runs much faster than software models, and provides real-time spike waveform outputs suitable for testing electrophysiology setups. Like software models, the FPGA-based model can be easily reprogrammed and is fully customizable. The model described here is easy to build from commercially available parts. The only expensive component is the FPGA development board (~\$200 USD); the camera lens need not be of high quality and an older-generation used one is entirely adequate. The limitations of our FPGA model should also be mentioned here: first, the user must do some assembly and configuration, though this is not very difficult; second, as mentioned previously, our FPGA model is not a biophysically realistic simulator – instead it simulates an "LNP" model of V1 simple cells. Code modifications could be considered in the future to enhance the model's biological realism, for example: band-pass temporal frequency response, contrast gain control or adaptation, burst-mode spiking, or simulated local field potentials.

To modify the FPGA model code, users would need to employ a hardware description language (HDL), such as VHDL or Verilog. VHDL is one of the most commonly used HDLs, and is the programming language of choice for describing mixed analog/digital systems. VHDL is analogous to conventional programming languages, such as BASIC, C, or assembly code, and shares some similarities: it has a specifically written syntax and semantics that are used to build a standard text-based expression of the desired system behavior. However VHDL also has some major differences from conventional software programming languages. First, in contrast to most software programming languages that are inherently procedural (single-threaded), sequentially running one instruction at a time, VHDL can specify multiple parallel processes that simultaneously execute independently of one another. Second, unlike conventional software compilers, a VHDL compiler transforms the code into physically realizable "gates and wires" to be instantiated on the FPGA chip. In this sense, a VHDL compiler actually configures the hardware, whereas a software compiler simply converts source code into microprocessor-specific object code. Designing a system in VHDL is generally more difficult and time-consuming than writing a program in a software programming language.

The model described here employs only two optical sensors because our Xilinx board only provides two analog-to-digital inputs. To achieve greater flexibility for possible receptive fields, we have developed a version with a small array of several optical sensors placed at the image plane of the lens. This version requires an external custom-made printed circuit board with additional pre-amplifiers (LTC6912-1, Linear Tech, CA, USA) and A/D converters (LTC1407A-1, Linear Tech, CA, USA). It does not require the optical grating filter because the spatial frequency and orientation selectivity are programmed into the FPGA's summation of the optical sensors. Since each of the simulated neurons has its own differential summation of different combinations of the sensors, each of the



**Fig. 2.** Examples of recorded signals from the FPGA model, simulating three neurons. (A) Two channels of recorded raw data from the FPGA model outputs, simulating two channels of a tetrode recording of three neurons. There is more than one spike waveform in each channel, and the spikes between the two channels are highly correlated with one another. (B) Spike waveforms of each of three neurons in the two channels (the second channel is delayed about 150 μs from the first, due to different level-crossing thresholds used in the Plexon Offline Sorter). By using spike-sorting methods, the three simulated neurons in each channel are clearly separated. (C) Spike raster plots of one neuron for different orientations of a drifting sinusoidal grating (2 Hz, 0.08 cycles/°). The spike firing was highly modulated at the temporal frequency of the drifting sinusoidal grating stimuli, and the neuron exhibited orientation selectivity. (D) Orientation tuning curve shows selectivity for horizontal gratings (90° and 270°). (E and F) Spatial and temporal frequency tuning curves, showing selectivity for about 0.08 cycles/°, and temporal frequencies below about 4 Hz. (G) Spatial receptive field structure of the neuron, at a latency of ~26 ms. Like a real V1 simple cell, the receptive field has spatially segregated ON (white) and OFF (black) subfields (in the web version, red and blue, respectively). In (D–F), dashed lines indicate spontaneous firing rate; plotted symbols indicate mean ± sd. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of the article.)

three neurons can have a distinct orientation preference. Further VHDL code modifications could conceivably also provide direction selectivity or complex cell behavior. Simulation of neural responses to other sensory modalities (auditory, somatosensory, electrosensory) could also be provided by substitution of other front-end sensors.

Much more powerful models could be implemented by use of a video camera with a higher-end FPGA board, though this would entail additional VHDL programming. Such an imaging input would enable simulation of large numbers of spatially distinct receptive fields, and/or more complex properties such as surround modulation (e.g., Tanaka and Ohzawa, 2009) or second-order responses (e.g., Song and Baker, 2007).

## Acknowledgements

A brief report of this work has been published in abstract form (Li et al., 2009). This research was supported by a Canadian Institutes of Health Research grant MA-9685 and a Natural Sciences and Engineering Research Council (NSERC) grant OGP0001978 to C.B., and an NSERC fellowship D2-348949-2007 to V.T.

## Appendix A. Supplementary data

Supplementary data associated with this article can be found, in the online version, at doi:10.1016/j.jneumeth.2010.07.031.

## References

- Delbruck T, Liu SC. A silicon early visual system as a model animal. *Vision Res* 2004;44:2083–9.
- French AS, Stein RB. A flexible neural analog using integrated circuits. *IEEE Trans Bio-Med Eng* 1970;Bm17:248–53.
- Gray CM, Maldonado PE, Wilson M, McNaughton B. Tetrodes markedly improve the reliability and yield of multiple single-unit isolation from multi-unit recordings in cat striate cortex. *J Neurosci Methods* 1995;63:43–54.
- Guerrero-Rivera R, Morrison A, Diesmann M, Pearce TC. Programmable logic construction kits for hyper-real-time neuronal modeling. *Neural Comput* 2006;18:2651–79.
- Li G, Yoonessi A, Talebi V, Baker CL. A real-time hardware spiking model of single visual cortex neurons. Program No. 353.4. 2009 Neuroscience Meeting Planner. Chicago, IL: Society for Neuroscience; 2009 [Online].
- McNaughton BL, O'Keefe J, Barnes CA. The stereotrode: a new technique for simultaneous isolation of several single units in the central nervous system from multiple unit records. *J Neurosci Methods* 1983;8:391–7.
- Peterson M, Ohzawa I. VNS (Visual Neuron Simulator). 1999-08-05. <http://neurovision.berkeley.edu/Demonstrations/VSOC/vsoc/vns/index.html>.
- Pearson MJ, Pipe AG, Mitchinson B, Gurney K, Melhuish C, Gilhespy I, et al. Implementing spiking neural networks for real-time signal-processing and control applications: a model-validated FPGA approach. *IEEE Trans Neural Network* 2007;18:1472–87.
- Rebrik S, Miller K. V1 Simple Cell Emulator. 2004-2-3. <http://millerlab.ucsf.edu/Hardware/V1SimpleCellEmulator/default.htm>.
- Saito Y, Sugino H, Ohzawa I. Real-time Visual Neuron Simulator. 2008-1-18. [http://visiome.neuroinf.jp/modules/xoonips/detail.php?item\\_id=3308](http://visiome.neuroinf.jp/modules/xoonips/detail.php?item_id=3308).
- Schwartz O, Pillow JW, Rust NC, Simoncelli EP. Spike-triggered neural characterization. *J Vision* 2006;6:484–507.
- Schweitzer DE. The photoneuromime – an artificial visual neuron for dynamic testing of computer-controlled experiments. *Behav Res Methods Instrum* 1983;15:9–12.
- Song YN, Baker CL. Neuronal response to texture- and contrast-defined boundaries in early visual cortex. *Visual Neurosci* 2007;24:65–77.
- Tanaka H, Ohzawa I. Surround suppression of V1 neurons mediates orientation-based representation of high-order visual features. *J Neurophysiol* 2009;101:1444–62.
- Theunissen FE, David SV, Singh NC, Hsu A, Vinje WE, Gallant JL. Estimating spatio-temporal receptive fields of auditory and visual neurons from their responses to natural stimuli. *Network Comput Neural* 2001;12:289–316.
- Weinstein RK, Reid MS, Lee RH. Methodology and design flow for assisted neural-model implementations in FPGAs. *IEEE Trans Neural Syst Rehabil Eng* 2007;15:83–93.
- Wu MCK, David SV, Gallant JL. Complete functional characterization of sensory neurons by system identification. *Annu Rev Neurosci* 2006;29:477–505.