MANCHESTER 1824

SpiNNaker-based Visual Systems

End-of-first-year report

Garibaldi Pineda García

Supervisor: Steve Furber
Co-supervisor: Dave Lester

Advanced Processing Technologies Group
School of Computer Science
University of Manchester
United Kingdom

University of Manchester
APT Group

Contents

Abstract

3d reconstruction
slam, current approaches too expensive
neural representations can reduce computations
study of the brain and state-of-the-art in neural and classic vision
first years work is input

Acknowledgements

CONACYT/SEP

CHAPTER 1

Introduction

Vision is probably the greatest sensory input the brain has, it allows us to perceive a vastly diverse set of phenomenons. From enjoying a baby smile to check if our meals are well cooked to avoiding traffic accidents, it all happens to the amazing visual processing capabilities of the brain. Furthermore, this kind of input has given humans the possibility of culture through reading and writing, to the point that we are now starting to unravel the secret of how the brain works.

What is known as vision can be thought of as a set of tasks that have been observed in humans and other animals, such as object detection and recognition, image segmentation or depth perception, to name a few. A complex but crucial aspect of vision is the ability to create mental maps of our current or, even past, locations. This may have been developed as a survival strategy that allows animals to reach known food or water resources. Creating a system that mimics this extraordinary capacity is still an open research problem. Since it's been observed that our nervous system is able to recreate environments, taking inspiration from nature is a good strategy to follow. The goal of this project is to create a bio-inspired 3D environment reconstruction system based on spiking neural networks. An additional benefit of this research is to acquire further knowledge of how the brain gains understanding and interprets the world through vision. Some applications to such a system are the mapping hazardous zones (e.g. nuclear power plants, war zones, active volcanoes), security (e.g. traffic camera analysis) or self or assisted guided vehicles (e.g. cars, drones, aeroplanes).

1.1 Problem description

Environment reconstruction is an active field of research, particularly from the Simultaneous Localization and Mapping (SLAM) community[1]. **SMALL DESCRIPTION OF SLAM!!!**

Humans are able to do something similar with an efficient highly-parallel neural computing system that requires about 20-watts to function. How exactly this is done in the brain is still an

open question. This research will provide a solution, inspired by state-of-the-art neuroscience, to the environment reconstruction problem using neuromorphic hardware.

Most research has cameras and a mixture of exotic depth sensors as inputs. One implication of this type of input is large quantities of information having to be processed, thus, needing high-performance and power-hungry devices to execute their algorithms. This is something that limits the actual utility of such systems for mobile applications. On the other hand, neuromorphic sensors have shown to reduce representations so that irrelevant information is not transmitted nor processed [2, 3].

Another disadvantage to using "classic" computer vision approaches is that computational resources would need to be shared inefficiently (i.e. a processor would have to switch between a facial recognition algorithm to a depth-estimation one). Having a neuro inspired system means that the tasks are executed by the same network.

SpiNNaker provides a massively-parallel high-efficiency computing platform, inspired by the brain. It's an excellent choice for neuroscience research, particularly to study spiking neural networks. Its software stack has many ready-to-use neuron models and development can be performed in a straight forward manner [4].

1.2 Objectives

The principal objective of this research is to develop a system that performs 3D environment reconstruction using spiking neural networks. In order to achieve this goal, some milestones will have to be reached.

The first step is to perform *image classification*; experiments have shown this can be done in about 150ms in the brain [5]. This is important for real-time systems to enable certain known objects to act as markers in a 3D environment. In order to achieve such classification speed, *spike-time coding*, a type of neural encoding, has been suggested as the best match due to its information representation capabilities [6]. Creating a procedure that allows spiking neural networks learn its weights using spike-time coding is still an open research question and quite unexplored territory. Hierarchical networks have proven to be a robust way to recognize images [7, 8], thus developing such a network seems like the most reasonable path.

Given that different views of objects in the real world are correlated in space and time, spiking neural networks should make an excellent match for *3D object recognition*. This is the second milestone for this research. It would allow the environment reconstruction system to keep track of objects regardless of their position, facilitating the localization part of the system.

The third milestone is a way to establish the distance of objects to the observer or *depth perception*. This could be done using binocular vision (either using two cameras or inferring the 3D transformation of the camera from optic flow), depth-from-defocus, or including other sensors, perhaps. **CITE!!!**

The *localization and mapping* problem has been proven to be easier to solve if taken simultaneously [9, 10]. From a neural networks point of view, the probabilistic models are stored in the network itself. Inspiration from rat hippocampus studies on location awareness have lead to neural network approaches [11].

The final step of this research is to *reconstruct an environment* from the knowledge stored in the neural network. This needs a top down approach, which is most commonly done by analysing network weights (sometimes using other additional neural networks) and, from that, infer what the original input values were [12].

In summary, the project consists of employing spiking neural networks and spike-time encoding to perform:

1. Image recognition.
2. 3D object recognition.
3. Depth perception.
4. Orientation and localization.
5. Environment reconstruction.

1.3 Report structure

As the inspiration for this work will be the properties of the brain, a description of the brain and its function can be found in Chapter 2 and we delve into the components of human vision in Chapter 3.

Neuromorphic hardware is a new trend in electronics and computer hardware design which takes inspiration from the brain, some examples of such platforms are explored in Chapter 4.

An overview of the current state-of-the-art in 3D environment reconstruction is presented in Chapter 5.

Input for spiking neural networks has to be in *spike trains*, which are a series of pulses emitted or received by a neuron in a given time slot. In order to use regular video sources they need be converted. There are few solutions which, mostly, require the use of custom hardware which is expensive may not be available to everyone. This years work consisted on creating a software-based encoding procedure using parallel programming on off-the-shelf hardware; details of this can be found in Chapter 6.

Conclusions and further work plans are presented in Chapte r7.

CHAPTER 2

A [very] brief look into the brain

Nervous systems in animals are different, from the simple ones found in insects to more complex ones in reptiles, birds and mammals. They are all composed mainly of a special kind of cell, the *neuron*, that excels at long distance communication. Most animal's nervous systems include an organ called the *brain* which plays a central role on the everyday life of the animal. In most cases the brain will be located in the head which is the closest location to the primary sensing organs like eyes, tongue or ears [13].

The **human brain** is an exquisite piece of evidence of energy-efficient biological computation; the result of millions of years of an evolutionary process. It's been subject of multiple studies and, yet, we are barely getting to know it. The human brain consists of around 10^{12} individual cells called *neurons* which are interconnected through about 10^{15} special structures known as *synapses*. Studies have found that the brain is formed by many components, a brief description of the principal parts (shown in Figure 2.1) is presented next [14].

Cerebrum , or *cortex* is a wrinkled sheet of neurons that's the largest portion of the brain and is responsible of high level cognition, motor control, memory and problem solving, to name a few.

Cerebellum , also known as the "small brain" is mostly involved in sensorimotor tasks like balance or the coordination of the body.

Brain stem , which is located under the cerebrum and in front of the cerebellum, it connects the brain to the spinal cord and is in charge of automatic functions like breathing or digestion.

Limbic System , had been thought of as the "old brain" sits between the brain stem and the cortex. Some of its functions include hormonal control, emotions, learning and memory consolidation.

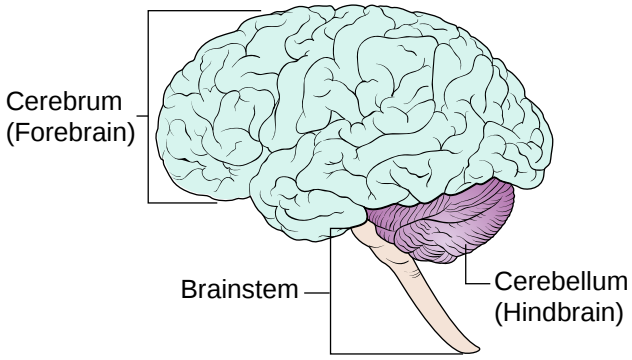


Figure 2.1: Main areas of the brain [15].

About 80% of the neurons in the brain are located inside **the cortex**. This thin sheet of about 1100 cm^2 area and a 2 to 4 mm. thickness is responsible for the high-level cognitive tasks humans can perform [14]. It makes us capable of the most diverse behaviours, from reading this report to cooking or running a marathon, the cortex is the motor behind our thoughts.

The fact that the cortex is thoroughly wrinkled allows a larger area sheet, and neurons, to fit in the same volume. It is composed of two symmetric shapes, the left and right *hemispheres* (left of Figure 2.2). Although both share functions, it has been observed that one hemisphere may dominate the other on some tasks [16].

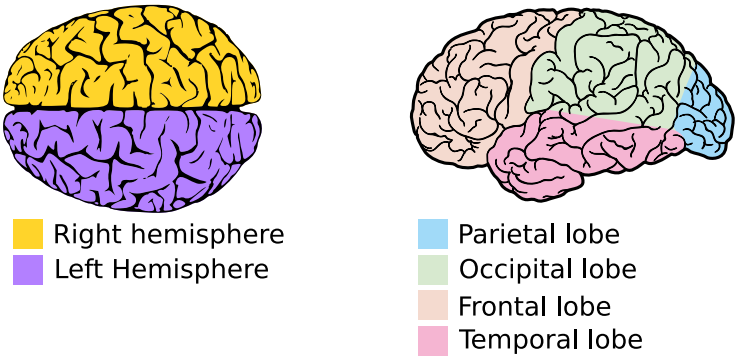


Figure 2.2: Left: The left and right brain hemispheres. Right: The brain divided into lobes.

Anatomists have divided the cortex into regions, known as *lobes*, separated by large creases (right of Figure 2.2). New classification of areas are constantly being discovered, whose class ranges from the functional to the physiological [17]. Some areas that most scientists agree on are the somatosensory cortex, that deals with touch; and the striate cortex, which plays a role in visual perception (more on this in Chapter 3).

2.1 Neurons and responses

Early neuroanatomists had different ideas of how the brain was composed, some stated that it was a continuous organ, that is neurons were fused into a single mesh of cells with no separation. In the early 1900, Ramón y Cajal demonstrated that nervous system consisted of individual cells and that they communicated through different parts of their anatomy [18]. Figure 2.3 shows one of his many drawings, it portraits neurons located near skin and muscular tissue.

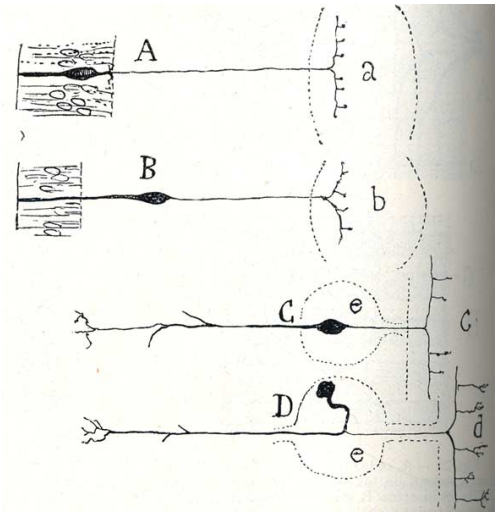


Figure 2.3: One of many Ramón y Cajal's drawings [19]

These cells are called *neurons* and they are particularly good at long-range communication [14]. While most cells in the body can "talk" to their neighbours, neurons have structures that allow them to communicate for up to a meter [17]. The flow of information through the billions of neurons in the brain is what originates behaviour, this is an amazing fact that has led many scientists to focus their minds into this phenomenon.

Basic neural anatomy

Neurons are composed of a *soma* or body, this part has similar components to other cells in the body; and specialized communication structures called the *axon* and *dendrites*. The **neuron soma** has many similar elements to other cells in the human body (e.g. nucleus, mitochondria, Golgi apparatus). All the neuron is enclosed in the cell membrane, just like any other cell. The soma is the biggest shape at the left of Figure 2.4. **Dendrites** are ramifications that come off the soma and, in most cases, are used as receivers of information from other neurons (red branch-like structures at the left of Figure 2.4). The shape of dendrites allows the nerve cell to have a larger contact area for other cells to communicate with. For some neurons, dendritic branches have "spines", these are synapses formed with another neuron.

The other specialized communication element of nerve cells is the **axon**, it's at the other end of the information exchange and can be seen as an output of the neuron. It usually has elongated tubular shape and ramifications at its end, shown at the right of Figure 2.4. As with dendrites, the tree-like shape at the end of the axon permits larger contact area, thus, more neurons can interconnect. The middle portion of the axon may, sometimes, be covered by a thin layer *myelin* (blue covering on the axon in Figure 2.4); a substance that acts as an insulator and improves signal transmission [14]. Although this is the basic way nerve cells communicate, there are variations that have been recently discovered and are subject of the late research efforts [20].

The place where two cells are in proximity and information is exchanged is called the **synapse** (detail circle on top of Figure 2.4). The cell sending information is labelled, *pre-synaptic*; and the one receiving, *post-synaptic*. The space between the pre- and post- synaptic cells is called the *synaptic cleft* and it measures about 20 nm.

Synapses

Communication between neurons can be done through two types of processes, electrical and chemical, the latter being the most common in mammalian brains. In **chemical synapses**, the pre-synaptic neuron sends neurotransmitter molecules through pipes (*microtubules*) that

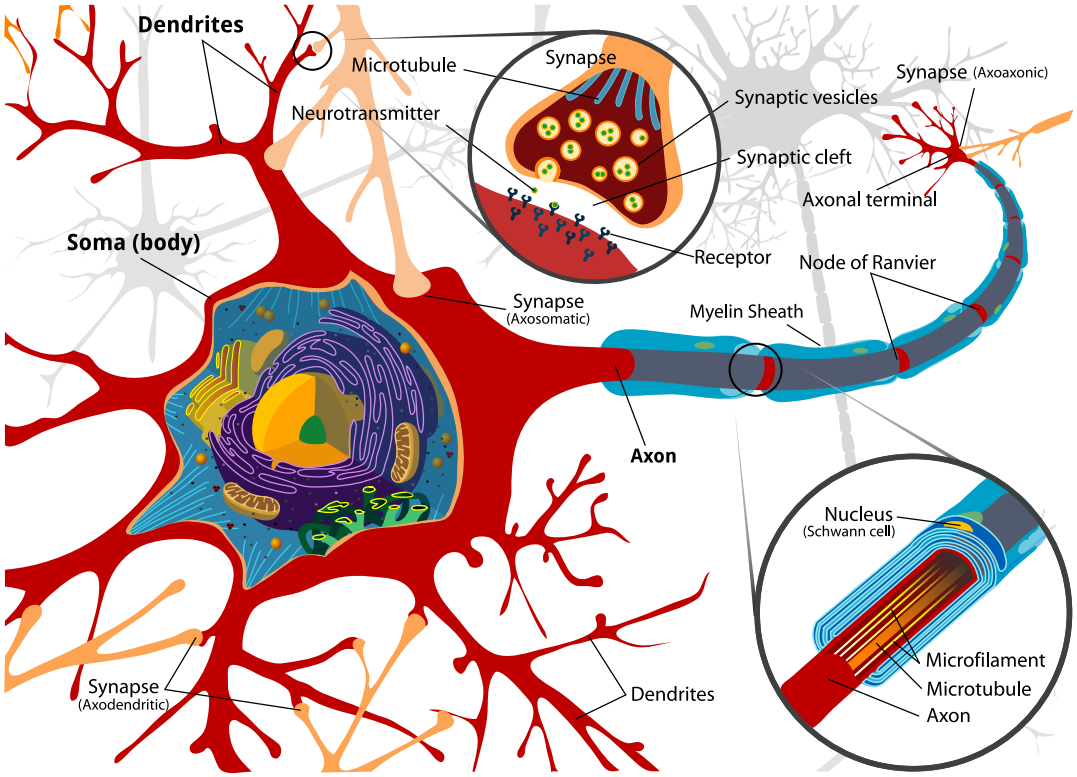


Figure 2.4: Principal components of a neuron [15].

run inside its axon. At the end of the pre-synaptic axon, these transmitters are stored in vesicles. When a synapse is active, that is it's exchanging information, vesicles merge with the cell membrane and release the transmitters it stored.

When the pre-synaptic cell releases neurotransmitters, these will spread across the synaptic cleft. On the receiver end, the post-synaptic cell membrane is covered by chemical receptors of different types. After the release of transmitters from the pre-synaptic cell, chemical receptors in the post-synaptic neuron will receive them if they are of the right type. This exchange of chemicals will create a reaction on the receiving cell. One of the major differences between chemical and electrical synapses is that, the former ones are plastic, they can be changed throughout its formation to amplify or diminish its activity.

Electrical synapses are also known as *gap junctions* and are thought to be rigid, it takes large changes in the cell to alter the synapse. This limits their information processing functionality because the way they are generated almost never changes. Gap junctions play an important role in most cells in the body, for example, they allow muscle cells to coordinate and create movement; researchers have found that they might also be critical for the development of the cortex [14, 21].

The neuron membrane has an electrical potential with respect to the exterior fluid, it goes from -90 to +50 mV while resting. Chemical and electrical synapses alter this potential, pushing it towards a positive (excited, depolarized) or negative (inhibited, hyperpolarized) state. Figure 2.5 shows the behaviour of the *post-synaptic potential* (**PSP**) to the two types of inputs (excitatory first, then inhibitory).

If the input is excitatory, the PSP will rise and have a better chance of generating an *action potential* and its known as an *excitatory post-synaptic potential* (**EPSP**). When the input is of the inhibitory kind, it will hyperpolarize the post-synaptic membrane potential; that is, it will diminish the probability of an action potential being generated. This type of response is known as *inhibitory post-synaptic potential* (**IPSP**).

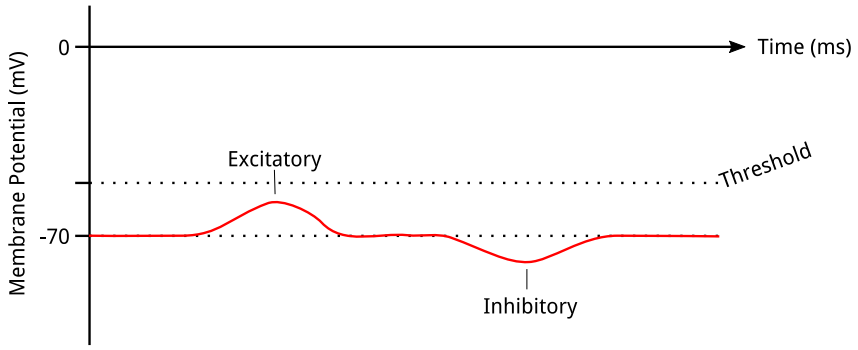


Figure 2.5: Post-synaptic membrane potential response to excitatory (left) and inhibitory (right) inputs.

Whenever the membrane potential goes over a certain threshold (usually due to the sum of multiple EPSP), the neuron will generate an **action potential** also known as a **spike**. Figure 2.6 shows an approximate graph of an action potential, each phase in the diagram is due to a change in ion concentration in the neuron (for a detailed explanation of this, see [14]).

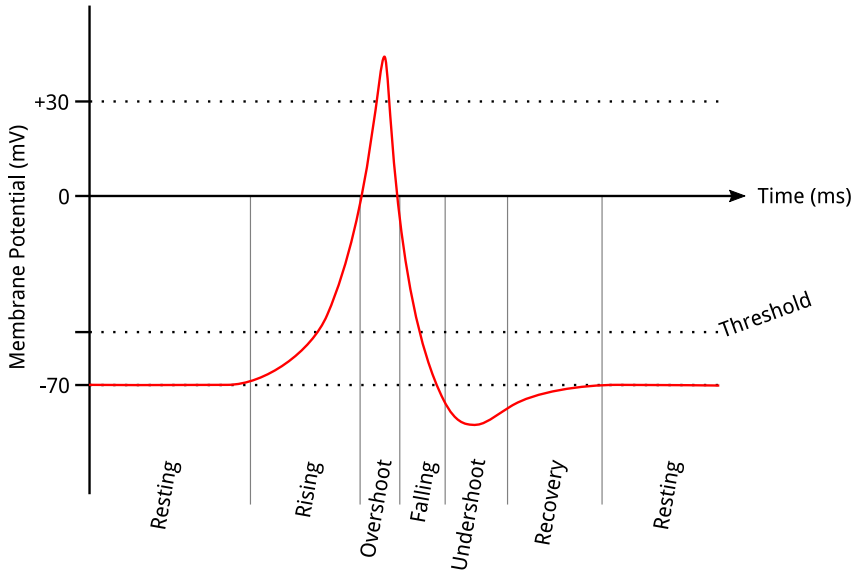


Figure 2.6: The action potential (spike) phases, each stage is associated with ion concentration changes in the neuron.

A spike is considered an ON or OFF response from neurons and is thought to be the normal neural communication mechanism. Some neurons use analog/continuous signals to transmit information, though they are mostly found near sensory organs.

In summary, neurons receive information, in the form of action potentials, mostly through their dendrites. Nerve cells send information to other neurons through their axon; the space between axon and dendrite terminals is called the synapse. If action potentials increase the probability of the receiving neuron to generate a spike, then the potential came through an excitatory input. On the other hand, if it decreases the probability of spike generation, the potential came through an inhibitory synapse.

Neuron models

Mathematical models for the membrane potential behaviour started appearing in the early 1900 CE. They range from the extremely detailed ones that consist of several differential equations to simple ones with just one or two and they all model the electrical properties of the nerve cells.

A particular group of models (self-compartment) describes the neuron as an isopotential sphere, that is, all its surface has the same electrical potential (Figure 2.7) [22].

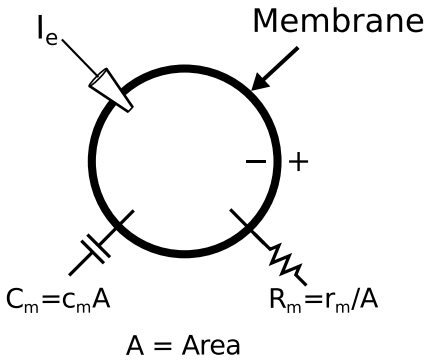


Figure 2.7: Diagram of the isopotential neuron. Adapted from *Theoretical Neuroscience* by Dayan and Abbott [22].

Since the neuron is modelled as a sphere, the membrane *capacitance* C_m and *resistance* R_m are specified in relation to its area A.

$$R_m = r_m/A \tag{2.1}$$
$$C_m = c_m A \tag{2.2}$$

where r_m and c_m are the resistance and capacitance per unit area, respectively. Their values are $r_m \approx 1M\Omega mm^2$ and $c_m \approx 10nF/mm^2$. The basic relations of the electrical properties of the membrane are shown in Fig. 2.7, are as follows:

$$\Delta V = I_e R_m \tag{2.3}$$
$$Q = C_m V \tag{2.4}$$
$$C_m \frac{dV}{dt} = \frac{dQ}{dt} \tag{2.5}$$

When the membrane's potential changes, it does so according to Eq. 2.3. Variable Q is the membrane's electrical charge which is proportional to its voltage and capacitance, as seen in Eq. 2.4. Currents originated by ion channels are thought to be linear, and can be modelled using Ohm's law

$$i_x = g_x(V - E_x) \tag{2.6}$$

where E_x is the *reverse potential* due to ion exchange in channel x and g_x is the per unit area conductance of the channel. The total membrane current due to channels (per unit area) will be

$$i_m = \sum_x i_x = \sum_x g_x(V - E_x) \tag{2.7}$$

For the total membrane current due to ion channels Eq. 2.7 has to be multiplied by total area A. The right hand side of Eq. 2.5 is the total current in the membrane. Since we are adding an external current I_e and is of opposite direction to I_m, the total current in the membrane is

$$I_T = I_e - I_m \tag{2.8}$$

Combining equations 2.5 and 2.8 gives the basic equation used by most self-compartment models [22].

$$C_m \frac{dV}{dt} = I_e - I_m \tag{2.9}$$

$$c_m \frac{dV}{dt} = \frac{I_e}{A} - i_m \tag{2.10}$$

Leaky Integrate-and-fire

This is one of the oldest neuron models, but it's still being used due to its simplicity. In the passive *integrate-and-fire* all the membrane conductances are modelled by a single term G_L (Equation 2.11).

$$C_m \frac{dV}{dt} = I_e - G_L \left(V - E_L\right) \tag{2.11}$$

the rightmost term is also known as leak current. If Eq. 2.11 is multiplied by the membrane resistance (R_m), we obtain

$$\tau_m \frac{dV}{dt} = R_m I_e - V + E_L \tag{2.12}$$

Integrating Eq. 2.12 results in an expression for the voltage behaviour under non-spiking conditions.

$$V\left(t\right) = E_L + R_m I_e + \left(V\left(0\right) - E_L - R_m I_e\right) e^{-t/\tau_m} \tag{2.13}$$

Spiking behaviour is added artificially once $V(t)$ reaches a certain threshold, afterwards it's reset back to $V(0)$.

Hodgkin-Huxley

In 1952, Hodgkin and Huxley published a paper that reflected their ground-breaking experimental work on the axon of the squid [23]. They found that the currents in the membrane are mainly due to changes in the concentration of three ions: potassium (K^+), sodium (Na^+) and chlorine (Cl^-). The first two are related to voltage dependant conductance and the last to a leak current. Under these considerations, Eq. 2.9 becomes [24]:

$$C_m \frac{dV}{dt} = I_e - g_K n^4 (V - E_K) - g_{Na} m^3 h (V - E_{Na}) - g_L (V - E_L) \tag{2.14}$$

the variables n, m and h have the following behaviour

$$\frac{dn}{dt} = \alpha_n(V)(1 - n) - \beta_n(V)n \tag{2.15}$$

$$\frac{dn}{dt} = \alpha_m(V)(1 - m) - \beta_m(V)m \tag{2.16}$$

$$\frac{dn}{dt} = \alpha_h(V)(1 - h) - \beta_h(V)h \tag{2.17}$$

The Hodgkin-Huxley model is one of the most detailed neural models so far, it's served as inspiration and foundation of many studies. The main issue with this level of detail is that it comes at a price, it's computationally expensive; so for system simulating a big number of neurons the hardware has to be equally powerful.

Simple model

The dynamics of the Hodgkin-Huxley were studied using bifurcation diagrams and approximated by FitzHugh [25]. Using similar ideas and techniques, Izhikevich developed what he named the *simple model* of spiking neurons [26]. This model emulates the dynamics of membrane voltage on the sub-threshold area, for modelling the spike behaviour comes at the cost of tiny time-steps that increase the computational cost. Izhikevich's simple model consists of a pair of equations:

$$\frac{dv}{dt} = 0.04v^2 + 5v + 140 - u - I \tag{2.18}$$

$$\frac{du}{dt} = a(bv - u) \tag{2.19}$$

where v represents membrane voltage and u a negative feedback to v. The rising part of spiking behaviour is produced by the equations, though an artificial voltage reset is needed afterwards. When variable v reaches 30 mV or more, variables v and u are set as follows

$$v = c \qquad u = u + d \tag{2.20}$$

Parameters a, b, c and d are dimensionless and time has a ms resolution [24].

The simple model has, at least, two advantages: first, many observed neural behaviours can be replicated by modifying the model's parameters; and second, it keeps biological plausibility while having low computational cost [27]. This model seems to be the right candidate for large-scale, biologically-plausible and energy-efficient systems.

2.2 Different languages

Neurons communicate using different "languages", spike-codes

Spike rate

How many spikes where produced in a time slot. Not much information can be encoded this way. Easy to transfer previous neural net work. Not entirely biologically plausible, specially for high cognitive tasks.

Spike timing

The precise time a spike was emitted. Lots of information, but still difficult to use. Polychronization might be the answer to learning/training.

Rank-order

Only the order of spike events are important, not the particular timing. Might be more robust than spike-timing but can encode less information. Some problems on training as well.

Input from sensors is most likely rate-based, though processing time and energy consumption in the brain suggests a different one is used for further processing

2.3 Artificial neural networks

First modelled as an on-off threshold gate, perceptron

Multi-layered networks and feedback, Hebbian learning

Spiking neural networks, third gen, include time as a factor, more accurate, more powerful mathematical properties,

learning studied using Hebbian back-prop, stdp, bcm

still work to be done on time-based learning

2.4 Conclusions

conclusions the brain

CHAPTER 3

Vision

Vision is one of the most important senses for animals; humans use it extensively for all kinds of tasks. Hunting, assessing danger, reading, driving, drawing, predicting rain from grey clouds, etc., these are all tasks that involve *seeing*.

There is a vast collection of knowledge about the components of vision (primates in particular), though a unification (or the answer to *How do we see?*) has not yet been achieved.

Vision starts at the eye, which transforms electromagnetic radiation that assembles an image, into voltage pulses that our brain may interpret. This encoded images are sent to the posterior region of the brain through the optic nerves. The cortex then performs many computations that result in our ability to see.

3.1 The eye and the retina

Our everyday experience might lead us to believe that the eyes are sensory organs developed completely separate from the brain but, in fact, the retina is an extension of the brain that performs spatio-temporal compression of a continuous flow of "images" of the world.

The eye is composed of many parts that resemble a camera (LENS, CAMERA OSCURA, FILM)

After light has been transformed into an electrical representation, the retina takes over and computes a representation of it.

Photoreceptors have the task of transforming light into an electrical signal. Colour is perceived by special type of receptors *cones*. For low-light conditions and higher contrast sensitivity, we use *rods*. Vertebrates have both rods and cones. Evolutionary adaptation has made eyes in different animals have special ratios of cones and rods. Reptiles and fish have more cones, most likely because they "live" on daytime for a lack of worm blood.

Many mammals have retinas with more rods than cones. For primates the retina has has two almost dual sensor zones. Most of the photosensitive area has more rods than cones; a tiny

region called the *foveal pit* has almost no rods, is densely packed with cones for high-resolution vision and is virtually blind when there is not enough light.

Horizontal cells average spatially (surround), input from photoreceptors; output to bipolar and to photoreceptors (adapt to different light conditions)

Bipolar cells, centre behaviour, input from horizontal and photoreceptors

IMAGE OF CENTRE SURROUND!!!!

First layers (photoreceptors, bipolar, horizontal cells) use analog signals, ganglion cells use spike trains.

Most authors agree that ganglion cells can be modelled by a *Difference of Gaussians* due to its centre-surround behaviour.

Ganglion cells extend to the Lateral Geniculate Nucleus, where information is relayed and organized so that the cortex can interpret it.

Organization makes left visual field sent to right hemisphere, right field to left hemisphere.

Redundancy keeps things working even if some neurons/receptors die out. To avoid saturation of nerve fibres and over-representation lateral inhibition might play a big role. It's specially useful for spike-timing encoding, since sensors give a rate based output that needs to be re-encoded.

3.2 The visual cortex

The portion of the cortex that is involved with visual processing has been estimated to about 30%.

It has been studied and areas have been labelled due to their function.

V1, V2, V...

3.3 Conclusions

CHAPTER 4

Neuromorphic hardware

4.1 Classical computing

classical computing

4.2 Neuromorphic trends

neuromorphic hardware trends

4.3 SpiNNaker

spinnaker info

4.4 Event-based model

event-based programming/infrastructure

4.5 Conclusions

conclusions neuro hardware

CHAPTER 5

3D environment reconstruction

Environment reconstruction has been receiving a lot of attention from research community, specially with things like Simultaneous Localization and Mapping (SLAM). Typically performed using cameras

5.1 Simultaneous localization and mapping

Examples of SLAM
- High performance hardware and/or exotic sensors (laser/kinect-like)
- Mix of Kinect + DVS
- Rat neuro based SLAM

5.2 Visual cortex models

Lowe's work inspired by neuro
- Hierarchical has been shown to provide geometric transformation invariance
- Hierarchical neural networks for image interpretation
- Hierarchical temporal memory

5.3 Conclusions

CHAPTER 6

From video to spikes

Spiking neural networks require inputs encoded as spike trains. The most common way to do this is to perform a continuous value to frequency transformation using Poisson sources. For most of sensory input in the body, this might be good enough but the retina performs spatio-temporal compression before feeding any information into the cortex.

There are some video-to-spike encoders but have some issues. Real-time encoders require custom hardware and are hard to come by. For off-line encoding, applications are limited to certain type of research, that is no real-time experiments could be performed. Our objective this year was to generate a real-time video-to-spike encoder using of-the-shelf components.

Of special interest are mobile applications, if we can provide a low-power solution to a silicon retina emulator, we could enable millions of phones, tablets or computers to work as an input to neural computations (QUALCOMM CHIP, SPINNAKER) and keep the traditional camera functionality.

6.1 Real-time encoding

For mobile and robotics applications a real-time encoder is needed. Hardware based real-time video encoders are expensive and not massively produced, thus their availability is limited. Creating one with off-the-shelf components opens the potential users. Two different models chosen whose computational cost was low enough to keep them operating at real-time and had kept biological plausibility constraints.

General purpose computing in the graphics processor unit

GPU History GPU programming OpenCL Memory hierarchy

The foveal pit model

The highest resolution area of the eye is the foveal pit (see section 3.1). A functional model for this region of the retina was developed by Sen & Furber, they called the implementation the *Filter-overlap Correction algorithm* (FoCal)[28]. It's based on the response and physiology of the fovea. The authors concluded that using four different layers of ganglion cells, most of the visually relevant information could be recovered after encoding. Furthermore, the encoder outputs a collection of rank-ordered spike trains.

The ganglion cells themselves where modelled using Difference of Gaussians (DoG), Equation 6.1.

$$DoG_w(x,y) = \pm\frac{1}{2\pi\sigma_{w,c}^2}e^{\frac{-(x^2+y^2)}{2\sigma_{w,c}^2}} \mp \frac{1}{2\pi\sigma_{w,s}^2}e^{\frac{-(x^2+y^2)}{2\sigma_{w,s}^2}} \tag{6.1}$$

The size of the receptive field of the simulated cells depends on the layer they belong to, this is reflected in the convolution kernel's width and parameters. Variables $\sigma_{w,c}$ and $\sigma_{w,s}$ are the standard deviation for the centre and surround components of the DoGs for layer w. The signs for the equation will be (−,+) if the ganglion cell is *off-centre* and (+,−) if it is *on-centre*. The parameters for this equation can be found in table 6.1.

Table 6.1: Simulation parameters for ganglion cells

Layer	Behaviour	Matrix width	Centre std. dev. (σ_c)	Surround std. dev. (σ_s)	Sampling resolution (cols, rows)
1	OFF-centre	3	0.8	$6.7 \times \sigma_c$	1, 1
2	ON-centre	11	1.04	$6.7 \times \sigma_c$	1, 1
3	OFF-centre	61	8	$4.8 \times \sigma_c$	5, 3
4	ON-centre	243	10.4	$4.8 \times \sigma_c$	5, 3

For each cell type, a convolution kernel must be computed and stored in a matrix (DoG_w). For each element in the matrix we use Equation 6.1, substituting parameters specified in Table 6.1 and integer valued x-y coordinates whose origin is the centre of the matrix. For example, for the 3×3 kernel (layer 1 cells), the upper-left value would be calculated as follows:

$$\begin{aligned} DoG_3(x,y) &= -\frac{1}{2\pi\sigma_{3,c}^2}e^{\frac{-(x^2+y^2)}{2\sigma_{3,c}^2}} + \frac{1}{2\pi\sigma_{3,s}^2}e^{\frac{-(x^2+y^2)}{2\sigma_{3,s}^2}} \\ DoG_3(-1,-1) &= -\frac{1}{2\cdot\pi\cdot 0.8^2}e^{\frac{-((-1)^2+(-1)^2)}{2\cdot 0.8^2}} + \frac{1}{2\cdot\pi\cdot 5.36^2}e^{\frac{-((-1)^2+(-1)^2)}{2\cdot 5.36^2}} \\ &= 0.27399398 \end{aligned} \tag{6.2}$$

The procedure to encode images can be broken into two parts. First, algorithm 1 simulates the ganglion cells. It requires four independent 2D convolutions (Eq. 6.3) using DoG kernels calculated as explained in the previous paragraph.

$$C(x,y,w) = I * DoG_w = \sum_i \sum_j \left(I(i+x, j+y) \cdot DoG_w(i,j)\right) \tag{6.3}$$

We'll call coefficients to the pixel values that come out of the convolutions (Figures 6.1b, 6.1c, 6.1d and 6.1e). This coefficients are interpreted as a quantity that is inversely proportional to the spike emission time. That is, the pixel with the largest coefficient value represents the ganglion cell that will spike first.

In order to check the validity of the generated spikes, a reconstruction procedure is employed (Equation 6.4). Each coefficient in C has an origin layer w, a value c and a position (k, l). For all coefficients in C, a DoG from it's respective layer will be weighed by it's value and be added

Algorithm 1 FoCal, Part 1

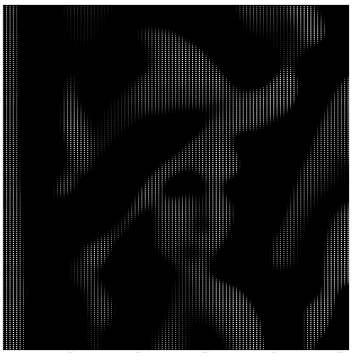
procedure GANGLIONCELLS(image I, kernels DoG)
  $C \leftarrow \emptyset$
  for all $w \in Layers$ **do**
    $C \leftarrow C \cup I * DoG_w$
  end for
  return C
end procedure

(a) Original image

(b) Layer 1

(c) Layer 2

(d) Layer 3



(e) Layer 4

Figure 6.1: Results of simulating ganglion cell layers (convolved images were enhanced for better contrast)

at the it's position to the reconstructed image R. The procedure is based on the assumption that the DoG are orthogonal basis. Figure 6.2b shows the result of the image reconstruction procedure without any redundancy correction applied.

$$R(x,y) = \sum_i \sum_j \sum_w C_w(i-x, j-y) DoG_w(i,j) \tag{6.4}$$

The eye is unlikely to provide unnecessary information to the brain, that is redundant information is filtered somehow before it's delivered. In the retina, lateral inhibition is the most likely candidate to minimize information redundancy. It's still a matter of discussion where and by which cells is lateral inhibition performed in the retina. It is most likely to happen in layers prior to the ganglion cell layer. The DoG kernels are only an approximately orthogonal basis, thus the resulting coefficients from the convolutions in Algorithm 1 suffer from redundant information. That is, two neighbouring pixels might contain information that represent the same feature in the image. The main issue with this redundancy is that neighbouring coefficients might encode almost the same information with a similar value. Since the value provides the

(a) Original image (b) 100% of raw spikes

Figure 6.2: Reconstruction results without overlap correction.

order of the spikes, this phenomenon will push other important coefficients into the later, less important, parts of the spike representation. In order to correct for redundancy, FoCal performs a second step (Algorithm 2).

Algorithm 2 FoCal, Part 2

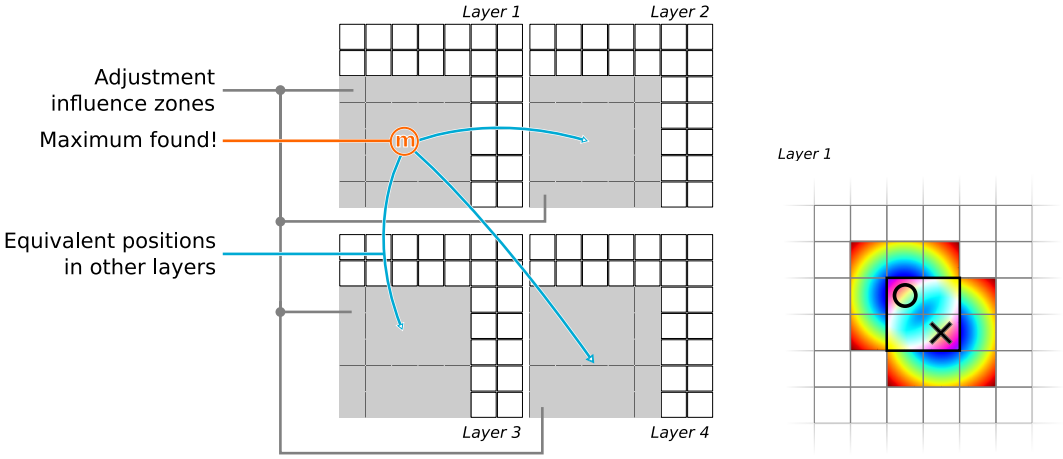
```
procedure CORRECTION(coeffs C, correlations Q)
    N ← ∅                                          ▷ Corrected coefficients
    repeat
        m ← max(C)
        M ← M ∪ m
        C ← C \ m
        for all c ∈ C do                           ▷ Adjust all remaining c
            if Q(m, c) ≠ 0 then                    ▷ Adjust only near
                c ← c − m × Q(m, c)
            end if
        end for
    until C = ∅
    return M
end procedure
```

All the coefficients that where obtained from Algorithm 1, are put in a set C. For every step of the correction procedure the maximum coefficient is searched and its spatially surrounding pixels in all layers (Figure 6.3a) will be adjusted according to the correlation due to overlap between the maximum coefficient's convolution kernel and the other layer's kernels. The bold square in Figure 6.3b shows the overlap of two 3×3 kernels of neighbouring pixels, a similar overlap is considered for the interaction between layers.

After this correction algorithm is applied only non-redundant spikes are preserved, this results in a much better reconstruction (Figure 6.4b). Not only is it more visually pleasing, but the fidelity of the reconstruction has been tested quantitatively; another interesting result is that only 10% of the rank-ordered and FoCal corrected spikes are needed to preserve 90% of the visually important information [29].

Implementation details

Different ways of applying convolutions to images on a GPU where implemented and evaluated. The first one, the **naïve approach**, implies a discrete convolution with the full 2D kernels. Since we are using squared kernels, this means $N^2 \times W \times H$ operations for a $W \times H$ image using a kernel of width N. As expected, performance drops quickly and the biggest problem for this approach was that biggest kernel (243×243 elements) requires more resources than the GPU's



(a) FoCal correction influence zones.

(b) Kernel overlap

(a) Original image

(b) 100% of *corrected* spikes

(c) 30% of *corrected* spikes

Figure 6.4: Results of reconstruction procedure

constant memory can provide (240 KBytes vs. 64 KBytes). This results in execution errors that may only be fixed using memory with greater latency to store the convolution kernel.

The second approach to perform a 2D DoG convolution with an image is to rely on **kernel separability**. A convolution kernel K is said to be *separable* if $K = K_1 * K_2 \dots K_n$. Gaussian kernels are separable (Eq. **??**). and, fortunately a DoG is merely the subtraction of them (Eq. 6.5).

$$O = I * DoG = I * G_c - I * G_s \tag{6.5}$$

Applying the algebraic properties of convolutions and the fact that Gaussian kernels are separable, the full 2D DoG convolution can be performed using four 1D separated ones (Eq. 6.6).

$$O = I * DoG = G_{v,c} * G_{h,c} * I - G_{v,s} * G_{h,s} * I \tag{6.6}$$

The main advantage of the separated kernel approach is a reduction of the number of operations needed ($4N \times W \times H$). An exception happens for the 3×3 kernel, in this case there are 12 operations vs. the 9 needed for the naïve approach.

The last approach, *Tiled Convolution* is reported by Advanced Micro Devices (AMD) [30]. They only present kernels of size 3×3, but we have an 11×11 convolution working; we are still developing solutions for the larger kernels.

Convolution alone is a compute intensive task and we obtain about 12 frames-per-second (FPS) on videos with 640×360 8-bit grayscale pixel resolution. Encoding was carried out using a desktop computer running 64-bit GNU/Linux, with a Core i5-4570 4-core CPU @ 3.20 GHz

processor with 8 GBytes of 64-bit DDR3 RAM @ 1600 MHz and a GeForce GT 720 GPU with 192 CUDA cores @ 797 MHz, 1 GBytes of 64-bit DDR3 RAM @ 1800 MHz.

Table 6.2: Convolution performance comparison.

	Layer 0	Layer 1	Layer 2	Layer 3
Naïve	0.0009s	0.0031s	0.0587s	N/A[1,2]
Separated	0.0021s	0.0055s	0.0172s	0.0472s
Tiled	0.0009s	0.0044s	0.1643	N/A[2]

[1] Unable to fit convolution kernel into constant memory.
[2] Unable to compile OpenCL code.

The performance of convolution in GPUs is bound by memory transfers, even if some of the information is reused.

In the retina, redundancy of information is reduced via lateral inhibition prior to any ganglion cell activity. In this algorithm, we perform a correction on the convolved images by adjusting the pixel values according to the correlation between convolution kernels (Alg. 2). The results of using correction (Fig. 6.2b) or not (Fig. 6.4b) show that the convolution stage can only provide redundant information. Furthermore, using only 30% of the corrected weights still provides enough visual information to reconstruct the original image [28].

Correcting the spikes for redundancy is a highly time consuming task which might be better suited for event-based programming, such as the one found on the SpiNNaker platform. We are still working on an implementation for this approach.

12fps is for good most phenomenon, full image encoding
This probably happens only once every so many ms

A dynamic vision sensor emulator

Output what a DVS does but with a camera as a source

Convolution of current and past frames ? centre - current / surround - past

Per-pixel adaptive threshold keeps fast changing pixels from spiking constantly, emulates refractory period of cells.

A second way of encoding is to simulate the early stages of the retina, which sense changes in intensity on the photoreceptors. This is quite similar to what real Dynamic Vision Sensors (DVS) do but with limited dynamic range and lower temporal resolution [2, 3]. The main advantage is that no specialized hardware is needed and the operation is so fast that any recent computer should be able to do it. For this type of encoding procedure we hypothesize that the bigger the change, the sooner a cell would spike and, thus, we can obtain a spike timings given the difference of two video frames. So far we can process about 20 and 25 FPS using a Numpy and an OpenCL back-end, respectively (using the same hardware set-up previously described). Although it's currently a good approximation, more research on this algorithm is needed to better approximate to biology.

6.2 Dataset creation

dataset for article

6.3 Conclusions

conclusions rank-ordered images

CHAPTER 7

Conclusions

Bibliography

1. Thrun, S. & Leonard, J. Simultaneous localization and mapping. *Springer handbook of robotics,* 871–889 (2008).
2. Leñero-Bardallo, J., Serrano-Gotarredona, T. & Linares-Barranco, B. A Five-Decade Dynamic-Range Ambient-Light-Independent Calibrated Signed-Spatial-Contrast AER Retina With 0.1-ms Latency and Optional Time-to-First-Spike Mode. *Circuits and Systems I: Regular Papers, IEEE Transactions on* **57,** 2632–2643. ISSN: 1549-8328 (2010).
3. Lichtsteiner, P., Posch, C. & Delbruck, T. A 128 x 128 120 dB 15 us Latency Asynchronous Temporal Contrast Vision Sensor. *Solid-State Circuits, IEEE Journal of* **43,** 566–576. ISSN: 0018-9200 (2008).
4. Furber, S. B., Galluppi, F., Temple, S., Plana, L., *et al.* The spinnaker project. *Proceedings of the IEEE* **102,** 652–665 (2014).
5. Thorpe, S, Fize, D & Marlot, C. *Speed of processing in the human visual system.* 1996. doi:10.1038/381520a0.
6. VanRullen, R., Guyonneau, R. & Thorpe, S. J. Spike times make sense. *Trends in Neurosciences* **28,** 1–4. ISSN: 01662236 (2005).
7. Behnke, S. *Hierarchical Neural Networks for Image Interpretation* 244. ISBN: 3540407227. doi:10.1287/mksc.1060.0207 (2003).
8. Bengio, Y. Learning Deep Architectures for AI. *Foundations and Trends® in Machine Learning* **2,** 1–127. ISSN: 1935-8237 (2009).
9. Durrant-Whyte, H & Bailey, T. Simultaneous localization and mapping (SLAM). *IEEE Robotics & Automation Magazine* **13,** 99–116. ISSN: 1070-9932 (2006).
10. Fuentes-Pacheco, J., Ruiz-Ascencio, J. & Rendón-Mancha, J. M. Visual simultaneous localization and mapping: a survey. *Artificial Intelligence Review,* 1–27. ISSN: 02692821 (2012).
11. Milford, M., Wyeth, G. & Prasser, D. RatSLAM: a hippocampal model for simultaneous localization and mapping. *Robotics and Automation, ...* 403–408. ISSN: 1050-4729 (2004).
12. Anh-Dung, D. Deconvolutional Networks.
13. Braitenberg, V. Brain. *Scholarpedia* **2,** 2918 (2007).
14. Thompson, R. *The Brain: A Neuroscience Primer* ISBN: 9780716732266. <https://books.google.co.uk/books?id=PPAaU1cQUPsC> (Worth Publishers, 2000).

15. Wikipedia. *Diagrams of brain and neural anatomy, modified to fit paper.* <http://wikipedia.org/> (2015).

16. Nielsen, J. A., Zielinski, B. A., Ferguson, M. A., Lainhart, J. E. & Anderson, J. S. An Evaluation of the Left-Brain vs. Right-Brain Hypothesis with Resting State Functional Connectivity Magnetic Resonance Imaging. *PLoS ONE* **8,** e71275 (Aug. 2013).

17. Hubel, D. H., Wensveen, J. & Wick, B. *Eye, brain, and vision* (Scientific American Library New York, 1995).

18. Nemri, A. Santiago Ramón y Cajal. *Scholarpedia* **5,** 8577 (2010).

19. Cervantes, C. V. *Images of neurons from: Ramón y Cajal. Recuerdos de mi vida.* <http://cvc.cervantes.es/ciencia/cajal/cajal_recuerdos/recuerdos/laminas.htm> (2015).

20. Bullock, T. H. *et al.* The Neuron Doctrine, Redux. *Science* **310,** 791–793 (2005).

21. Goodenough, D. A. & Paul, D. L. Gap junctions. *Cold Spring Harb Perspect Biol* **1,** a002576 (2009).

22. Dayan, P. & Abbott, L. F. *Theoretical neuroscience* (Cambridge, MA: MIT Press, 2001).

23. Hodgkin, A. L. & Huxley, A. F. A quantitative description of membrane current and its application to conduction and excitation in nerve. *The Journal of Physiology* **117,** 500–544. ISSN: 1469-7793 (1952).

24. Izhikevich, E. M. Dynamical Systems in Neuroscience, 441 (2007).

25. FitzHugh, R. Impulses and physiological states in theoretical models of nerve membrane. *Biophysical journal* **1,** 445 (1961).

26. Izhikevich, E. M. Simple model of spiking neurons. *IEEE Transactions on neural networks* **14,** 1569–1572 (2003).

27. Izhikevich, E. M. Which model to use for cortical spiking neurons? *IEEE transactions on neural networks* **15,** 1063–1070 (2004).

28. Sen, B. & Furber, S. *Evaluating Rank-order Code Performance Using a Biologically-derived Retinal Model* in *Proceedings of the 2009 International Joint Conference on Neural Networks* (IEEE Press, Atlanta, Georgia, USA, 2009), 1835–1842. ISBN: 978-1-4244-3549-4.

29. Sen, B. *Information Recovery From Rank-Order Encoded Images* Doctor of Philosophy Thesis (Faculty of Engineering and Physical Sciences, University of Manchester, 2008).

30. AMD. *Tiled Convolution: Fast Image Filtering* <http://developer.amd.com/resources/documentation-articles/articles-whitepapers/tiled-convolution-fast-image-filtering/> (2015).