RetINaBox: A hands-on learning tool for experimental neuroscience

Brune Bettler[1,*], Flavia Arias Armas[1,*], Erica Cianfarano[1,2,*], Vanessa Bordonaro[1], Megan Liu[1], Matthew Loukine[1,2], Mingyu Wan[1], Aude Villemain[1], Blake Richards[1,2,3,4], and Stuart Trenholm[1,#]

[1]Montreal Neurological Institute, McGill University, Montreal, Canada
[2]Mila, Montreal, Canada
[3]School of Computer Science, McGill University, Montreal, Canada
[4]CIFAR Learning in Machines and Brains, Toronto, Canada
** These authors contributed equally*
 # Corresponding author: stuart.trenholm@mcgill.ca

Abstract:

An exciting aspect of neuroscience research is developing and testing hypotheses via experimentation. However, due to logistical and financial hurdles, this compelling part of neuroscience research is generally lacking in classroom education. To address this issue, here we introduce RetINaBox: a low-cost open-source electronic retina simulator that provides users with a hands-on system to discover how the visual system builds feature detectors. RetINaBox features an LED array for generating visual stimuli and a photodiode array that acts as a mosaic of model photoreceptors. Custom software on a Raspberry Pi computer reads out responses from model photoreceptors and allows users to control the polarity and delay of the signal transfer from model photoreceptors to model retinal ganglion cells. Interactive lesson plans are provided, guiding users to discover different types of visual feature detectors—including center-surround, orientation selective, and direction selective receptive fields—as well as their underlying circuit computations.

RetINaBox

Introduction

The manner in which the brain encodes sensory stimuli varies across brain areas and is often far from predictable[1–5]. Thus, to discover how sensory inputs are represented in the brain we need to directly record from individual neurons while providing sensory stimulation[5]. A challenge in neuroscience education is how to incorporate such experimental work into lesson plans, so that instead of solely learning lists of facts, students get hands-on experience that captures the excitement of discovery. Laboratory classes have long been used to address this challenge, but their scope is often limited by financial, infrastructural, technical, ethical, and training constraints. For example, *in vivo* single-unit recordings from animal brains during sensory stimulation are widely used to capture the neural code but are too difficult to recapitulate in pedagogical settings.

To this end, we developed RetINaBox ("retina in a box") as a neuroscience focused educational tool that provides users with an interactive, hands-on system in which to discover how the visual system implements several important feature selective computations that were discovered through single-cell neurophysiological recordings[5]. RetINaBox consists of an ultra-low-cost computer, simple electronic components (LEDs and photodiodes), 3D printed parts, and custom-written open-source software. Through several lesson plans, RetINaBox exposes users to numerous computations in the visual system, including center-surround receptive fields[6,7], orientation selectivity[3,8] and direction selectivity[9,10]. RetINaBox also includes a discovery mode: users load a preset into RetINaBox but are not shown the wiring scheme; users then need to test different visual stimuli until they discover the target stimulus; finally, users are tasked with discovering the circuit wiring scheme that underlies the target stimulus selectivity.

Materials and Methods:

RetINaBox comprises an electronic retina combined with software to readout and control incoming visual signals. A complete parts list for building RetINaBox is provided in the user manual, which is included in the supplementary materials and on Github. The user manual also includes a step-by-step guide for building RetINaBox. To ensure that RetINaBox is as cost-effective as possible, we built it around a Raspberry Pi 500, which features a Raspberry Pi 5 minicomputer integrated into a keyboard, with easy access to GPIO pins for sending signals to external devices and monitoring external signals. We provide custom-written open-source software for running RetINaBox. The user manual includes instructions for software installation and use.

Results:

Design of RetINaBox

We sought to develop a model retina that could be paired with a computer to allow users to explore how visual inputs detected by retinal photoreceptors are processed to generate feature selective responses in downstream visual neurons. To do so, we designed a simple electronic retina, with a built-in visual stimulator, and connected it to a Raspberry Pi computer loaded with software for simulating visual circuit processing.

Simplified model retina: While the vertebrate retina is a complex tissue comprised of several different cell types located in specific anatomical layers[11] (**Figure 1A**), here we designed a simplified retina with an array of 9 model photoreceptors that can be connected to 2 model retinal ganglion cells (RGCs; **Figure 1B**). In the retina, photoreceptors form a single layer of cell bodies and are spatially organized as a lattice, or mosaic, next to one another, such that the light-receptive outer segments of neighbouring photoreceptors detect changes in light intensity in neighbouring regions of visual space[11–13]. To model photoreceptors, we used photodiodes, semiconductor diodes sensitive to changes in photon flux. To model a mosaic of photoreceptors, we built a 3 x 3 array of photodiodes (**Figure 1B,C**). This is sufficient to implement center-surround, orientation selective and direction selective computations. The photodiodes are powered by the Raspberry Pi, and the output of each photodiode is sampled by a different GPIO pin from the Raspberry Pi.

Visual stimulation: So that each photodiode can be independently activated, we aligned the photodiode array with a 3 x 3 LED array (**Figure 1C**), with each LED being independently controlled by a different GPIO pin from the Raspberry Pi. To ensure that photodiode activation is selectively modulated by the stimulation LEDs and not from variations in ambient light levels, we used infrared (IR) stimulation LEDs and IR-sensitive photodiodes. However, so that users can see the pattern and timing of LED stimulation on RetINaBox (the IR LEDs being invisible to the human eye), we added a second 3 x 3 array of visible (white light) LEDs pointing upward (i.e. away from the photodiodes; **Figure 1C**)).

Graphical user interface for 'performing experiments': To control LED stimulation, monitor model photoreceptor responses, and connect model photoreceptors to model retinal ganglion cells, we designed a graphical user interface (GUI) to control RetINaBox (**Figure 1D,E**). The GUI has 3 panels: a) Visual Stimulus Controller; b) Connectivity Manager; c) Signal Monitor (**Figure 1D**).

The Visual Stimulus Controller (**Figure 1D**) allows the user to turn on each LED in the 3 x 3 array. Visual stimuli can be displayed in a static manner or can be made to move either leftward or rightward at 3 different speeds. As described in the lesson plan document, for initial testing of circuits in RetINaBox we recommend using the Visual Stimulus Controller to deliver specific patterns of light to the model photoreceptors. However, once the user is satisfied that they have generated appropriate feature selective circuits for a given activity, we recommend turning on all the stimulus LEDs and placing real-world stimuli between RetINaBox's stimulus LEDs and photodiodes to test the robustness of the ganglion cell's feature selectivity. Along these lines, in the user manual users are instructed to make shapes with modelling clay on a clear plastic sheet which serves as the Visual Stimulus Tool. Alternatively, users can use their hands or pieces of paper/cardboard of various shapes to create visual stimuli.

The Connectivity Manager (**Figure 1D,E**) allows users to connect the output from each model photoreceptor to a model retinal ganglion cell. The GUI features 2 model retinal ganglion cells. By clicking on a model ganglion cell (**Figure 1D**), a pop-up allows users to modify the connectivity of each photoreceptor to that ganglion cell (**Figure 1E**). Each model ganglion cell can receive input from up to all 9 model photoreceptors. For each connection, the polarity (silent, +, or -) and time delay (none, short, medium, long) can be adjusted. The response threshold for each ganglion cell can be set from between 1 to 9, indicating how many positive photoreceptor inputs, calculated after pooling all its inputs, it needs to receive to respond. Ganglion cell output is binary: on or off.

The Signal Monitor (**Figure 1D**) plots when each model photoreceptor is activated by the visual stimulus, the polarity of the signal transfer from each photoreceptor to each model ganglion cell, and each model ganglion cell's output response (**Figure 1D,F**). In addition, the output signals of the 2 model retinal ganglion cells are indicated by 2 colored LEDs (RGC1, green; RGC2, yellow) on the top of RetINaBox (**Figure 1C,F**). Furthermore, the output of the ganglion cells can be used to drive external electronic components via output pins on the rear of RetINaBox that send out 3.3 V signals when a given model retinal ganglion cell is activated (**Figure 1C,F**).

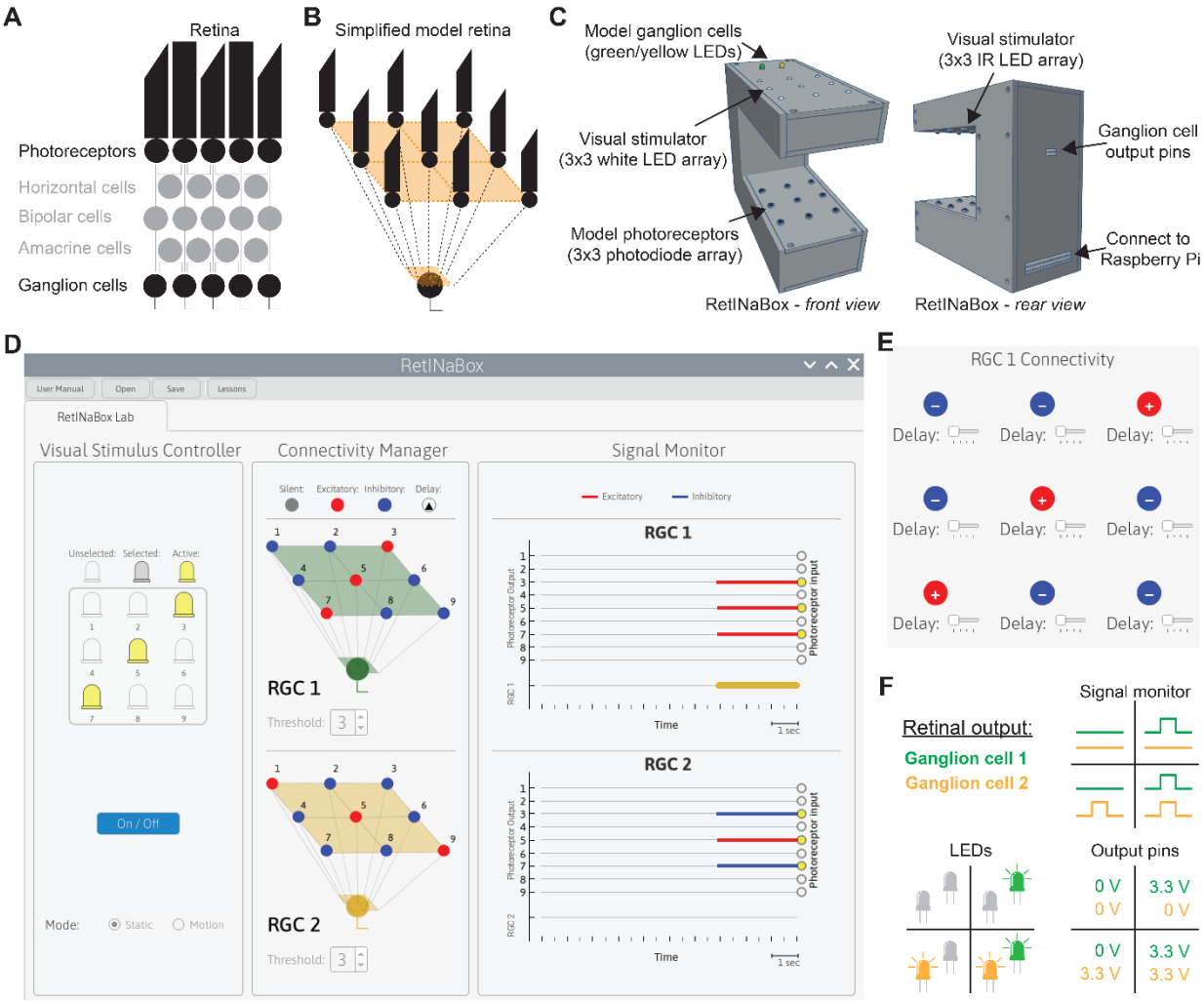

Figure 1. *RetINaBox design and GUI overview*. A schematic of the actual retinal circuitry (**A**) and the simplified model circuitry of RetINaBox (**B**). **C**, An overview of RetINaBox, including the 3D printed case, the visual stimulator (3x3 LED array) and photoreceptor mosaic (3x3 photodiode array), along with the connectivity port between the RetINaBox and the Raspberry Pi, and the output signals of the two model retinal ganglion cells (2 color LEDs on top of RetINaBox and 3.3V output pins on the back of RetINaBox). **D**, The main RetINaBox GUI, showing the Stimulus Controller, Connectivity Manager, and Signal Monitor. **E**, The Connectivity Manager pop-up window, which allows users to set the connectivity (silent, +, or -) and delay (none, short, medium, or long) of the connection between each of the 9 model photoreceptors and each of the two retinal ganglion cells (RGCs). **F**, Method for monitoring RGC activity, either by watching the RGC responses in the GUI's Signal Monitor, or by monitoring the yellow and green LEDs on the top of RetINaBox, or by connecting external electronic devices to the 3.3V output pins on the back of RetINaBox.

Lessons

Once users have built RetINaBox and installed the software on the Raspberry Pi (all these steps are covered in detail in the user manual), users are guided through 4 lesson plans. The first 3 lessons guide users through the circuit mechanisms underlying canonical visual system computations: center-surround (lesson 1), orientation selectivity (lesson 2), and direction selectivity (lesson 3). Lesson 4 is focused on discovery, with users tasked with discovering the preferred visual stimulus of a mystery RetINaBox ganglion cell and then discovering the underlying circuit connectivity.

Lesson 1 – Crack a code with center-surround receptive fields: Following experiments that discovered that retinal ganglion cells have spatially-localized receptive fields (i.e. they only 'see' within a small part of the visual scene), with some responding only to increases in luminance over their receptive field, some only responding to decreases in luminance, and some responding to both increases and decreases in luminance[14], it was then discovered that there were additional spatial details regarding their receptive fields: center-surround organization[5–7,11]. As an example, it was found that for an ON-center retinal ganglion cell, increasing luminance with a spot of light located directly above the cell—usually corresponding to the location of its cell body and most of its dendritic tree—maximally activated the cell. However, if the size of the visual stimulus is increased beyond the receptive field center, the cell's response decreased due to activation of an inhibitory surround. Such center-surround receptive field structure means

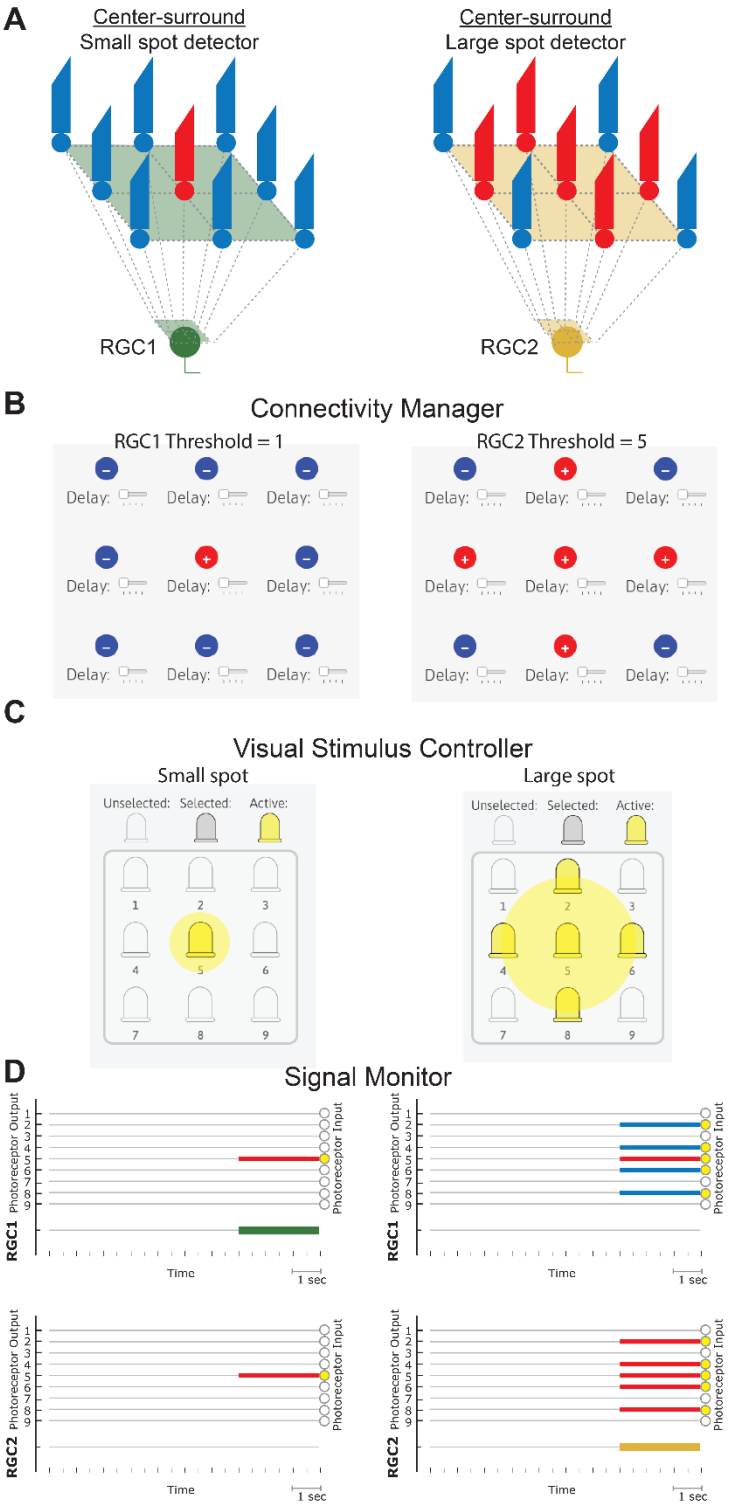

Figure 2. *Lesson 1 – Center-surround receptive fields*. **A**, Example circuits for generating RGCs with center-surround receptive fields with preferences for small (*left*) and large (*right*) spots. Example settings for the Connectivity Manager (**B**) and the Visual Stimulus Controller (**C**). **D**, Example Signal Monitor output for the settings outlined above, with the stimulus LEDs turned on for 3 seconds.

that retinal ganglion cells are optimized to respond to local luminance contrast[5]. The surround inhibition that underlies center-surround receptive fields arises from inhibitory lateral connections, which in the vertebrate retina can arise from horizontal cells and amacrine cells[11].

Lesson 1 of RetINaBox asks users to consider what ecological purposes a center-surround receptive field has and how they could build a ganglion cell with a center-surround receptive field organization. The goal is to connect the photoreceptors to RGC1 so that it will respond when a single model photoreceptor is activated, but not when that photoreceptor is activated at the same time as other photoreceptors in the array (**Figure 2A,B**; e.g. generate a ganglion cell that is selective to a small spot of light centered on the middle of the photoreceptor array). The next goal is to generate a second ganglion cell with the same feature selectivity but located in a different part of the visual field. Finally, users are tasked with generating two ganglion cells with the same receptive field center location, but with preferences for spots of different sizes (**Figure 2B-D**). If users need assistance solving these problems, they can consult the lesson plan document or check out example Connectivity Manager solutions which can be loaded as presets from the "Lessons" tab in the menu bar. In addition, the GUI allows users to save their own solutions and access them later.

Lesson 1 ends with users solving a code breaking game, which gets users to apply concepts related to center-surround receptive fields to solve the problem. Upon selecting a code breaking challenge in the GUI, users are provided with a code in the form of a series of visual stimuli, with each visual stimulus representing a letter from the alphabet (**Figure 3**). Users are also provided with a cipher containing information they need to use to solve the problem (**Figure 3**). The cipher indicates what the feature preference should be for the two model RGCs and then indicates which letters of

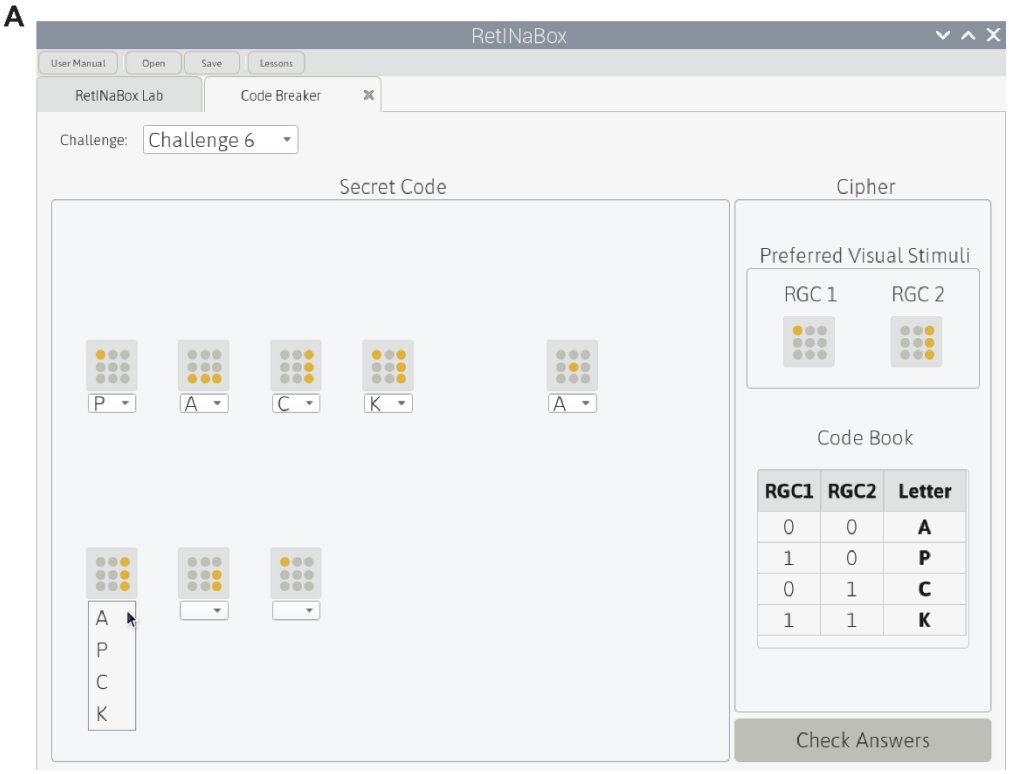

Figure 3. *Lesson 1 - Code breaker activity with center-surround receptive fields*. **A**, A visualization of the code breaker GUI tab related to Lesson 1. Users must use the cipher (*right*) to correctly set up the Connectivity Manager for the two RGCs to obtain the indicated feature selective responses. Next, users place the visual stimuli indicated on the left into RetINaBox and use the cipher instructions (*bottom right*) to transcribe the RetINaBox output into the correct letter for each visual stimulus in the code.

the alphabet correspond to neither ganglion cell being activated, one or the other ganglion cell being activated in isolation, or both ganglion cells being activated together. Once the Connectivity Manager is set according to the cipher (and in a manner that allows for the possibility that both RGCs can be co-activated), users present RetINaBox with the indicated visual stimuli using the Visual Stimulus Tool. Based on the output of the ganglion cells to each visual stimulus, users enter the corresponding letters into the GUI to solve the code (**Figure 3**). RetINaBox is packaged with multiple codes that users can try to crack.

Lesson 2 – Build a shape detector with orientation selective receptive fields: Following the work described above that discovered center-surround receptive fields of retinal ganglion cells[6,7], it was found that many neurons in the retino-recipient lateral geniculate nucleus (LGN) also tended to show center-surround receptive fields[15]. However, recordings from primary visual cortex (V1)—the part of cortex that serves as the gateway for conscious visual perception[16]—which receives strong visual inputs from the LGN, showed that most V1 neurons do not possess center-surround receptive fields. Instead, many V1 neurons exhibit orientation selective tuning, meaning that they are optimally activated by an extended edge (or line) in a specific part of the visual field, aligned in a specific orientation. It was posited that such orientation selective responses arise when a

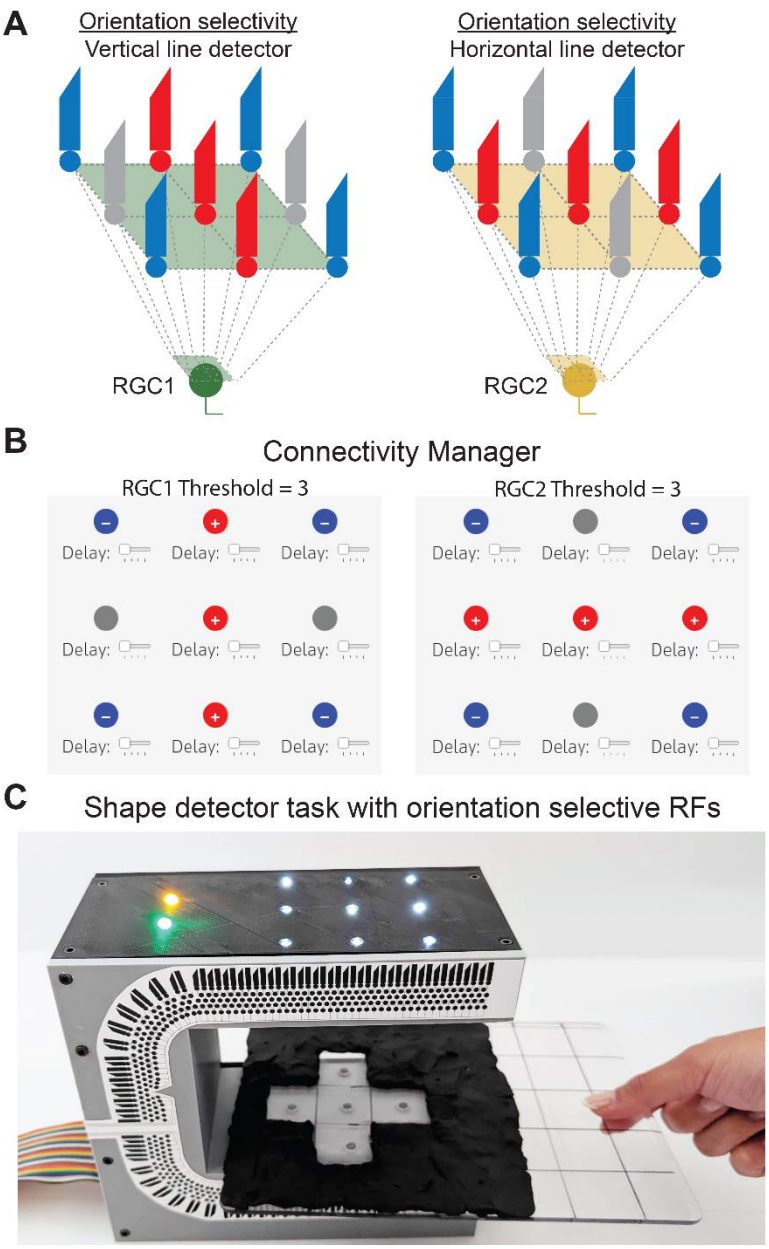

Figure 4. *Lesson 2 – Shape detector activity with orientation selective receptive fields*. **A**, Example circuits for generating RGCs with orientation selective receptive fields, but preferring lines of different orientations. **B**, Example settings for the Connectivity Manager. **C**, Demonstration of the shape detector activity: the settings in (**B**) were entered into the Connectivity Manager. A shape corresponding to that created by joining the preferred stimuli of RGC1 and RGC2 was made with the Visual Stimulus Tool, which was then placed in the correct spatial location between RetINaBox's stimulation LEDs and photoreceptor array. Both RGCs are activated: both yellow and green LEDs are turned on.

single V1 neuron pools inputs from LGN neurons whose receptive fields are spatially offset along a line. Such V1 neurons selectively respond to line-like stimuli that follow the alignment of their pooled LGN inputs[8,16]. While early work in cats and primates found that orientation selectivity appears first in V1[8,17,18] (but see also[19]), subsequent work in other species, including rabbits[20] and mice[21,22], has found that orientation selectivity can also arise at the level of retinal ganglion cells. Orientation selective signals can then be combined in downstream neurons in various ways, resulting in spatially invariant orientation selective cells (if a downstream neuron receives input from multiple orientation selective cells with the same orientation preference but with spatially offset receptive fields[8]) and in cells that respond to combinations of edges/lines, which can in turn serve as building blocks for visual object detectors[23].

Lesson 2 of RetINaBox begins by asking users to build a retinal ganglion cell with an orientation selective receptive field organization. The goal is to connect the photoreceptors to a ganglion cell in such a way that it will respond when a specific set of 3 adjacent photoreceptors are activated (i.e. a 3-LED-long line) but not when any other arrangements of photoreceptors are activated (**Figure 4A,B**). Next, the user is tasked with generating a second retinal ganglion cell with a preference for a line of a different orientation (**Figure 4A,B**). The next goal is to generate two ganglion cells with preferences for lines of the same orientation, but of different thicknesses. If users require help solving these problems, they can load preset Connectivity Manager solutions from the "Lessons" tab in the menu bar or consult the lesson plan document.

Lesson 2 ends with users being asked to build a shape detector, which forces users to apply concepts related to orientation selective receptive fields to solve the problem. First, users need to optimize the two ganglion cells to be selective for lines of different orientations at specific locations over the photoreceptor array, with the combination of the two lines representing a shape (**Figure 4C,D**). Then, users are instructed in basic electronics (see the user manual) to connect the digital outputs of the two ganglion cells (3.3V outputs located on the rear of the RetINaBox) to generate a buzzer sound that is activated via an AND gate, meaning that both ganglion cells need to be activated for the buzzer to sound. This is meant to be analogous to the way that some neuroscientists work, when they 'listen' to their experiments in real-time by playing their electrophysiological recordings through a speaker (which can be helpful for quickly finding stimuli that activate neurons).

Lesson 3 – Play a video game with direction selective receptive fields: David Hubel and Torsten Wiesel discovered that in V1 of cats, some cells, aside from being orientation selective, were also direction selective[8]. Soon afterward, direction selective retinal ganglion cells were described in the rabbit retina[9,10]. Direction selective retinal ganglion cells were subsequently found in a variety of species, including mice[24,25] and primates[26,27]. Direction selective visual responses have been particularly well studied in the fly retina[28]. Direction selective responses can help to stabilize eye/head movements with respect to the visual scene[29–31], and help an animal distinguish between external and self-generated movements in the visual scene[32,33]. Various models have been proposed and discovered in the brain for generating DS responses, including spatially offset excitation/inhibition and spatially asymmetric delays in inputs along the preferred-null axis[34].

Lesson 3 of RetINaBox asks users to build a retinal ganglion with a direction selective receptive field organization. The goal is to connect the photoreceptors to a retinal ganglion cell in such a way that it

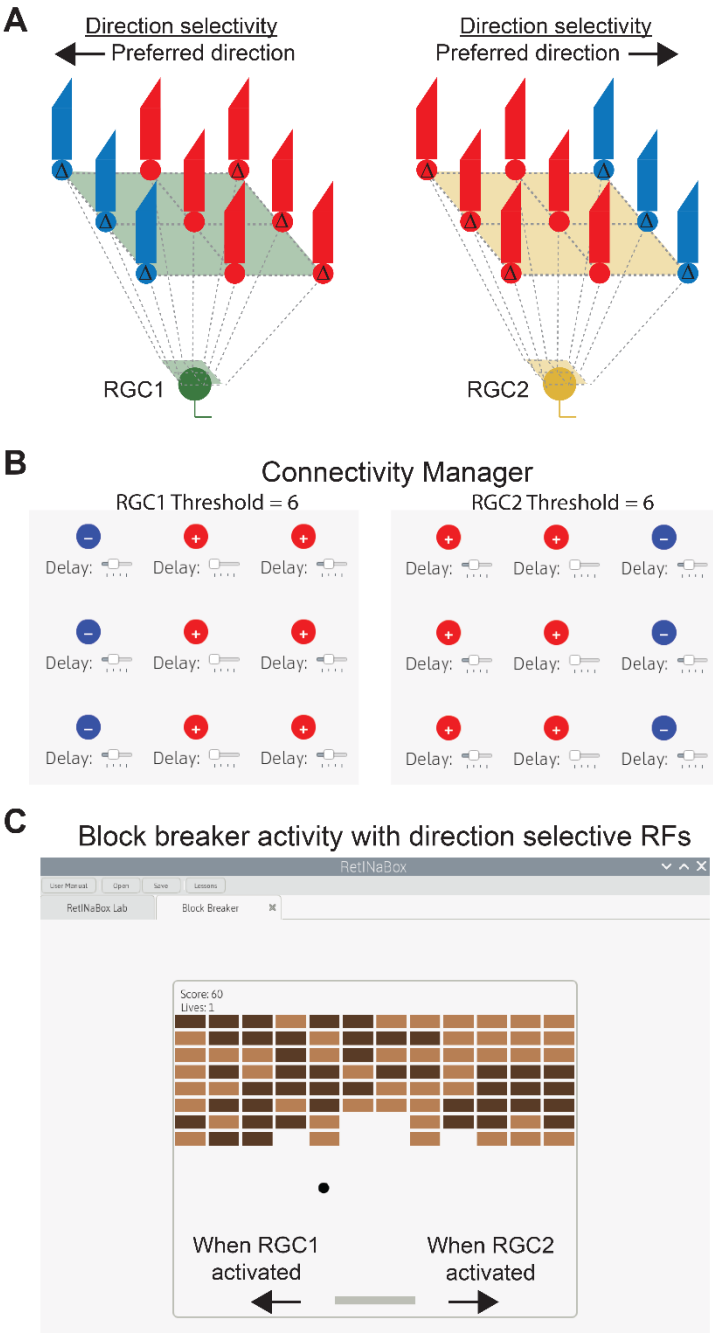

Figure 5. *Lesson 3 – Block breaker video game with direction selective receptive fields*. **A**, Example circuits for generating direction selective RGCs with preferred directions for leftward (*left*) and rightward (*right*) motion. **B**, Example settings for the Connectivity Manager. **C**, The RetINaBox GUI can be used to load a block breaker video game that takes its left and right input commands from the activity of RGC1 and RGC2, respectively.

will respond when the user moves a visual stimulus (e.g. the user's hand) in one direction over the photoreceptor array, but not respond when the same visual stimulus is moved in the opposite direction (**Figure 5A,B**). The next goal is to generate a second ganglion cell with a preference for motion in the opposite direction (**Figure 5A,B**). Then, users are instructed to generate 2 ganglion cells, each with a preference for the same visual stimulus moving in the same direction, but at different speeds. If users need assistance solving these problems, they can refer to example solutions by loading presets from the "Lessons" tab in the menu bar.

Lesson 3 ends with users being asked to build a virtual brain machine interface, which forces users to generate robust direction selective detectors and apply the output of their retina to control a video game, with left and right commands. First, users need to optimize the two ganglion cells to be selective for motion in opposite directions, as outlined above (**Figure 5A,B**). Then, users are instructed to open a pre-packaged video game in the GUI (**Figure 5C**) which uses the ganglion cell outputs as the digital inputs to the game.

Lesson 4 – Discovery mode: What does RetINaBox want to see? So far, each lesson has centered around a well-established feature selective computation in the visual system. These feature selective properties were discovered by scientists presenting varied sets of visual stimuli while recording from neurons, until eventually the stimulus that best activated each neuron was found. Once the preferred visual stimulus of a neuron was found, scientists then turned their attention to discovering the circuit wiring that enabled such a feature selectivity. To replicate this discovery aspect of scientific exploration,

Lesson 4 tasks users with discovering the visual features that drive mystery ganglion cells in RetINaBox and uncovering the circuits that underlies such feature selectivity (**Figure 6A-C**).

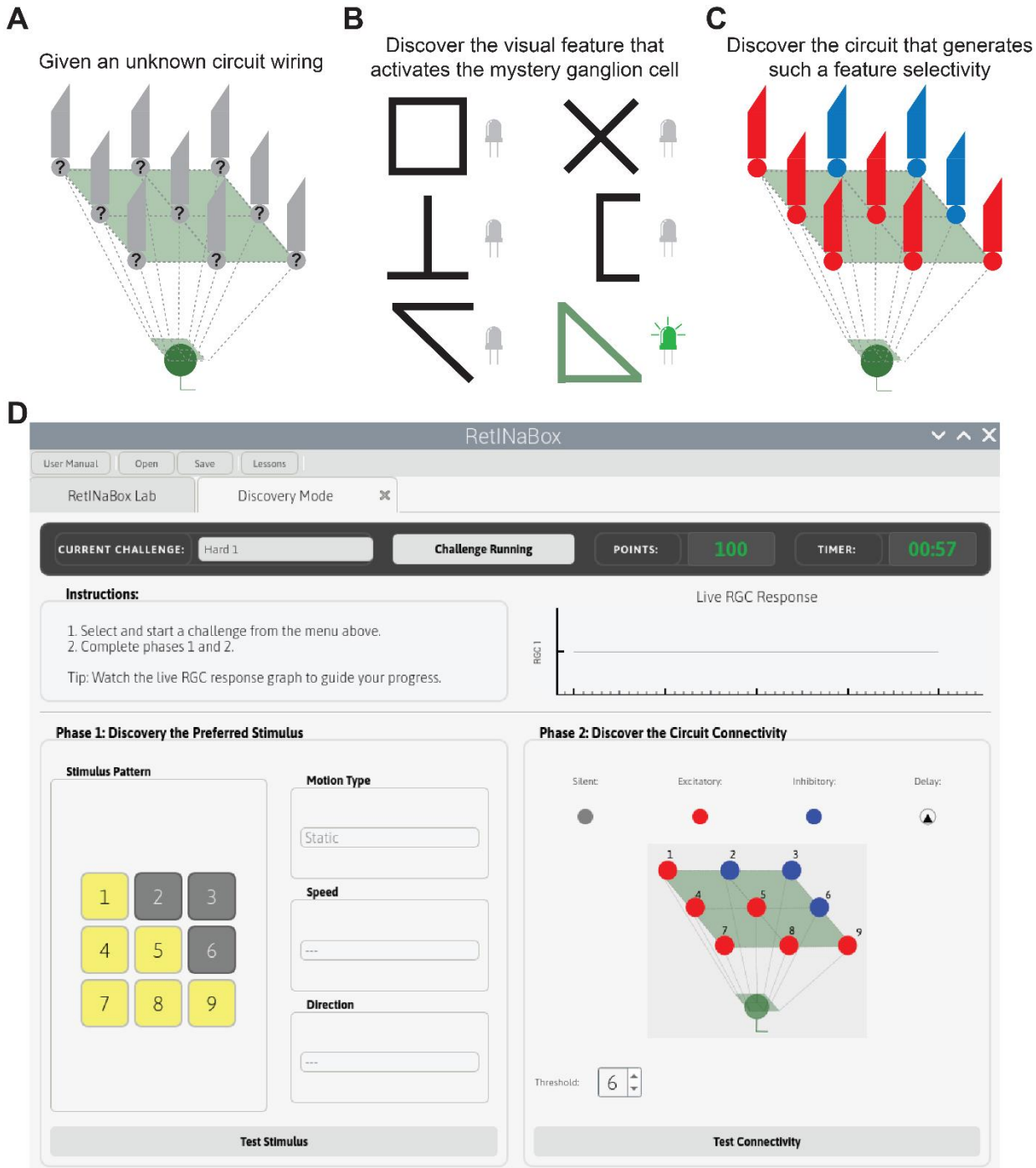

Figure 6. *Lesson 4 – Discovery mode.* Users select a mystery ganglion cell (**A**), test various visual stimuli to discover the preferred stimulus of the mystery ganglion cell (**B**), and discover the circuit connectivity that generates such a feature selectivity (**C**), all using the Discovery Mode GUI tab (**D**).

Discovery Mode opens a new GUI tab (**Figure 6D**). Discovery mode starts with users selecting a mystery ganglion cell from a list in the GUI. Mystery ganglion cells in the Discovery Mode menu section are organized based on difficulty (easy, medium, and hard). First, users are tasked with figuring out which visual stimulus activates the mystery ganglion cell by using the Visual Stimulus Tool, or their hands, to test various stimuli at different positions, orientations, and directions/speeds of movement over RetINaBox's photoreceptor array. Upon identifying the preferred visual stimulus for a ganglion cell, users can verify in the software that they have correctly identified the target visual stimulus (**Figure 6D**). Once users have found the target stimulus, users must discover the circuit that enables a ganglion cell to be selective for the target visual stimulus they just discovered (**Figure 6D**).

Discussion:

Taking inspiration from other groups that have sought to develop interactive learning tools targeted towards experimental neuroscience education[35,36], we developed RetINaBox in order to provide users with an opportunity to experience the joy of discovery in neuroscience experiments in a class-based setting.

RetINaBox is an electronic visual stimulation/detection device paired with a computer, that enables users to experiment with how the visual system takes photoreceptor-derived signals and generates various feature selective responses, including center-surround, orientation selective and direction selective circuits. For each of the 4 lessons that we developed, we provide detailed lesson plans. These lesson plans provide users with a problem-based learning approach for each lesson, in which users are encouraged to complete the tasks by reflecting on what visual feature detector they are trying to build, and which circuit components are available to them. Users are then guided to develop and test hypotheses while working towards a circuit model for each distinct feature detector. While going through the lessons and building/testing circuits, users learn important concepts in neuroscience, including excitatory and inhibitory synaptic connectivity, response thresholds, and spatio-temporal processing.

To ensure that RetINaBox is accessible to as many people as possible, we built it with low-cost parts and open-source software. All custom-written code, as well as detailed instructions for building and running the system, are available in the user manual. We provide 4 comprehensive lesson plans.

It is important to note that RetINaBox represents a simplification of actual biological circuits (e.g. as outlined in **Figure 1**, we bypass the cells in the inner retina and replace them with sign and delay functions). Nonetheless, we believe that RetINaBox provides a useful heuristic for approximating visual neuroscience experiments for educational purposes. Furthermore, in the lesson plans we include details about the biological circuits that underlie the various canonical visual feature detectors covered in lessons 1-3. In addition to lesson plans, we also provide teaching slides, meaning that RetINaBox can be easily incorporated into existing neuroscience classes. By varying the complexity of the lesson plans and the expected outcomes for each lesson, we believe that RetINaBox can serve as a powerful learning tool from high school to graduate school levels.

Contributions:

The software was developed by B.B. and S.T. and all code was written by B.B. RetINaBox's electronic retina was designed by B.B., F.A.A. and S.T.. The 3D printed components were designed by F.A.A. and S.T.. RetINaBox was assembled by F.A.A. The lesson plans were developed by E.C. and S.T and made by E.C.. The user manual was written by B.B., F.A.A., and S.T.. Beta testing was performed by V. B., M. Liu, and M. Loukine. Translations of the lesson plans and user manual were undertaken by F.A.A., V.B., M. Loukine, M.W., and A.V.. Teaching slides were generated by E.C.. The RetINaBox logo was designed by B.B. and the decorative retina image on our RetINaBox (as seen in Figure 4C) was designed by V.B.. B.R. assisted with conceptual development of the project. The manuscript was written by S.T.

Acknowledgements:

We thank A. Krishnaswamy for critical discussions. We thank H. Velde for noting that "retina in a box" featured "ina in a," which prompted us to shorten the name from Retina-In-A-Box to RetINaBox. We thank the following funding sources: a National Sciences and Engineering Research Council (NSERC) and Fonds de Recherche du Québec (FRQ) scholarship to E.C.; a Canadian Institutes of Health Research (CIHR) PhD fellowship to M. Liu; an NSERC Undergraduate Summer Research Award, a Tanenbaum Open Science Institute Launchpad award, and an NSERC graduate student scholarship M. Loukine; a Tomlinson PhD fellowship to M.W.; CIFAR Canada AI Chair (Learning in Machine and Brains Fellowship) to B.R.; a Canada Research Chair, an Alfred P. Sloan Foundation Research Fellowship, and NSERC Discovery Grants (RGPIN-2018-03852 and RGPIN-2025-05567) to S.T. We also acknowledge outreach funding from Brain Canada, the Vision Science Research Network, and the Canadian Association for Neuroscience.

Custom code, 3D-print files, the user manual, and the lesson plans can be found here: https://github.com/Trenholm-Lab/RetINaBox

References:

1. Lettvin, J. Y., Maturana, H. R., McCulloch, W. S. & Pitts, W. H. What the Frog's Eye Tells the Frog's Brain. *Proc. IRE* **47**, 1940–1951 (1959).
2. The Man Who Tried to Redeem the World with Logic - Nautilus. https://nautil.us/the-man-who-tried-to-redeem-the-world-with-logic-235253/.
3. Hubel, D. H. & Wiesel, T. N. Receptive fields of single neurones in the cat's striate cortex. *J. Physiol.* **148**, 574–591 (1959).
4. Hubel, D. H. & Wiesel, T. N. Chapter 7 - Our first paper, on cat cortex, 1959. in *Brain and Visual Perception* 59–67 (Oxford University Press).
5. Trenholm, S. & Krishnaswamy, A. An Annotated Journey through Modern Visual Neuroscience. *J. Neurosci. Off. J. Soc. Neurosci.* **40**, 44–53 (2020).
6. Barlow, H. B. Summation and inhibition in the frog's retina. *J. Physiol.* **119**, 69–88 (1953).
7. Kuffler, S. W. DISCHARGE PATTERNS AND FUNCTIONAL ORGANIZATION OF MAMMALIAN RETINA. *J. Neurophysiol.* **16**, 37–68 (1953).
8. Hubel, D. H. & Wiesel, T. N. Receptive fields, binocular interaction and functional architecture in the cat's visual cortex. *J. Physiol.* **160**, 106–154 (1962).
9. Barlow, H. B., Hill, R. M. & Levick, W. R. RETINAL GANGLION CELLS RESPONDING SELECTIVELY TO DIRECTION AND SPEED OF IMAGE MOTION IN THE RABBIT. *J. Physiol.* **173**, 377–407 (1964).
10. Barlow, H. B. & Levick, W. R. The mechanism of directionally selective units in rabbit's retina. *J. Physiol.* **178**, 477–504 (1965).
11. Krishnaswamy, A. & Trenholm, S. The Retina. in *The Open Brain* (2023).
12. Ahnelt, P. K. The photoreceptor mosaic. *Eye* **12**, 531–540 (1998).
13. Ahnelt, P. K. & Kolb, H. The mammalian photoreceptor mosaic-adaptive design. *Prog. Retin. Eye Res.* **19**, 711–777 (2000).
14. Hartline, H. K. THE RESPONSE OF SINGLE OPTIC NERVE FIBERS OF THE VERTEBRATE EYE TO ILLUMINATION OF THE RETINA. *Am. J. Physiol.-Leg. Content* **121**, 400–415 (1938).
15. Hubel, D. H. & Wiesel, T. N. Integrative action in the cat's lateral geniculate body. *J. Physiol.* **155**, 385-398.1 (1961).
16. Angelucci, A. & Trenholm, S. Primary Visual Cortex. in *Adler's Physiology of the Eye* 612–626 (Elsevier).
17. Hubel, D. H. & Wiesel, T. N. Receptive fields and functional architecture of monkey striate cortex. *J. Physiol.* **195**, 215–243 (1968).
18. Hubel, D. H. & Wiesel, T. N. Ferrier lecture. Functional architecture of macaque monkey visual cortex. *Proc. R. Soc. Lond. B Biol. Sci.* **198**, 1–59 (1977).
19. Passaglia, C. L., Troy, J. B., Rüttiger, L. & Lee, B. B. Orientation sensitivity of ganglion cells in primate retina. *Vision Res.* **42**, 683–694 (2002).
20. Levick, W. R. Receptive fields and trigger features of ganglion cells in the visual streak of the rabbit's retina. *J. Physiol.* **188**, 285–307 (1967).
21. Baden, T. *et al.* The functional diversity of retinal ganglion cells in the mouse. *Nature* **529**, 345–350 (2016).
22. Nath, A. & Schwartz, G. W. Cardinal Orientation Selectivity Is Represented by Two Distinct Ganglion Cell Types in Mouse Retina. *J. Neurosci.* **36**, 3208–3221 (2016).
23. Hesse, J. K. & Tsao, D. Y. The macaque face patch system: a turtle's underbelly for the brain. *Nat. Rev. Neurosci.* **21**, 695–716 (2020).
24. Weng, S., Sun, W. & He, S. Identification of ON–OFF direction-selective ganglion cells in the mouse retina. *J. Physiol.* **562**, 915–923 (2005).

25. Sun, W., Deng, Q., Levick, W. R. & He, S. ON direction-selective ganglion cells in the mouse retina. *J. Physiol.* **576**, 197–202 (2006).
26. Kim, Y. J. *et al.* Origins of direction selectivity in the primate retina. *Nat. Commun.* **13**, 2862 (2022).
27. Wang, A. Y. M. *et al.* An ON-type direction-selective ganglion cell in primate retina. *Nature* **623**, 381–386 (2023).
28. Borst, A., Haag, J. & Mauss, A. S. How fly neurons compute the direction of visual motion. *J. Comp. Physiol. A* **206**, 109–124 (2020).
29. Oyster, C. W., Takahashi, E. & Collewijn, H. Direction-selective retinal ganglion cells and control of optokinetic nystagmus in the rabbit. *Vision Res.* **12**, 183–193 (1972).
30. Yoshida, K. *et al.* A Key Role of Starburst Amacrine Cells in Originating Retinal Directional Selectivity and Optokinetic Eye Movement. *Neuron* **30**, 771–780 (2001).
31. Yonehara, K. *et al.* Congenital Nystagmus Gene FRMD7 Is Necessary for Establishing a Neuronal Circuit Asymmetry for Direction Selectivity. *Neuron* **89**, 177–193 (2016).
32. Britten, K. H. Mechanisms of Self-Motion Perception. *Annu. Rev. Neurosci.* **31**, 389–410 (2008).
33. Sabbah, S. *et al.* A retinal code for motion along the gravitational and body axes. *Nature* **546**, 492–497 (2017).
34. Mauss, A. S., Vlasits, A., Borst, A. & Feller, M. Visual Circuits for Direction Selectivity. *Annu. Rev. Neurosci.* **40**, 211–230 (2017).
35. Dragly, S.-A. *et al.* Neuronify: An Educational Simulator for Neural Circuits. *eNeuro* **4**, (2017).
36. Book: How Your Brain Works. *Backyard Brains* https://backyardbrains.com/products/book-how-your-brain-works.