As discussed in the introductory lecture, there are many levels at which we need to understand perception. Vision, audition, touch, etc are perceptual systems. As with understanding computers and computer systems, we need to think about how they work at high levels and low levels, from overall architecture down to the single cell where the most basic computations are carried out. This sometimes means jumping around from level to level, as certain aspects are best explained at one level rather than another. Even in a single lecture I may have to do some jumping around, and today is a good example of that.

I will begin today by briefly discussing the brain at a high level. I'll then discuss single cells and how they signal one another over distances. Finally I will get back to the retina and look at how the images are coded beyond the photoreceptors.

**Brain areas**

Beyond the sensory organs (eye, ear, skin, ...), most of the neural computation that we will be discussing occurs on the surface of the brain, known as the cortex. The brain’s surface is convoluted, with many small hills and valleys, and it is separated into lobes that have quite different functions. In the next several lectures we will be discussing some of the processing that occurs in the occipital lobe, which is at the back of the head (the right in the figure below). This is where the first steps of visual cortical computation occur. Further visual processing does occur in other lobes, for example, object and scene recognition and motion processing occurs in the temporal lobe and other spatial processing occurs in the parietal lobe.

![Brain Regions Diagram](https://askabiologist.asu.edu/brain-regions)

**Single neurons**

Last lecture I discussed briefly what it means for a neuron such as photoreceptor to "respond" to a stimulus (light). What does it means for a more general a neuron to "respond"? First, note in the figure below that a neuron’s cell body = extends into branches, like a tree, and these branches are called dendrites. When neurotransmitters are released from neighboring cells, these neurotransmitters can bind to the dendrites, which will causes a changes in the neuron’s membrane. As in the case of photoreceptors, the change is that membrane channels will open or close, allowing ions such as potassium and sodium to travel in or out of the cell. The net effect – what we usually call the "response" – is that the concentration of ions inside versus outside the cell will vary over time.
time, or equivalently, there may be a difference in electrical potential across the cell membrane. (One can measure this difference using clever "electrophysiology" techniques that I won’t go into.)

The resting (average) potential of a cell is typically about -70 mV (millivolts), namely the inside of the cell has more negative charge than the outside. If the potential difference is above -70 mV, then we say that the cell is depolarized (i.e. closer to 0), and if the potential differences is below -70 mV then we say that the cell is hyperpolarized (i.e. further from 0).

The communication between cells occurs at a location called a synapse. The cell releasing the neurotransmitter is the pre-synaptic cell and the cell receiving (binding) the neurotransmitter is the post-synaptic cell. Synapses (or the neurotransmitters that are released there) can be either depolarizing or hyperpolarizing. As we will see next, a depolarizing neurotransmitter is excitatory and a hyperpolarizing neurotransmitter is inhibitory.

**Spikes (Action Potentials)**

How do cells communicate over long distances? For example, how does a cell in the eye send a signal that can reach another cell in the visual cortex at the back of the head? How does a motor cell send a signal from the brain down the spinal cord and eventually reach the fingers or toes. Long distance signal can be done with a relay of cells, and often surprisingly few links in the relay are needed. The basic mechanism for long distance signally is called a spike or action potential, which is sudden depolarization of the cell membrane.
An action potential is triggered when the cell membrane reaches a certain depolarization threshold, which causes it to depolarize further and even become positive. The action potential is propagated over a distance as a single wave (spike) along a special part of the cell called the cell axon. Think of an axon as a long wire which is a cylindrical tube wrapped in a fatty insulator called myelin. There is a limit on how often action potentials can be sent, since after an action potential the cell needs to pause (“refractory state”). A cell typically can ”spike” up a rate of just over 100 times per second.

Models of spikes go back to the 1940’s. The McCulloch-Pitts model, for example, says that a neuron performs a sum of weighted inputs and if this sum is greater than some threshold, the neuron response is 1 (spike or ”fire”) and otherwise the response is 0. Other versions of this model use a continuous response which is sigmoid in shape instead of a ”step” function.

There is much to say about spikes but let’s just consider a few important facts for now. First, for a given cell, every spikes looks the same. The information carried by spikes is purely in the timing of the spikes, not the shape. There has been much effort in the past few decades to understand exactly how much the timing matters. On the one hand, the initiation of a spike depends on a somewhat noisy signal (namely, binding of neurotransmitters from neighboring cells) and so it is difficult to imagine how the exact timing could be reproducible and hence reliable. On the other hand, some computations do require precise timing. The most obvious example that we’ll see later is in the auditory system. But there are examples in the visual system too where timing matters a lot.

With that background, let’s now return to the eye and specifically look at what happens after light is sensed by photoreceptors.

The Retina

As the figure below illustrates, the retina consists of several layers of cells. The first layer contains the photoreceptor cells, and is followed by three layers: horizontal cells, bipolar cells, and amacrine cells which are not sensitive (directly) to light, but which perform computations to encode the image. The cells in these initial four layers have graded responses – not action potentials.

The fifth layer contains the ganglion cells which are quite different from the other cells in the retina since they each have an axon that goes to the brain and that transmit spikes. The axons
from each eye are bundled together to form the optic nerve. (Remember the blind spot? That’s where the gathering takes place.) That is how the optic nerve carries the encoded retinal image from the eye to the brain.

What information do the spikes from each retinal ganglion cell encode about the retinal image? The ganglion cells do not simply encode a pixel by pixel copy of the LMS photoreceptor image. Rather, they pre-process the image to make some aspects of the image more explicit. Indeed all layers of the retina contribute to this pre-processing. Rather than looking at the detailed circuits (low level hardware), let’s look at what information the circuits are computing and what problem they are solving.

**Spectral differences: Color Opponency**

The visual system encodes differences in the image. These can be differences between the L,M,S values at a point, or differences between LMS values of neighboring points, or differences in LMS values over time. After the LMS cones measure the light arriving at the retina at each location \((x, y)\), subsequent layers of cells in the retina compute these differences by taking linear combinations of the LMS responses. From many experiments over the years, neuroscientists have learned that the retina transforms the LMS measurement as follows: "L + M" measures the overall physical brightness in the medium and long wavelengths, "L-M" compares the long and medium wavelength response, and "L + M - S" compares the medium/long overall intensity to the short wavelength intensity.

Why are the L and M grouped like this? Why not group the M and S or do some other linear combination? The answer is to recall that the S cones are sampled relatively sparsely and as such they cannot contribute to the highly detailed spatial coding of the image. Rather, the detailed spatial coding is carried by the L+M channel, since L and M are both sampled finely. Also note that "L+M-S" should not be taken literally as if the weights were 1 for L and M and -1 for S. What matters is the sign of the weights, not the exact magnitude. As for the L-M channel, it is useful because the L and M spectral sensitivities are quite similar. So to get information about the details of a spectrum in the middle and long wavelengths, the difference between L and M responses is computed.

Measuring differences in cell responses is called opponency. L - M is called red-green opponency. L + M - S is called yellow-blue opponency. The reason \(L + M\) is called “yellow” is that if you mix together two lights that appear red and green, then you get a light that appears yellow. Color opponency is a very old idea and can be expressed in many ways. For example, in school you may have learned about primary colors and secondary colors and how to use them. (See ASIDEs below.) In vision science, the idea of opponency goes back to Hering in the late 1800’s. One of the key observations is that some colors seem to be in-between other colors, e.g. we perceive orange as reddish yellow, as if both red and yellow are both in orang. Similarly, we perceive cyan as blueish green, and we perceive purple as reddish blue. However, we cannot perceive a color to be blueish yellow, or reddish green. These observations are believed to be the direct perceptual equivalent of underlying opponency circuitry, namely computing the LMS differences mentioned above.
ASIDE (clarification of discussion in the lecture)

In art class, you learn about primary and secondary colors. The terms primary and secondary refer to paints (reflectances), not to lights. The idea is that primary color paints can be mixed together to form secondary color paints, but not vice-versa. The primary colors are red, blue, yellow. The secondary colors are green (blue + yellow paint), orange (red + yellow paint), and purple (red + blue paint). I should emphasize that mixing colored paints is not the same as mixing colored lights. For example, if you mix a red and green paint, you get a dark brown whereas if you mix a red and green light, for example by shining a red spot light and a green spot light onto a white paper, then you get a bright white spot.

Opponency (or opposite colors) comes up in art class too, but in a different way. In art, a primary color goes well with its "complementary" color, which is defined by mixing paints of the two other primary color paints e.g. blue paint goes well with orange paint which is mix of red and yellow paints.

Hue, saturation, value (HSV)

There are many ways to encode 3D color space, and one of the commonalities is that one distinguishes colors based on the relative amounts at different wavelength versus the overall total amount. In LMS theory, the former concerns the two difference channels and the latter concerns the L+M channel. If one thinks of a color circle, then the points on the circle define colors that are as pure as can be obtained and points in the interior of the circle (see right above) correspond to a mix of pure colors with white (or grey or black). By using a polar coordinate system for points in the circle and its interior, one can sweep out a range of colors. The angle or direction from the center of the circle defines the (maximally) pure color – often called the hue. The distance from the center is the purity – often called the saturation.

The third color dimension is often called the value, or lightness, or luminosity. (In fact, these terms all have specific technical meanings in other contexts, so if you look them up them prepare to be confused.) In the specific case of saturation of 0, i.e. the center of the color circle, the value dimension ranges from black to grey to white. Think of this third dimension as coming out of the page.
If you have used color pickers in MS paint or Powerpoint to select colors, then you will be familiar with these terms. I encourage you to experiment for a few minutes and see how RGB values gives rise to different HSV (or HSL) codes.

**Spatial differences: Lateral inhibition**

Because interactions in the retina are spatially localized, each ganglion cell can respond to only a spatially restricted region in the retina. The corresponding set of directions in the visual field (i.e. projecting out in the world) is called the *receptive field* of the cell. One of the central goals of studying ganglion cells is to identify the receptive field, and also to understand how the image in that part of the receptive field can affect the response. (Recall the linear McCulloch-Pitts model.) It turns out that the ganglion cells computing spatial differences in a very interesting way. They also compute temporal differences, but I won’t mention that today.

In 1950s, a researcher named Steve Kuffler measured spike trains from single ganglion cells of the cat retina. He recorded from single cells over time, while shining a tiny spot of light on the retina. He carried out these experiments in a very dark room, so that the only light shining on the retina was the tiny spot. He found that for each retinal ganglion cell there was small region of the retina that affected the spike rate of that cell, i.e. the receptive field.

Kuffler found many ganglion cells for which the firing rate increased when the tiny spot of light shone on a particular region. This is called *ON region* for that cell. He also found that surrounding this ON region was an annulus (ring) shaped region in which the tiny spot of light *decreased* the firing rate of the cell. This surrounding region is now called the *OFF region*. Because these cells were excited by light in the center and inhibited by light in the surround, these cells are called *ON center/OFF surround*.

Kuffler also found retinal ganglion cells that had the opposite property, namely there was a central round region in which the cell’s response decreased when the tiny spot of light was shone there, and a surrounding annulus region in which the response increased when light was shone there. These cells are called *OFF center/ON surround*.

Thinking again in terms of McCulloch-Pitts, one can assign weights to the different points in the receptive field and model the ganglion cell’s response as the sum of the weighted intensities over the receptive fields. Here it is simplest just to think of the L+M channel. (In Assignment 1, I ask you to think about difference channels too.)
Finally, because the intensities in the surround have the opposite effect as the intensities in the center, you can think of the image in the surround as inhibiting the response to the image in the center. This local spatial inhibition (or opponency) is often called lateral inhibition.

**DOG model**

One idea for achieving the center-surround effect is to suppose that there is one mechanism for excitement over a neighborhood and that the effect falls off with distance from the center of the receptive field, and that there is a different mechanism for inhibition that also falls off with distance. If the excitation were to come from a small neighborhood and be strong in that neighborhood and if the inhibition were to come from a larger neighborhood and be weaker over that neighborhood, then this would naturally lead to an ON-center/OFF-surround receptive field.

Rodieck and Stone (1965) proposed a specific model which was based on 2D Gaussian functions. The model uses a 2D Gaussian weighting function

\[
G(x, y) = \frac{1}{2\pi\sigma^2} e^{-(x^2+y^2)/2\sigma^2}
\]  

(1)

where a 2D Gaussian is just the product of two 1D Gaussians,

\[
G(x) = \frac{1}{\sqrt{2\pi}\sigma} e^{-x^2/2\sigma^2}
\]

\[
G(y) = \frac{1}{\sqrt{2\pi}\sigma} e^{-y^2/2\sigma^2}
\]

The 2-D Gaussian is *radially symmetric* in the sense that it only depends on the squared radius \(x^2 + y^2\). Note that this Gaussian is centered at \((0, 0)\) but more generally it could be centered at any \((x_0, y_0)\) by shifting. Also, note that we are ignoring the time dimension.

The difference of Gaussian function is then defined:

\[
DOG(x, y, \sigma_1, \sigma_2) = G_1(x, y) - G_2(x, y)
\]

and again it is centered at \((0, 0)\). Here 1 and 2 are the center and surround, i.e. \(\sigma_1 < \sigma_2\). This center would be ON-center and OFF-surround. To obtain OFF-center ON-surround, one would use \(\sigma_1 > \sigma_2\).

Finally, the response \(r(x_0, y_0)\) of a retinal ganglion cell whose receptive field is centered at \((x_0, y_0)\) is modelled as the inner product of the DOG with the image

\[
r(x_0, y_0) = \sum_{x,y} DOG(x - x_0, y - y_0) I(x, y)
\]
This linear model is a good first approximation, but it is now understood that the response is better described by including non-linearities too. Some of these are just obvious (and would have been recognized by in 1965): cells have a maximum firing rate, so if we were to increase the image intensity by an arbitrary scaling factor, one cannot expect the response to increase by the same scaling factor. Similarly, cell’s cannot have negative responses, so if the image is positive only in the negative part of the DOG, then the model wouldn’t make sense. One can model these non-linearities in a variety of ways, for example, by remapping \( r() \) using a sigmoidal shape curve, or by just half-wave rectifying by setting all negative response values to 0. See slides. As long as one has both an ON-center OFF-surround and an OFF-center ON-surround at each point, half wave rectification of both of these would be fine since it would carry all information about

**Cross-correlation**

To understand retinal processing of images, we want to know not just the response of a single cell to the images, but also also the responses of a family of cells that all have the same receptive field shape. One can define the cross correlation of two functions, in this case \( DOG \) and \( I \) by:

\[
DOG \otimes I(x, y) = \sum_{x', y'} DOG(x', y') \cdot I(x + x', y + y') = \sum_{x', y'} DOG(x' - x, y' - y) \cdot I(x, y).
\]

Think of the \( DOG \) as a template, and imagine sliding that template across the image. See slides. The formulas above make sense when the template is at \((x, y)\). Think of \((x', y')\) as a \((\Delta x, \Delta y)\) step away from the center position \((x, y)\) of a particular cell.

Finally, I briefly discussed how to compute cross-correlation in Matlab. One issue that comes up is that when you slide the template over the image, its unclear what to do when the template partly overlaps the image boundary. In that case, there will be pixels of the template that have no corresponding image pixels. How should the response of the template be defined in that case? One can just ignore that cases, and only consider the "valid" \((x, y)\) positions, i.e. where each pixel of the template corresponds to a point in the image. In that case, one will have responses for a subset of pixels only. Or, one can treat the image as having value 0 outside the pixels on which it was originally defined, and consider responses for all positions of the DOG template. This could either be all positions in which the center of the DOG template is at a pixel position in the original image (the 'same' condition below), or it could include any position in which the DOG template overlaps the image (the 'full' condition below). For the assignment, use 'same' condition since that way the output of \texttt{filter2} command will have the same size as the input image \( I \).

\begin{verbatim}
filter2( DOG, I , 'valid')  \% returns (N-(M-1)) x (N-(M-1)) result
filter2( DOG, I , 'same')   \% is the same as filter2( DOG, I )
filter2( DOG, I , 'full')   \% returns (N+M-1) x (N+M-1) result
\end{verbatim}

There is a specific function \texttt{xcorr2} which can be used for 2D cross-correlation, but I would just use the \texttt{filter2} function which has more options. Suppose \( DOG \) is an \( M \times M \) matrix (called a "filter" in linear systems language, as we’ll see later) and suppose \( I \) is an \( N \times N \) matrix that represents the image – just one channel, not RGB.