

Modeling the top neuron layers of the human retina

Laurens Koppenol (3245578)
23-06-2013
Universiteit Utrecht

Contents

Introduction.....	3
Background.....	4
Photoreceptors.....	4
Scotopic vision.....	5
Photopic vision.....	5
Horizontal cells.....	6
Bipolar cells.....	6
Further layers.....	7
Model.....	8
Back propagation learning.....	10
Parameters.....	11
Results.....	12
Untrained network.....	12
Cone layer.....	12
Mach band.....	12
Trained network.....	14
Number recognition.....	14
Conclusion.....	16
Sources.....	17

Introduction

Neural networks in computer science are models of the neural networks that exist in the human body, mostly in the human brain. They are generally used for one of two purposes. They can either be trained to perform specific tasks, for example character recognition, or they can be used to give insight in how are brain works, for example what neuronal structures are or are not capable of.

The human retina also contains a neural network. Using a computer model of the first 3 layers of the retinal neuron structure we will see some of the capabilities of this neural network. Specifically the foveal area of the retina has been selected for this model. The fovea has been selected because of the abundance of information and research on this part of the retina, and this area being the area most easy to relate to, as this corresponds to our point of focus.

Firstly there will be a brief description of the anatomy of the human visual system, until the visual cortex. Then we will focus on the modeled parts, being the first three neuron layers of the retina. The third layer is the first layer to clearly combine the signals from the former two layers and give interesting output. Finally we will see the results of the computer model. We will see that the Mach band effect can explained already with this simple model. We will also see that there are tasks this network cannot complete or learn while regular three-layer neural networks can.

A similar study was performed by Kunihiro Fukushima[1]. However Fukushima's neural network, named Neocognitron, is 'zoomed out'. While this thesis is about the first three layers of the retinal structure, Fukushima is taking into account the visual pathway from photoreceptor to grandmother cell. Therefore he is paying less attention to the detailed structure of the retinal layers. Another major difference is the self-organizing structure of the neocognitron. The neocognitron is able to reorganize while learning while my model is not.

Background

The function of our visual system is to process visual input, light, to signals to our brain. This happens in our retina. After the visual input is processed to neural signals the signal is preprocessed by multiple layers of neurons. We will discuss the biological properties of the layers modeled in our artificial neural network.

Light that enters the eye is refracted by the cornea, aqueous humor, lens and vitreous humor. The cornea, aqueous humor and vitreous humor have a fixed refraction while the lens can accommodate to form a sharp image on the retina, however these parts of the eye are irrelevant for our model and can be neglected. The light that is coming from the object is projected on the retina upside down and reversed left to right. In our model we will ignore this mirroring effect for convenience, and assume a sharp image is projected on the retina.

The retina consists of multiple layers of neurons. The first layer is the photoreceptor layer. The neurons in this layer convert light into electrical signals. The next layer, the horizontal cell layer, contains neurons that connect photoreceptor cells to combine their signal. Next is the bipolar cell layer. Bipolar cells measure the difference in electrical potential (signal) between horizontal cells and photoreceptor cells. Before leaving the eye via the optic nerve, the signal is further processed by ganglion cells and amacrine cells. We will however, only model the photoreceptors, horizontal cells and bipolar cells.

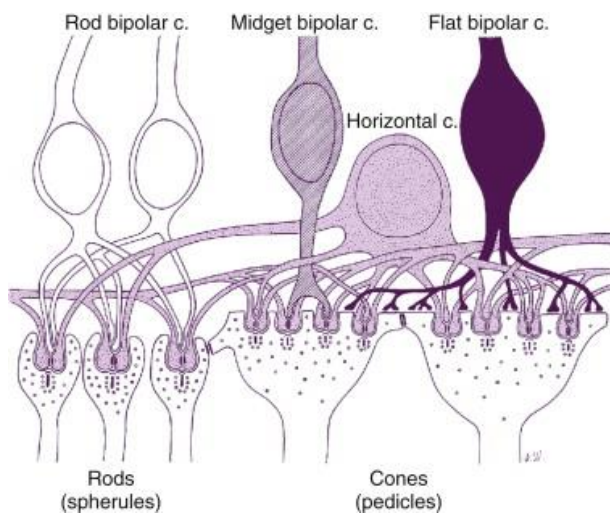


Figure 1. Layout of photoreceptors (cones / rods), horizontal cells and bipolar cells.

Photoreceptors

Photoreceptors convert light into electrical signals. Photoreceptors are sending a continuous signal to the next layer of neurons which is temporarily interrupted when the outer segment of the photoreceptor is hit by a certain amount of photons. Therefore photoreceptors signal incoming photons (light) as an interruption of current. Since we are interested in a 'snapshot' of neuron activity, not in motion and change of neuron activity over a certain time, we will model the interruption in signal as sending a signal.

There are two types of photoreceptors, rods and cones. The photopic system consists of the different types of cones while the scotopic system consists of rods.

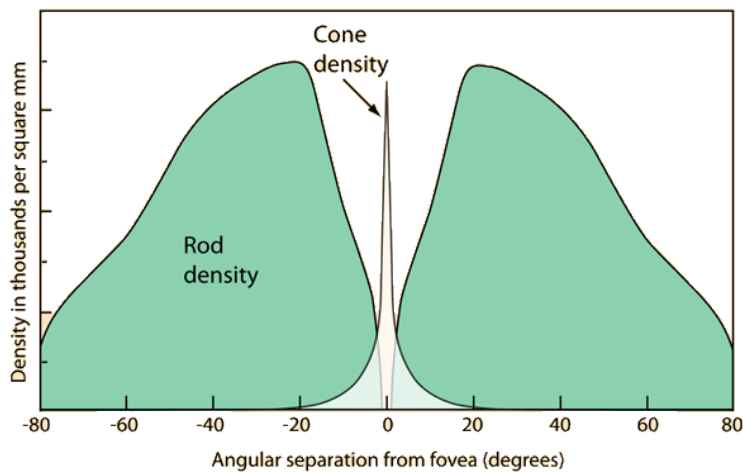


Figure 2. Rod and cone density on the retina

Scotopic vision

The scotopic system consists of rods. Rods are very sensitive to light, but cannot distinguish color. We will focus on the center of the retina, called the fovea, which has no rods. The rods spread can be seen in figure 2. (This is why it is harder to see stars when focusing on them). Because the fovea has the highest density of photoreceptors in total this is the most interesting area to model. Therefore the rods are neglected in our model.

Photopic vision

The photopic system consists of cones. There are 3 kinds of rods in the human retina, S-cones, M-cones and L-cones. Named after the wavelengths they respond best to, respectively small, medium and large wavelengths. Figure 3 shows the response of the cones to various wavelengths. The cones are sometimes referred to as respectively blue-, green- and red cones. As you can see this explains why we cannot see infrared light, which has a wavelength of 700 nm to 1 mm, or ultraviolet light, which has a wavelength of 10 to 380 nm.

The fovea is not only an area without rods, it is also an area without S-cones (blue). The ratio of M-cones versus L-cones is approximately 1:2, but this can greatly vary amongst individuals. The fovea contains approximately 125,000 photoreceptor cells per mm^2 . However this number is adjustable in the model to get favorable results depending on the input.

The photoreceptors are implemented as neurons connecting to pixels of the input image. Photoreceptors can be accessed by horizontal and bipolar cells.

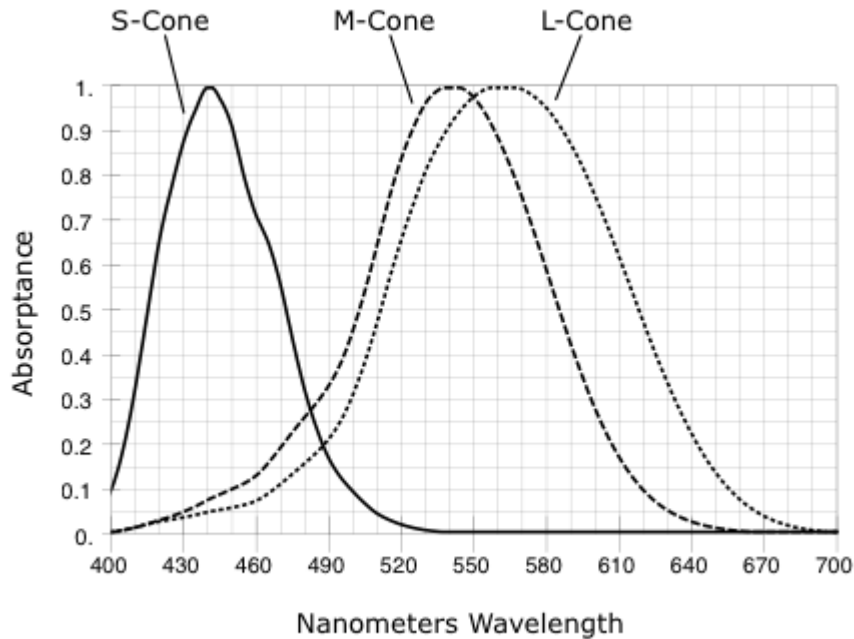


Figure 3. Sensitivity of cones.

Photoreceptor cells have graded potentials, meaning that increase of input leads to an increase in output, via a sigma function opposing threshold activation where the output is 0 until the input is increased to a certain value which changes the output to 1.[2]

Horizontal cells

The next layer of neurons in the retina is the horizontal cell layer. Horizontal cells are, as the name suggests, interconnecting photoreceptor cells on the same level, as can be seen in figure 1. Like photoreceptor cells, horizontal cells appear in various forms. Intuitively named HI, HII and HIII. However finding out their exact rate of appearance and difference in number of connections and/or activation rate would require extensive research and/or experiments. Therefore the horizontal cells will be implemented as one type connecting to all types of photoreceptors. Horizontal cells have graded potentials like the photoreceptor cells have. [3]

Horizontal cells are implemented as neurons connecting to multiple photoreceptor cells within range and can be accessed by bipolar cells. Both the range and number of connections are parameterized.

The number of connections (and thus occurrences) of horizontal cells varies greatly depending on location on the retina. This value is also parameterized to get favorable results depending on the input.

Bipolar cells

The final layer of modeled neurons is the bipolar cell layer. Bipolar cells connect to both horizontal cells and photoreceptor cells. Again there are various kinds of bipolar cells. For the sake of simplicity the various types have been neglected and modeled as one general type.

Bipolar cells have two connection types, center connections and surround connections as can be seen in figure 4. The center connections are connections to photoreceptor cells while the surround connections are connections to horizontal cells. ON-center bipolar cells give positive weights to connections to photoreceptors and negative weights to surround connections, OFF-center bipolar cells give negative weights to connections to photoreceptors and positive weights to surround connections. In any case the bipolar cell determines the difference between the center and the surround cells, and takes this as input for its activation function. [4]

Bipolar cells are implemented as neurons connecting to a number of photoreceptors and horizontal cells within range. Both the range and number of connections for both types of connections have been parameterized.

The number of connections (and thus occurrences) of bipolar cells varies greatly depending on location on the retina and the type of bipolar cell. Since types are not implemented this value is also parameterized to get favorable results depending on the input.

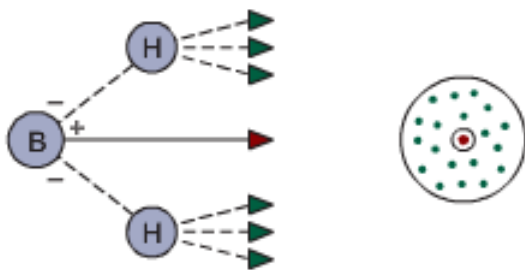


Figure 4. ON-center bipolar cell. Triangles represent photoreceptors.

Further layers

Between the bipolar cell layer and the optic nerve, where the signal leaves the eye to enter the brain there are more layers including the amacrine cell layer and retinal ganglion cell layer. As mentioned before these layers have not been implemented in the model but could be interesting to implement in further research.

Model

A modeled neuron, in this case either a photoreceptor cell (cone), horizontal cell or bipolar cell is called a node and consists out of the following:

- an array of connections with the previous layer. For the photoreceptor nodes these are connections to the pixels of the input image, for the horizontal nodes these are the connections to the photoreceptor nodes and for the bipolar nodes these are connections to the horizontal and photoreceptor nodes. This array also holds the weight of the connection. Formatted as $[[x,y,w],...]$
- The average of all output values from nodes in the previous layer this nodes is connected to, times the weights of these connections. This value is updated every round as we will see in the next section. Initial weights for all connections are 0.85-1.15, except for the bipolar cells connections to the photoreceptor cells, which have an initial weight of -0.4 to -0.5. The value of 0.85 has been selected to keep the output levels around between 0 and 1, because the sigmoid function has a steep slope between 0 and 1.
- An activation function, determining the output of this node based on the before mentioned average. For the horizontal and bipolar nodes this is a sigmoid function $f(x) = 1/(1+e^{-1*(x-0.8)})$, for the photoreceptor nodes this is, as shown in figure 5, different for each type. This sigmoid function is selected because it has a steep slope between 0 and 1.

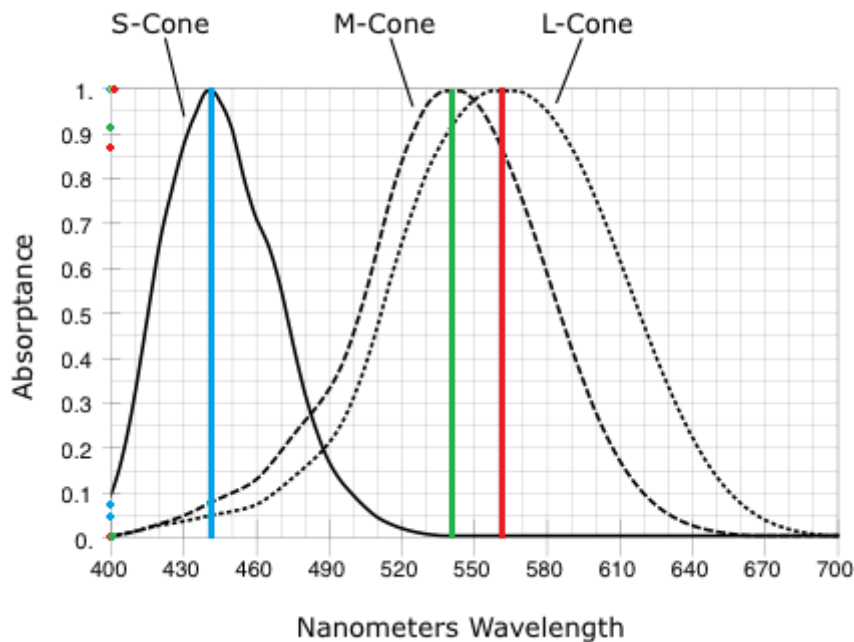


Figure 5. Activations per cone.

Every node has a location on the two-dimensional grid. Output values are shown as grey values, white being 1 and black being 0. Figure 6a shows a possible representation of the output of the photoreceptor nodes. Since not every layer has the same resolution of nodes, the following layer, the layer connecting to this layer is evenly distributed over the current layer, as shown on figure 6b. Every type of node has a range in which it can connect to nodes in the previous layer, and a minimum and maximum number of

nodes it will connect to. The example horizontal node in figure 6b has a range of 1, and has 4 photoreceptor nodes which it is connected to. In this example there are three times as many photoreceptor nodes as horizontal nodes in each direction. Connections are made randomly to nodes within range. A squared range is implemented to make it easy to connect to all nodes in the previous layer, whereas a circular range would make it harder to tweak the number of nodes in the neural network.

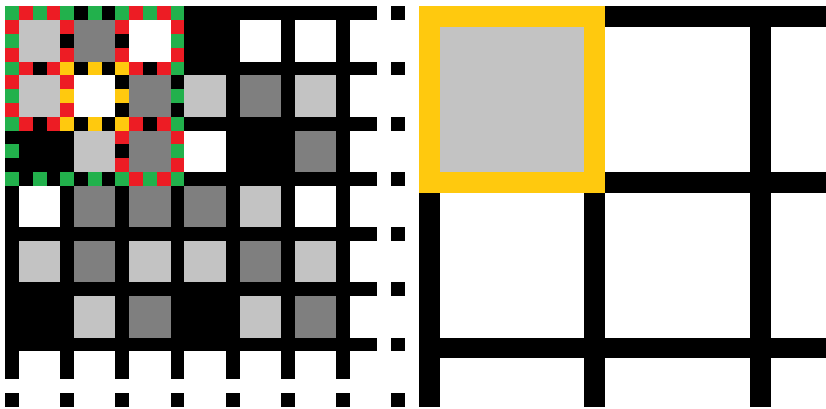


Figure 6a. Photoreceptor node layer Figure 6b. Horizontal node layer

The bipolar layer is constructed much in the same way, except for connecting to horizontal nodes and to photoreceptor nodes. Both connections are made in separate phases and are independent of each other.

The only layer that has a different connection algorithm is the photoreceptor layer. The input picture is divided in as many areas as there are photoreceptor nodes. The sum of all RGB-value of that area is turned into a 3-dimensional vector, and depending on the type of photoreceptor node the corresponding activation function is used to determine the output value. The R G B values are idealized as single peaks as shown in figure 1. The total input for S cones is equal to the B dimension of the vector since there is 0 response for wavelengths over 530nm. For M cones this is $0.08 * B + 1 * G + 0.88 * R$, and for L cones this is $0.05 * B + 0.92 * G + 1 * R$.

After initializing all nodes and connections the values are evaluated once in order of layer. First all photoreceptor output values are evaluated, then the horizontal output values and finally the bipolar output are evaluated. Even the untrained network shows some interesting results as we will see in the next chapter.

Back propagation learning

[5] Backpropagation learning is used as this model is a supervised feed forward network with extra connections between the input (photoreceptor) layer and the output (bipolar) layer.

The following algorithm is used to determine the error and update the weights in the network.

- Error is determined per node in the bipolar layer; $\text{error} = b_o (1 - b_o) (t_o - b_o)^2$ where t_o = correct output and b_o is the node's output. This error function has been selected as it is a standard error function for neural networks.
- For every node in the horizontal layer the error value is calculated as follow:
 - o Sum up the error values of all the nodes in the bipolar layer connecting to the current node multiplied by the weight of this connection
 - o Multiply this value by $h_o(1-h_o)$ where h_o is the output of the current node
- The weights of the connections from bipolar nodes to cone / horizontal nodes and the weights of connections from horizontal nodes to cones are updated using the follow update rule. This rule is known as the backpropagation learning rule.
 - o $\text{New weight} = \text{old weight} + \text{error of deeper node} * \text{output of higher node} * \text{learningrate}$, where bipolar nodes are deeper than horizontal nodes, which are deeper than cones.

Parameters

Resolution of photoreceptor layer: default 357 x 357, corresponds to 1 square mm on the fovea[6]

Ratio s:m:l cones: default 0:1:2

Resolution of horizontal layer: default 300 x 300

Ratio HI:HII:HIII

Different behavior of horizontal cells not implemented.

Resolution of bipolar layer: default 300 x 300

Ratio bipolar types

Different behavior of bipolar cells not implemented

Learning rate: default 0.01

Output function horizontal / bipolar layer: $f(x) = 1/(1+e^{-1*(x-0.8)})$

Output function photoreceptors

$$S: 1 * B$$

$$M: 0.08 * B + 1 * G + 0.88 * R$$

$$L: 0.05 * B + 0.92 * G + 1 * R$$

Where R G and B are the respective dimensions of the RGB vector mentioned earlier

Reach, minimum and maximum number of connections for every type of connections.

Types of connections are bipolar to horizontal, bipolar to cone, horizontal to cone.

Results

There are two ways to look at our model. We can either train the model on a training set to find out whether it is capable of specific tasks, for example character recognition. Or we can mimic natural situations and have our network mimic our experience of these natural situations.

Untrained network

Cone layer

First example is prompting the network with a red square. Taking an even colored red image as input, the photoreceptor layer shows the response as can be seen in figure 7. Even colored meaning every pixel has the exact same RGB-vector. The response shown is the output of a 28 x 28 photoreceptor layer. Other parameters are irrelevant since the network is untrained and horizontal and bipolar layer are not taken into account.

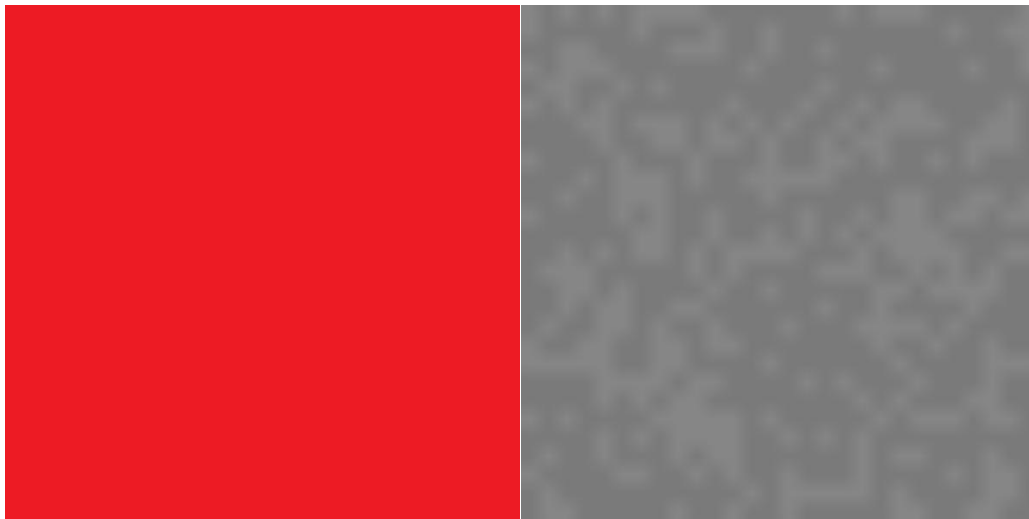


Figure 7. Red input image and photoreceptor response.

The 'noise' that can be seen on image 7 is caused by the different types of photoreceptors. The L-cones have a higher response to red light than the M-cones. The number of photoreceptors in this area corresponds to approximately 0.6% of a mm² on the human fovea.

Mach band

The following parameters are used to create the response images in figure 9:

Variables	
Learning	
learningrate:	<input type="text" value="0"/>
Dimensions	
Cones width/height:	<input type="text" value="300"/>
Horizontals width/height:	<input type="text" value="300"/>
Bipolar width/height:	<input type="text" value="300"/>
Horizontal nodes	
Reach:	<input type="text" value="5"/>
Minimum connections:	<input type="text" value="10"/>
Maximum connections:	<input type="text" value="20"/>
Bipolar nodes	
Reach (cones):	<input type="text" value="0"/>
Min. connections (cones):	<input type="text" value="1"/>
Max. connections (cones):	<input type="text" value="1"/>
Reach (horizontals):	<input type="text" value="5"/>
Min. connections (hors.):	<input type="text" value="10"/>
Max. connections (hors.):	<input type="text" value="20"/>

Figure 8. Parameters to recreate the Mach band effect.

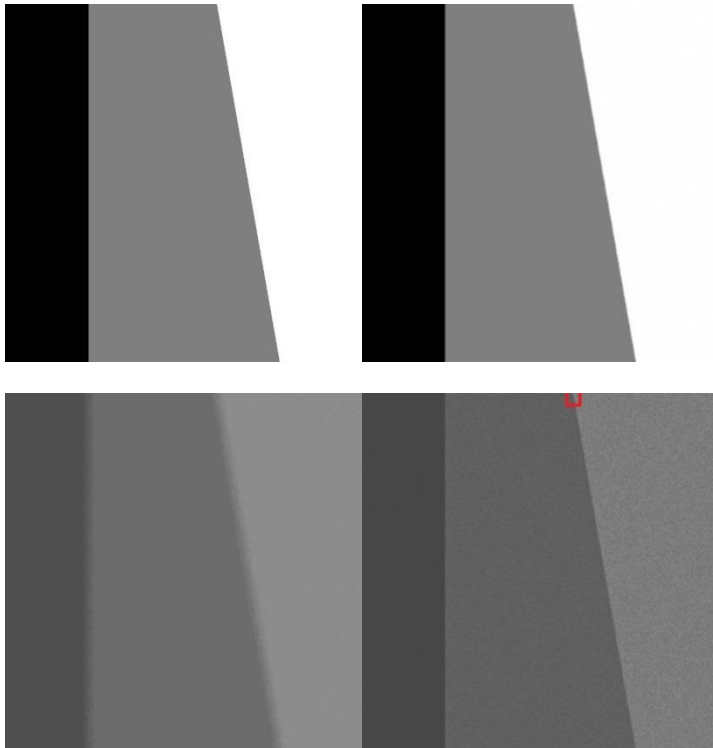


Figure 9. left to right top to bottom: Input image, cone layer, horizontal layer, bipolar layer

When describing the visual effect of above images, the cone output levels look very similar to the input image. The image is blurred due to the decrease of resolution. The original image has a resolution of 1000 by 1000 pixels, where there are only 300 by 300 cones. The horizontal layer is more blurred than

the cone layer, while the resolution remains the same; there are also 300 times 300 horizontal cells in this layer. Since the horizontal cells connect with multiple cone cells and average the result, a blurring effect can be seen. Finally the bipolar output shows the Machband effect. When zoomed in, the bright areas show a brighter area at the contrast line, whereas the dark areas show a darker line at the contrast line. Figure 10 shows the red area from figure 9 enlarged. There is a light grey area nearby the edge that is lighter than the area on the right of the image.

As explained before, the bipolar nodes work with an off-center cone area. As can be seen in the parameters, the bipolar cells connect with 1 cone directly in front of the bipolar cell (radius 1) and with up to 20 horizontal cells, which can connect to up to 20 cones. Because the input from the cone cell is subtracted from the average input from the horizontal cells there is an edge-detecting effect. This result has been accomplished without learning.



Figure 10. Zooming in on the bipolar layer

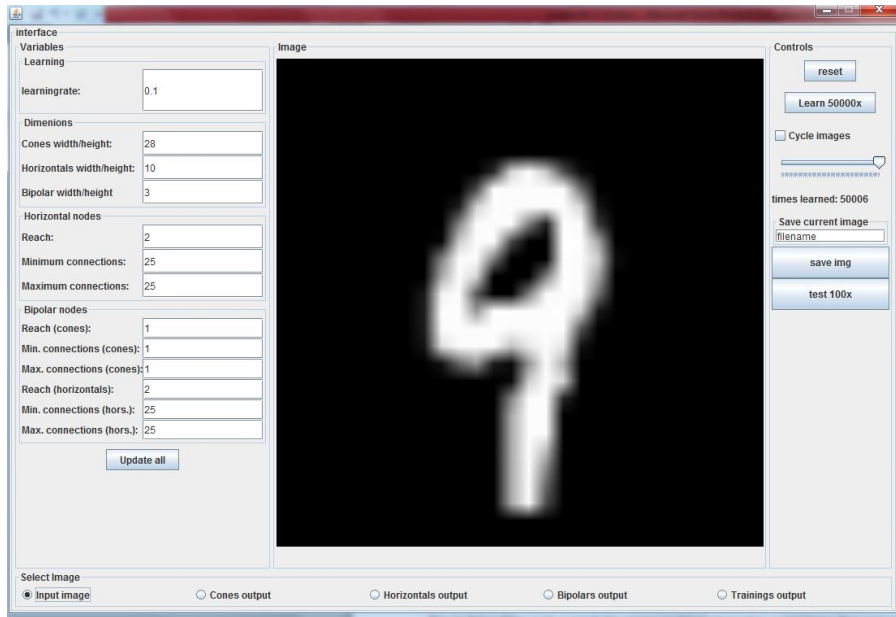
Trained network

Number recognition

Using the MNIST database[7] the network is learned to recognize characters. The database consists of 60,000 training images and 10,000 test images, all 28 by 28 pixels. The images represent numbers from 0 to 9, including 0 and 9.

Result of the network is presented in a 3 by 3 bipolar layer, where the node with the highest value represents the guess of the network, (0,0) being 1, (3,3) being 9 and all black being 0. After learning

50006 times with parameters as shown in figure 11 the network guessed 11 out of 100 characters from the testset, which is terrible.



Conclusion

We modeled the retinal neuron structure as a feed forward back propagation neural network that allows direct connections from the input layer to the output layer, and where nodes can only connect to other nodes within range.

The model is able to detect edges without learning if the parameters are entered correctly. On the other hand is the network unable to find relations between input nodes that are not in range of the bipolar nodes. Considering the human visual system this is no surprise as shape or even object recognition happens in the visual pathways.

This model could be easily expanded to a larger model, implementing various types of neurons and implementing more layers. However, since the limited time for this thesis, working out further layers or extensive further experimenting was not possible. Also making the model more realistic required more literature research. Further papers on this subject should focus on distinguishing different kind of nodes and implementing more realistic parameters.

Sources

[1] Fukushima, Kunihiko, 1980, Neocognitron: A Self-organizing Neural Network Model for a Mechanism of Pattern Recognition Unaffected by Shift in Position, *Biological Cybernetics* 36, p. 193-202.

[2] Wolfe, Jeremy M., et al., 2012, *Sensation & Perception*, 3rd edition, Sinauer associates, Inc. p. 32-41.

[3] Remington, Lee Ann, 2012, *Clinical Anatomy and physiology of the visual system*, St.Louis : Elsevier/Butterworth-Heinemann, chapter 4.

[4] Schwartz, Steven H., 2010, *Visual perception: a clinical orientation*, New York: McGraw Hill Professional, p. 118

[5] Chakraborty, R.C., 2010, *Back Propagation Network: Soft Computing*, http://www.myreaders.info/03_Back_Propagation_Network.pdf (June 23, 2013)

[6] Jonas, Jost B., et al., 1992, Count and density of human retinal photoreceptors, *Graefe's Arch Clin Exp Ophthalmol* 230, p. 505-510

[7] LeCun, Yann, et al., The MNIST database of handwritten digits, <http://yann.lecun.com/exdb/mnist/> (July 1, 2013)