



Available online at [www.sciencedirect.com](http://www.sciencedirect.com)

SCIENCE @ DIRECT<sup>®</sup>

ELSEVIER

Neurocomputing 58–60 (2004) 827–833

---

---

NEUROCOMPUTING

---

---

[www.elsevier.com/locate/neucom](http://www.elsevier.com/locate/neucom)

# A temporal trace and SOM-based model of complex cell development

Thomas J. Sullivan<sup>a,b,\*</sup>, Virginia R. de Sa<sup>a</sup>,

<sup>a</sup>Department of Cognitive Science, University of California, San Diego, 9500 Gilman Dr., MC 0515, La Jolla, CA 92093, USA

<sup>b</sup>Department of Electrical and Computer Engineering, University of California, San Diego, CA 92093, USA

---

## Abstract

An activity-dependent model of complex cell development is described. The learning rules are a combination of a temporal trace rule, a Hebbian rule, and a self-organizing map (SOM). They are local, unsupervised, and biologically realistic. The only input that is assumed is spontaneous retinal wave activity. The results show complex cells with phase invariance and orientation tuning that form an organized map.

© 2004 Elsevier B.V. All rights reserved.

**Keywords:** Complex cell; V1; Trace rule; Self-organizing map

---

## 1. Introduction

Complex cells are an integral part of visual object recognition within the brain. Understanding how complex cells develop is important for several reasons. V1 is a crucial part of the stream of visual processing that recognizes visual objects. Also, visual object recognition in the cortex is arranged in a hierarchical fashion with each block within the hierarchy having a similar organization. For this reason, the study of the development of one of these pieces of the system is in many ways the study of the entire system.

---

\* Corresponding author. Univ. of California, San Diego, Cognitive Science, 9500 Gilman Dr., MC 0515, La Jolla, CA 92093, USA. Tel.: +1-858-822-2402.

E-mail address: [tom@sullivan.to](mailto:tom@sullivan.to) (T.J. Sullivan).

## 2. The model

The model described in this paper is inspired by Földiák's complex cell model [6]. Like Földiák's model, there are two layers of neurons that are fully connected. The first layer contains simple cells, while the other contains the neurons that will develop into complex cells. The network is fully connected so that each neuron in the second layer receives a weighted input from each neuron in the first layer. The simple cells are arranged in a two-dimensional grid. Each grid location contains several neurons (16 is a common example) which all respond to stimuli within the same receptive field. The neurons corresponding to one grid location each respond to lines of a particular orientation. The complex cells are arranged on a similar two-dimensional grid, but have only one neuron per location. All the complex cells are presumed to respond to the same receptive field (this is a bit unrealistic since simple cells and complex cells have receptive fields of similar size).

The simple cells in this model are essentially line detectors. If a line of the preferred orientation is located in the neuron's receptive field, the neuron will respond maximally. Like real simple cells, the virtual neurons will respond to lines of various orientations. Each simple cell has an associated orientation tuning curve. It is created with an equation which is based on the behavior of real simple cells [3] (see Fig. 5).

The learning rule is a combination of three learning paradigms: Hebbian learning [8], a temporal trace rule [6], and self-organizing maps (SOM) [9,10]. Hebbian learning was chosen because various studies [2] have found that the weight changes between neurons are a result, at least in part, of correlations in their activations. This idea is expressed in this model by the following equation:

$$\Delta w_{ij} = \beta \bar{y}_i (x_j - w_{ij}),$$

where  $\Delta w_{ij}$  represents the change to a weight at a particular time step.  $\beta$  is a parameter that controls the rate of learning.  $\bar{y}_i$  is the magnitude of activation of the complex cell  $i$ .  $x_j$  is the magnitude of activation of the simple cell  $j$ .  $w_{ij}$  is the absolute magnitude of the weight before the update. In the equation the input activation multiplied by the output activation gives a measure of the correlation between the two neurons. Subtracting the absolute weight value keeps the weight from increasing without bound.

The temporal trace rule is described in Földiák's work [6]. The motivation is that visual stimuli that are seen close together in time are likely generated by the same object. For example, if we are looking at a book at time = 0 s, it is quite likely that the object we see at time = 30 ms is that same book. The trace rule models this by slowing down the response time of the neurons in the complex cell layer. If a complex cell has the biggest response within the layer and wins a competition, it is slow in turning off. This neuron will therefore have an increased chance of winning the next competition. In this way, input stimuli that are grouped close together in time will have a good chance of being grouped together in the complex cell layer representation.

Physiologically, the temporal trace could arise from the dynamics of feedback loops. A system whose output feeds back to its inputs will have temporal dynamics that are slower than that of its internal components. Consider, for example, the loop formed by feedback connections from V2. The complex cells of V1 feed into the input layer

neurons (layer IV) of V2. The output layers of V2 (layers VI and II/III) have connections back to the complex cells. This kind of feedback loop could be responsible for a temporal trace on the V1 output, and therefore responsible for the learning of spatio-temporal correlations in V1. The Elman net is an example of how a feedback loop can lead to temporal sequence learning [4].

The equation that implements the temporal trace rule in this model is

$$\bar{y}_i^{(t)} = (1 - \delta)\bar{y}_i^{(t-1)} + \delta y_i^{(t)},$$

where  $\bar{y}_i^{(t)}$  is the trace value for complex cell  $i$ . It is the sum of the previous trace value,  $\bar{y}_i^{(t-1)}$ , for this neuron, and an instantaneous activation based on the visual stimuli,  $y_i^{(t)}$ .  $\delta$  is a parameter with a value between 0 and 1 that controls how much memory to retain. According to this equation if a complex cell has a large activation in one time step, that increases its trace value for the next time step (thus increasing its chance of winning the next competition). This will lead to the temporal grouping as discussed above.

The calculation of the instantaneous activation,  $y_i^{(t)}$ , comes from the SOM part of the learning rule. When each input stimulus pattern is presented, activity is propagated to the complex cell layer through the weights using a dot-product calculation. A competition then takes place on the complex cell layer. The complex cell that is the most activated is called the winner. The competition is not winner-take-all, though. The winning neuron will have the highest instantaneous activation, but its neighbors will also be activated. The equation for the instantaneous activation of a second-layer neuron,  $i$ , is

$$y_i^{(t)} = \alpha e^{(-\|r_c - r_i\|^2)/2\sigma^2},$$

where  $\|r_c - r_i\|^2$  is the physical distance between the winning neuron  $c$  and neuron  $i$ .  $\sigma$  controls how many neighbors a winning neuron influences.  $\alpha$  is the magnitude of the instantaneous activation for the winning neuron. This SOM part of the learning rule is intended to model the effects of lateral connections on activity-dependent development. For examples of SOM's used in modelling lateral connections, see the excellent review by Erwin et al. [5]. Also, it has shown how the SOM activities can be implemented with lateral connections [12].

The model is initialized with small random weights between the two layers of neurons. A series of activity wave line stimuli are then swept, one at a time, across the neurons of the first layer (see Fig. 1). In the simulations, time was discretized. At each time step the activity wave front was advanced perpendicular to its orientation by a constant distance (in our simulations, this was 1.2 times the distance between two grid locations). The neurons that were passed over by the activity wave front would become active for one time step if the simple cell's orientation matched that of the activity wave stimulus. This method effectively integrates the responses of a subset of neurons over a small amount of time. After the activity wave sweeps across the entire neural field, a new wave of a random orientation begins.

This input activity is used as a model of pre-eye-opening retinal waves [7]. These are waves of activity that sweep across the retina in random directions. It is presumed that these waves of retinal activation are the input that the visual system uses to develop

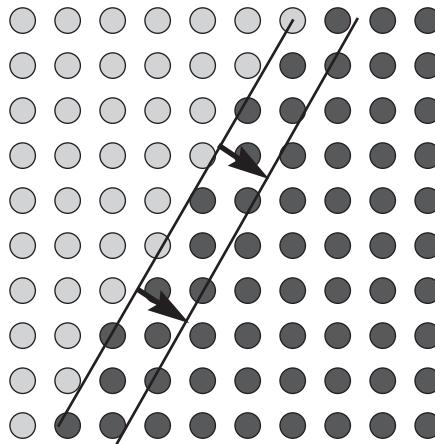


Fig. 1. Illustration of advancing activity front.

the simple and complex cell behaviors. In this model, the simple cells have already developed. (While the physiology literature is unclear on whether complex cells in layer V develop before simple cells, it is clear that in the cat simple cells are orientation selective before input reaches the complex cells in layers II/III [1].) It is assumed that this network operates on inputs that correspond to a small area of the retina. The wave will therefore always look as if it has no curvature and is travelling perpendicular to its orientation.

### 3. Results

The proposed model is complete enough to produce neuron behavior that is similar to complex cells. One of the most important properties that is captured is phase invariance. After training, the cells on the second layer become responsive to a line of a particular orientation that is placed anywhere in the simple cell layer. This is unlike the simple cells, which are responsive to a line in only one particular place, as well as one particular orientation.

To show the phases and orientations of maximal response, diagrams in Földiák's style [6] are used (see Fig. 2). On the left are lines in the orientation of maximal response of the second layer neurons. This is produced by presenting the input layer with lines of various orientations centered about the middle and for every second layer neuron recording the orientation that caused the largest activation. For the system in Fig. 2, for example, the line orientation that the top-left neuron is most activated by is about  $80^\circ$  ( $0^\circ$  is horizontal and clockwise rotations are measured as positive numbered degrees.) This example is taken from a network that has not been trained. The cells of the second layer appear to prefer random orientations. The grid of lines on the right of the figure is a representation of the weights between the simple cells and

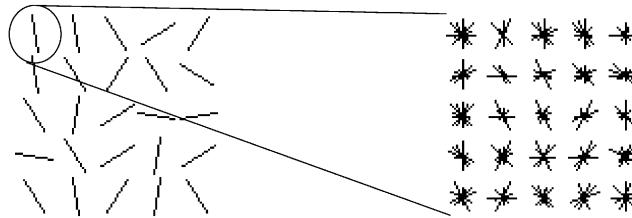


Fig. 2. Example of second-layer responses before training.

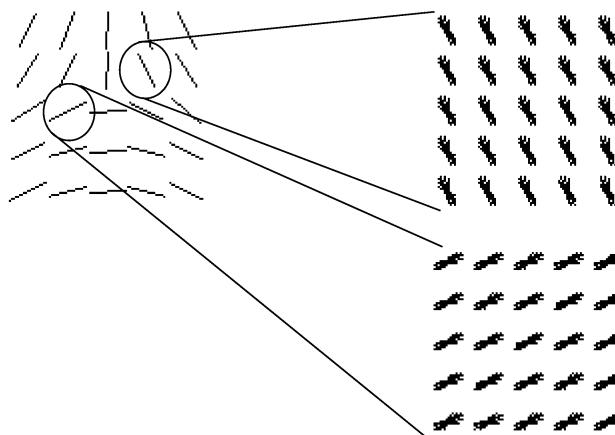


Fig. 3. Complex cells and their input weights after training.

one neuron on the second layer. In this example, there are eight simple cells per grid location, each with a preferred orientation. In the figure, the orientation of a drawn line corresponds to the orientation that the simple cell responds to maximally. The length of a drawn line indicates the strength of the weight between the simple cell which that line represents and one neuron of the second layer. The line lengths are normalized such that the largest weight from a grid location has a standard length. The other line lengths from that grid location are computed as the standard length multiplied by the ratio of the corresponding weight to the largest weight. In this way we can see the relative strengths of the weights within the grid locations. For the example in Fig. 2, the drawn lines within the grid on the right represent the weights that are connected from all the simple cells to the second-layer neuron in the top-left corner. It is obvious now why the second layer neurons are arranged randomly—the weights from the simple cells are completely random. Even though the orientation of maximal response is shown, there really is no strong preference for any direction.

Fig. 3 shows the same type of diagram for the model after training. The input weights corresponding to two different second-layer neurons are shown. It is obvious that the weights have similar values for each grid location. This means that all the



Fig. 4. Examples of complex cell organization after learning.

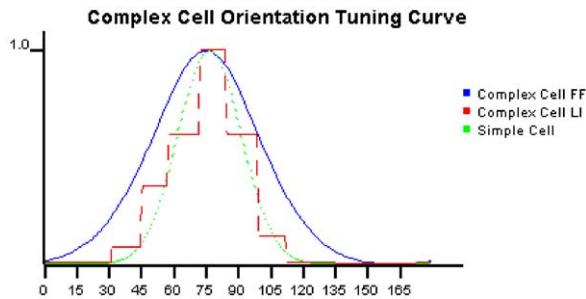


Fig. 5. Example of complex cell orientation tuning curves. Normalized response (on the *y*-axis) versus orientation (in degrees on the *x*-axis). The solid curve corresponds to the sum of the feedforward input to the cell. The curve with long dashes is the response after competition on complex cell layer. The curve with fine dashes is a simple cell response included for comparison.

grid locations respond to the same type of input. These second-layer cells will respond maximally to a line of a particular orientation located anywhere in the simple cell layer. Thus, phase invariance has developed.

This model of complex cell development also accounts for another key aspect of cortical neurons. In Fig. 3 we see that the maximal response orientations of the complex cells are arranged in an orderly fashion. This is consistent with how complex cells (and simple cells for that matter) are arranged in V1. The exact pattern that is formed depends on the random weights and random line orientations that are swept across the network. A few examples of complex cell maps resulting from various simulations are shown in Fig. 4. In all cases it appears that an orderly arrangement has resulted from the learning process.

A good way to see the preferred orientation of a developed complex cell is to plot its orientation tuning curve. This is done by presenting the simple cell layer with lines of various orientations and recording the responses of a complex cell. In Fig. 5, a typical example of a complex cell orientation tuning curve is shown. The solid curve corresponds to the sum of the feedforward input to the cell. The curve with long dashes is the response after competition on complex cell layer. The curve with fine dashes is a simple cell response included for comparison.

It is informative to see the influence of feed-forward activation on the complex cell behavior, but this is not what is measured in physiology experiments. Output

behavior is measured in those experiments, which is influenced by both complex cell competition and feed-forward activation. Lateral inhibition sharpens the tuning curve, since the complex cells which win the output competition suppress the response of the others. This idea of lateral inhibition affecting the shape of the tuning curve [11] can be seen in Fig. 5.

Our model predicts that interfering with the retinal waves during development will impair the development of complex cell receptive fields. This experiment has not yet been done, but if performed the results would either support or disprove this model of development.

## Acknowledgements

This material is based upon work supported by the National Science Foundation under NSF Career Grant No. 0133996.

## References

- [1] K. Albus, W. Wolf, Early post-natal development of neuronal function in the kitten's visual cortex: a laminar analysis, *J. Physiol.* 348 (1984) 153.
- [2] T.V.P. Bliss, T. Lomo, Long-lasting potentiation of synaptic transmission in the dentate area of the anesthetized rabbit following stimulation of the perforant path, *Nature Neurosci.* 232 (1973) 331.
- [3] P. Dayan, L.F. Abbott, *Theoretical Neuroscience*, MIT Press, Cambridge, MA, 2001.
- [4] J.L. Elman, Finding structure in time, *Cognitive Sci.* 14 (1990) 179.
- [5] E. Erwin, K. Obermayer, K. Schulten, Models of orientation and ocular dominance columns in the visual cortex: a critical comparison, *Neural Comput.* 7 (1) (1995) 425.
- [6] P. Földiák, Learning invariance from transformation sequences, *Neural Comput.* 3 (1991) 194.
- [7] L. Galli, L. Maffei, Spontaneous impulse activity of rat retinal ganglion cells in prenatal life, *Science* 242 (1988) 90.
- [8] D. Hebb, *The Organization of Behavior*, Wiley, New York, 1949.
- [9] T. Kohonen, Analysis of a simple self-organizing process, *Biol. Cybern.* 44 (1982) 135.
- [10] T. Kohonen, *Self-Organizing Maps*, 3rd Edition, Springer, Berlin, 2001.
- [11] K.D. Miller, D.J. Pinto, D.J. Simon, Processing in layer 4 of the neocortical circuit: new insights from visual and somatosensory cortex, *Curr. Opin. Neurobiol.* 11 (2001) 488.
- [12] J. Siros, R. Miikkulainen, Topographic receptive fields and patterned lateral interaction in a self-organizing model of the primary visual cortex, *Neural Comput.* 9 (1997) 577.