10 A Three-Dimensional Neural Network Architecture

Evangelia Micheli-Tzanakou, Timothy J. Dasey, and Jeremy Bricker

10.1 INTRODUCTION

The idea behind the presented architecture was to create a pattern recognition system using neural components. The brain was taken as a model, and although little is known about how pattern recognition is accomplished, much more is known about the cells that comprise the earliest levels of processing and analyzing the features of an environment most directly. By constructing cells with similar properties to the biological cells, we may gain an advantage in information conservation and proper utilization of neural architectures. The most important characteristic of these cells is their receptive field (RF). With this in mind, we could search for an adaptive mechanism that, by changing connective strengths, could give the desired RFs. Therefore, since we will know what information the algorithmic components are providing, when a method is found that provides the desired cell types, we may be able to trace back via the algorithm to see what information the neurons give.

10.2 THE NEURAL NETWORK ARCHITECTURE

The architecture chosen was that of a hierarchy of two-dimensional cell layers, each successive layer more removed from the environment (Figure 10.1). The first layer receives inputs from the external world and all other layers from the preceding layers. In addition, the cells may receive lateral connections from other neighboring cells within the same layer, depending on the particular choice of the architecture. The interlayer feed-forward connections are chosen so that a cell feeds its connections onto a neighborhood of cells in the lower layer. This neighborhood may have definite bounds so that all cells within it make connections, or it may have indefinite bounds in which the probability of a connection decreases as a Gaussian with distance.

The component cells themselves choose their outputs based on a weighted sum of all inputs passed through a function σ, such as

$$O_i(t) = \sigma\left[\alpha_i^* \Sigma_j C_{ij}^* O_j(t-1)\right] \tag{10.1}$$

where $O_i(t)$ is the output of neuron i at time interval t, C_{ij} are the connection strengths, bounded from [–β,β] where β is usually 1.0, and α is a constant. In the simulations, σ is usually a sigmoid of the form

$$\sigma(x) = 0.5^* a^* \left(1 + \tanh\left(b^* x^* c\right)\right) \tag{10.2}$$

where *a,b,c* are constants which fix the maximum value, steepness, and bias of the sigmoid, respectively. However, if we wish to allow the inhibitory components of the RF to be used by subsequent layers, then the sigmoid function must have a non-zero firing level for those negative inputs. This suggests the use of a spontaneous firing activity for all neurons. An additional requirement needed to keep the neurons useful and "responsive" is to keep that neuron from being pushed too far into the saturation level. If that occurs, input deviations will not be sensed well, if at all. Since each neuron receives several inputs, it is easy for this to occur. To prevent it from happening, α is usually chosen equal to the reciprocal of the number of connections to neuron *i*, so that the neuron simply passed a weighted average of the inputs through the sigmoid.

10.3 SIMULATIONS

A simulation usually consists of a sequence of presentations of random input patterns to the first layer and a learning rule imposed on the connections by analysis of the firings of the neurons. A random input was chosen so as to prevent the cells from being biased towards any specific environmental feature. Since neighboring inputs are uncorrelated, first layer cells that receive their influences are expected to have synapse patterns that would similarly wander aimlessly in the learning process. The first layer provides a spatial average of the overlying inputs. Since neighboring cells have the greatest overlap in their neighborhoods, they tend to have firing patterns which are most similar. This would cause cells in layer 2 to have synapses that originated from nearby cells to want to be alike. The actual training of the connections can be done in different ways.

I) Synapses can be changed based on a variation of the Hebbian rule[1] as follows:

$$C_{ij} = \delta^* O_i^* O_j \tag{10.3}$$

where δ is a small positive constant. Due to the correlation between neighboring level 1 cells, the synapses to the cells in later layers would tend to want to be all alike without additional constraints. In order to guarantee both positive and negative synapses to every cell, an additional "resource" constraint is imposed, which takes the form of

$$\Sigma_j C_{ij} = 0 \tag{10.4}$$

The third restriction is a bounding of the connections to the interval [–1, 1]. A synapse is allowed the freedom to switch from positive to negative and vice versa. This is not expected to alter the main results but only to prevent many of the synapses from disappearing with zero strength. Convergence usually occurs within 1000-5000 iterations, although faster convergence can be achieved with larger δ. Usually the final state of the synapses is at either the excitatory or the inhibitory limits.

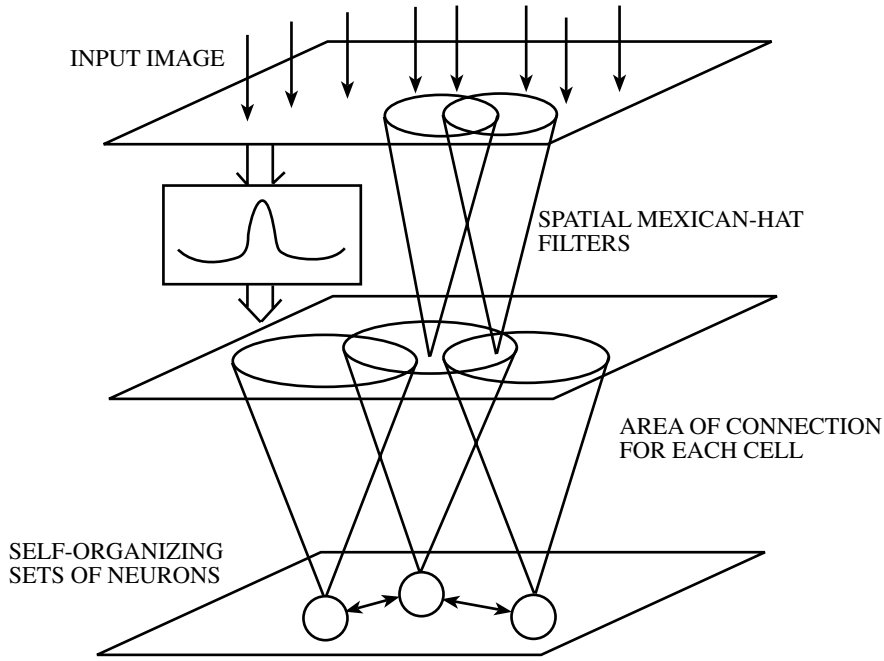

FIGURE 10.1 A schematic representation of the neural architecture.

10.3.1 Visual Receptive Fields

A network was created with three layers and 128 cells per layer. A square stimulus was assumed with 32×32 size. The maximum distance that these cells can affect is a radius of r, with minimum weight –1 and maximum weight values of +1. The network had a total of 7071 connections. In the training mode, the minimum stimulus value was assumed to be zero and the maximum equal to 10. No noise was imposed on the system. The results obtained show the emergence of cells with edge-type RFs in layer 2 (Figure 10.2a). The orientation of the edge appears to be totally arbitrary, even between neighboring cells. In layer 3, these edge cell RFs often conflict to give RFs which have oblong centers and surrounds of the opposite polarity, but many times these centers draw to the edges with further learning. Thus the final RFs often look like an elliptical center touching the outside of the field, mostly surrounded by a horseshoe-shaped region of opposite polarity (Figure 10.2b).

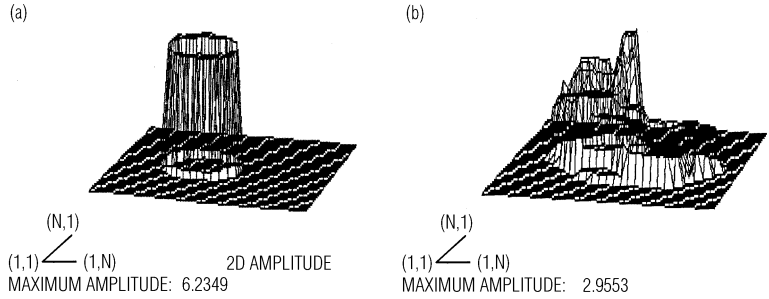

Figure 10.2 Receptive field characteristics for the neurons described in the text. (a) RF of layer 2. (b) RF of layer 3. Notice the center-surround organization of layer 2 and the elongated character of layer 3.

Figure 10.3 shows the results from a similar network, except that the minimum weight value is –0.5, i.e., less inhibitory effects. Notice that excitation spreads more and that the maximum amplitudes are much larger. Also notice that the layer 3 RF is much longer than the one in layer 2.

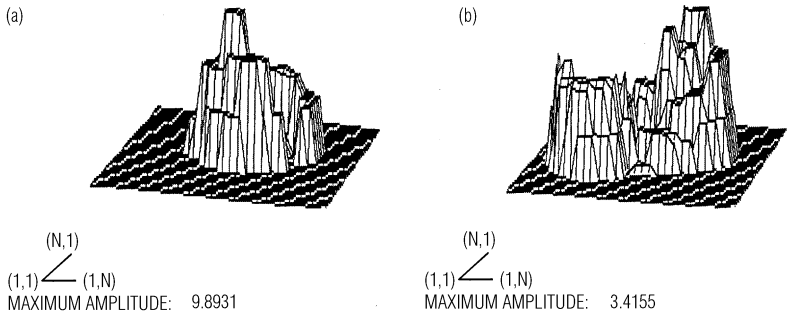

FIGURE 10.3 Receptive field organization for layer 2 (a) and layer 3 (b) when the inhibitory effects are less than in Figure 10.2. Compare the amplitudes and the spread of the RFs to those of Figure 10.2.

In the frequency domain, these RFs show more fine tuning as we move to deeper layers of the system. Figures 10.4 and 10.5 represent the power spectra of Figures 10.2 and 10.3, respectively. Also, notice that the edge effects are more obvious in the spectra of layer 3.

II) The wider the variance of the firing rate of the cells the more information the cells can carry. With such a supposition we can use an optimization routine to find the values of the synapses to a cell such that the variance in the firing rate of the cell is maximized. The optimization system is a variation of the ALOPEX process.[2] In this process two random connection patterns can be presented, and the variance (V) of the cell output is estimated with a number of random input patterns. Since we want the pattern of connection strengths to affect the variance and not the strength of the connections themselves, the variance can be modified as

$$V_i = \left[(1/N)^* \Sigma_j \left(O_i^j - O_i^{ave}\right)\right] / \left(\Sigma_j C_{ij}\right) \tag{10.5}$$

The connections are then changed based on the relation between the last change in connections and the last change in the variance, with an added noise term to prevent local minima as follows:

$$C_{ij} = \beta^* \left(C_{ij}(t) - C_{ij}(t-1)\right)^* \left(V_i(t) - V_i(t-1)\right) + \text{ noise term} \tag{10.6}$$

Amazingly, with this modification, the same edge sensitive cell RFs emerge after only about 100 iterations and remain the same until about 400 iterations. This shows that the combination of Hebb's rule and ALOPEX is something desirable. It might also mean that the way in which the architecture of the network is set up biases them toward neurons with edge detection capabilities. Work by others[3] has indicated that certain forms of the Hebb rule can be used to perform principle component analysis, a variance maximization of sorts. In addition, both feed-forward and feedback connections are used, with feedback having a wider connective neighborhood than the feed-forward connections. All connections are variable. If the inhibitory connections are spread over a much wider area, they tend to cancel the excitatory influence, making the Hebb changes ineffective. In future work we will include feed-forward connections of cells with a Gaussian distribution, and with inhibitory connections and excitatory connections having a different spatial standard deviation. The present number of maximum synapses allowed does not give us the ability of obtaining statistical significance for initial random strength generation.

III) Both feed-forward and feedback connections can be used, with the feedback having a wider connective neighborhood than the feed-forward connections.

IV) Lateral connections on each layer are allowed and used, thus adding an extra feature of similarity to the biological system.

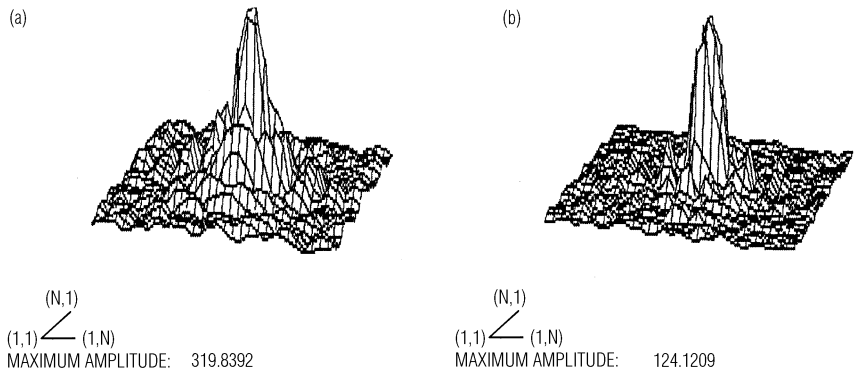

FIGURE 10.4 Power spectrum of the RF in Figure 10.2. (a) layer 2, (b) layer 3. Notice the fine tuning in layer 3.

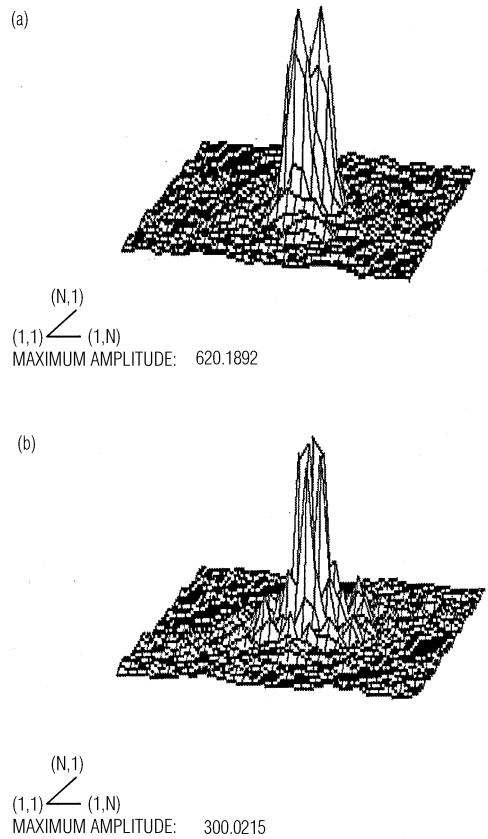

FIGURE 10.5 Power spectra of RFs in Figure 10.3. (a) layer 2, (b) layer 3. Compare with Figure 10.4. The edge effect is much more pronounced.

If each input signal value is thought of as a dimension in parameter space, any particular input will comprise a point in that space. The synapses of a neuron can then be thought of as describing a vector in the same space and the output of the neuron as the projection of the input point onto the synapse vector. If the choice of the synapses is initially random, chances are that the projections from many different inputs will lie close to one another, giving the neuron a response profile. Consider this to be the response profile of a neuron before optimization. In order to better distinguish between inputs, the synapses should be changed so that more of the neuron range can be utilized. An intriguing choice is for the neuron to perform a type of principal component analysis (PCA) (Karhunen-Loève feature extraction). Principal component analysis may be approximated by a search for the vector (described by the connection weight values), which maximizes the variance of the cell firing level. The choice of this property may serve to partition the input space into recognizable categories at the output. This analysis approximates the Karhunen-Loève search for the eigenvector of the maximum eigenvalue. For layers of neurons

that have a large amount of information with near neighbors, the use of low-level lateral inhibition should prevent the system from settling on the same vector for each neuron, providing instead a graded topography to the layer.

Depending on the partitioning of the input space, this processing mode of neurons could provide many different behaviors. If the input space has clusters, the neuron may provide classification. If, on the other hand, the inputs are "randomly" distributed in space, the neuron can choose any feature vector, but could be constrained by near neighbors interactions into how it forms topographic maps.

10.3.2 Modeling of Parkinson's Disease

We have created a network with eight neural layers in addition to a layer for stimuli. Each layer represents one physiological region of the brain or nervous system, as described by the model of DeLong et al.[4] By means of a series of excitatory and inhibitory feed-forward and feedback synapses, the brain stem (layer 7) is relatively active. Another layer was added to represent the motor neurons of the extremities (head, legs, and arms). The connections from the brain stem to the extremities are assumed to be inhibitory. Thus, in the normal state, the high activity of layer 7 subjects layer 8 to a large degree of inhibition, making this layer rather inactive. The Parkinsonian case is simulated by cutting off connections stemming from the input layer. When this happens, layer 7 is not excited to a degree as large as it is in the normal case. Sequentially, a smaller amount of activity exists with which to inhibit the extremities. This unusually high level of activation in the motor neurons of the extremities represents the tremors present in patients suffering from Parkinson's Disease.

A Pallidotomy is then simulated in the Parkinsonian scenario by destroying groups of neurons in the Globus Pallidus Internum (or GPi, layer 4). As is evident from DeLong's model (Figure 10.6), this action will reduce the total amount of activity present in the GPi, causing less inhibition to the layer following the GPi, followed by greater excitation of the cortex, greater excitation of the brain stem, and more inhibition to the motor neurons of the extremities, corresponding to a reduction in tremors. The Pallidotomy brings the degrees of activation on layers between the GPi and the extremities back to levels akin to those observed in the non-Parkinsonian scenario. The program allows the effects of different types and locations of lesions to be observed. In general, a lesion is targeted on the location in which the highest degree of activity in the GPi is recorded. The network is helpful in predicting the consequences of lesioning off-target or at a location other than the point of highest activity. Lesioning at multiple locations or on different layers may also be simulated.

The program created a network consisting of eight layers of neurons and one layer of input nodes. Each layer corresponds to one layer in DeLong's model.[4] On each layer are placed 200 neurons. This large quantity of neurons is necessary in order to visualize each layer well. The neurons are randomly scattered on each layer within the spatial bounds of $-2<x<2$ and $-2<y<2$. The input layer consists of two stimuli, each of which contains one input node. One stimulus is located at (–1,0) and the other at (1,0).

The input nodes connect to neurons on layer 1 within a radius of .5 units in the program's coordinate space. These connections are set at a constant value of 1. Each half of layer 1 connects to each of layers 2 or 4 with a radius of 10 units, causing each neuron on the left half of layer 1 to be fully interconnected with neurons on layers 2 and each neuron on the right half of layer 1 to be fully interconnected with neurons on layer 4 (the left of layer 1 does not connect with 4, nor the right of layer 1 with 2). The reason for this is to allow the entirety of layers 2 and 4 to play a role in the simulation while only being exposed to effects from the correct stimulus (as is seen in the figure). If the connections from layer 1 to layers 2 and 4 had a connective radius of only .5 units, then only the left-hand side of layer 2 and the right-hand side of layer 4 would be meaningful. By fully interconnecting the halves of layer 1 to these other layers, all the neurons of layers 2 and 4 are active in the model (technical note: the halves of layer 1 are connected to layers 2 and 4 by first destroying all connections between layer 1 and layers 2 and 4 and then restoring connections only between a circle on the left half of layer 1 with layer 2 and between a circle on the right half of layer 1 with layer 4). Between all other layers of the network, the connective areas have radii of .5 units. Connections are created as displayed in Figure 10.6, but the initial connection strengths (except between stimuli and layer 1) are randomly distributed between –1 and 1 (as opposed to being uniformly inhibitory or excitatory between any two layers as the figure suggests). Connections between stimuli and layer 1 are set at constant values of –1 or 1 (inhibitory or excitatory, as seen in Figure 10.6). In this figure thick arrows represent excitation and thin arrows represent inhibition. For comparison, the left side of the figure is a schematic representation of a normal subject, while the right-hand side depicts a Parkinsonian subject.

The activation function of the neurons in the network takes the form of a sigmoid with minimum value 0 and maximum value 1; when the sum of the inputs equals zero, a neuron will generate an output of 0.5. Even though this propagates a signal through the network when no stimulus exists, the conventional sigmoid (with bounds between –1 and 1, where zero input gives zero output) would not work well for the application in question. Assume, for instance, that layer 1 has a highly positive activation. Then, this would give a large degree of inhibition (a very negative input) to layer 2 (due to the negative weights of the synapses connecting these two layers). Now, if the range of neural activation reaches below zero, the neurons on layer 2 will fire with a highly negative activation. This negative activation will combine with the inhibitory (negative) synapses between layers 2 and 3 to produce an *excitatory* (positive) input to layer 3, causing the neurons on layer 3 to take a highly positive activation. Such a situation does not recreate the conditions this model attempts to emulate. Therefore, the necessity arises of using an activation function that allows the neurons only positive activation. Inhibition and excitation are thus an immediate function of the synaptic connection strengths, not of the neural activity. In this case, high activation on layer 1 causes a large negative input to layer 2, creating a small positive (near zero) output on layer 2. This small positive output on layer 2 feeds into layer 3 as a small negative input. As DeLong's model requires, layer 3 experiences an inhibitory input (though not as inhibitory an input as layer 2 experienced) and thus generates a small positive

output (though larger than the output of layer 2). Evidently, restricting the activation function to a range between 0 and 1 is necessary in order to allow synapses to be inhibitory or excitatory in the fashion this model requires.

The problem now facing the network is that of configuring its connection strengths to DeLong's model (inhibitory or excitatory as seen in the figure) while producing little activation on layer 8 in the normal case and higher activation (tremors) on layer 8 in the Parkinsonian case. This configuration must come about by supervised training of the network with ALOPEX. Training is done with two templates, one representing the normal case and the other representing the Parkinsonian case. In the normal case, the input stimuli both take a positive activation (remember that the connection strengths are what create inhibition or excitation in this model; the activation of a neuron or input node is always positive) and layer 8 (which may be considered the output layer) is to show only a small amount of activity. The template for the Parkinsonian situation has input nodes, each of which has an activation of zero (which is the same as cutting off the inputs), while layer 8 is forced to a higher degree of activity than in the normal state. A special bias in the connection strengths of the network must also be implemented in order to satisfy the condition that these connection strengths be inhibitory between certain layers and excitatory between others (Figure 10.6).

If the network could be trained in such a manner, then a Pallidotomy could be simulated. A specific area (or areas) of neurons on layer 4 (the GPi) would be lesioned (killed), and the result of this lesioning would become apparent on the activation of layer 8 (reduction of tremors in the extremities).

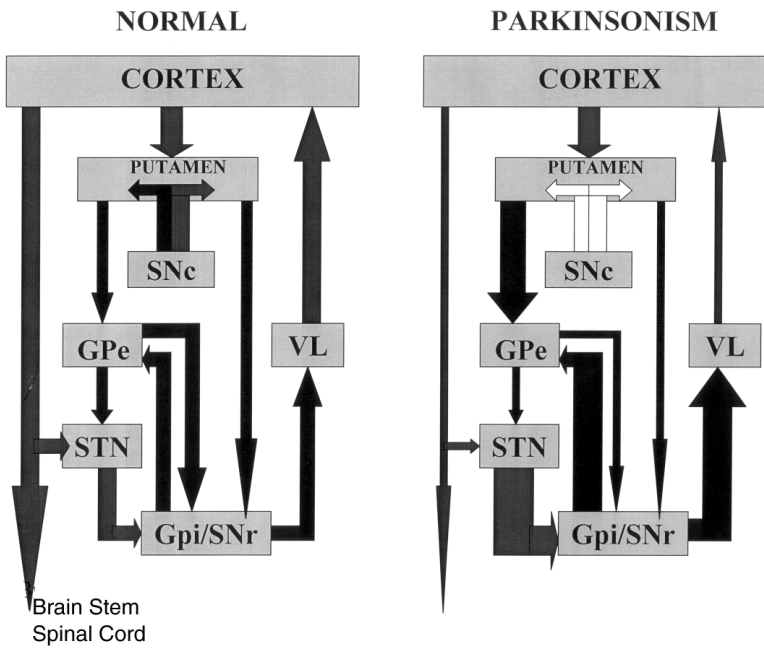

FIGURE 10.6 The DeLong model. Adapted from Reference 4.

10.4 DISCUSSION

For the neural network architecture presented, certain assumptions were made and various constraints were imposed, so that it resembled as much as possible the biological equivalents of feature detectors and edge detectors. In dealing with the above, the neural network can "learn" from stimuli alone, without a set of templates with which to compare the stimuli. To deal with this, the neural network can implement an unsupervised training with a variation of a Hebbian learning rule.[2] The connection strengths among the neurons of this network (weights) thus become the means of storing memories of the presented stimuli, where the same stimulus, if reapplied, will bring the same output to the neural network. These outputs can become the templates to a new neural network—to a different region or even the same region implementing a different function. In recollection, external stimuli must be correlated with memories already stored as templates. In the case of using another neural network for this purpose, the ALOPEX training algorithm[1] can be applied with supervision in the form of previously stored memories.

The storage/recollection process is a dynamic one, and these networks need be coordinated well in order that new "experiences" can affect both networks in a proper fashion. Damage within a network may affect storage or recognition (or both).

The results of an experiment like the simulation of Parkinson's disease as presented here would need to be compared to actual data from the operating room so that the validity of this network could be assessed. One way of comparing results from the network with those from the Operating Room (OR) is by observing (quantitatively) how a patient's motor activity is altered depending upon the location of the lesion relative to points of high and low activity in the GPi. These same observations would then be made on the network. The goal of this experiment would be to match the results of the network as closely as possible to those obtained from the OR. The network contains many parameters that would need to be "played with" in matching its results to real data. Some of these parameters are the connective areas, activation function slope, magnitude of activation function range, spacing and number of neurons on a layer, connection strength range, and other network parameters, as well as the validity of DeLong's model itself.

REFERENCES

1. Micheli-Tzanakou, E., Ueyda, E., Ray, R., Ramanujan, K. S., and Dong, J., Comparison of neural network algorithms for face recognition, *Simulation*, 64(1), 15, 1995.
2. Hebb, D., *The Organization of Behavior. A Neurophysiological Theory*, John Wiley & Sons, New York, 1949.
3. Oja, E., A simplified neuron model as a principal component analyzer, *J. Math. Biol.*, 15, 267, 1982.
4. DeLong, M. R., Activity of basal ganglia neurons during movement, *Brain Res.*, 40(1), 127, 1972.