Searching for an IT model with columnar organization

Jeffrey L. McKinstry

Department of Mathematics and Computer Science, Point Loma Nazarene University,

San Diego, California

jeffmckinstry@ptloma.edu

Abstract

The inferotemporal visual cortex (IT) is at the top of the 'what' stream. Cells in IT are known to respond to complex features and have a columnar organization; nearby columns prefer stimuli with similar features [5]. This research demonstrates that a hill-climbing search strategy can discover a model of IT that exhibits columnar organization, assuming that the patterns do not overlap significantly. This is a first step towards a more complete model of IT. Following others who have used search techniques to adjust neurocomputational model parameters successfully [2], this work further demonstrates the benefit of search methods in computational neuroscience.

Keywords: hill-climbing, inferotemporal cortex, pattern recognition, winner-take-all network

1. Introduction

Hill-climbing [3], a search strategy which always chooses to go up-hill with respect to some objective function, was used to search for a set of parameters that would develop models of IT with a columnar organization.

2. The network

The network is similar to the biologically plausible self-organizing map used by Sirosh and Miikkulainen [4] to model column formation in primary visual cortex. A mean-firing rate neural model was chosen. A small (1 by 10 excitatory and 1 by 10 inhibitory neurons) IT network was used to allow the search algorithm to complete in several hours time on a 1.8 GHz Pentium 4 processor. Each excitatory neuron received input from a common 20 by 20 visual area. The excitatory and inhibitory IT neurons were reciprocally connected. The firing rate of each neuron is calculated as follows:

$$\mathbf{f}(t+1) = \tanh(p\,\mathbf{f}(t) + (1-p)(c_{\text{input}}\,\mathbf{w}_{\text{input}}\cdot\mathbf{x} + c_{\text{inhib}}\,\mathbf{w}_{\text{inhib}}\cdot\mathbf{f}(t) + c_{\text{recurrent}}\,\mathbf{w}_{\text{recurrent}}\cdot\mathbf{f}(t))-\theta)$$

where $c_{\text{conn\_type}}$ is the connection influence for connection type *conn_type*, θ is the threshold, and $w_{\text{conn\_type}}$ is the weight vector for connection type *conn_type*. Weights were updated after each cycle with the following formula:

$$\mathbf{w}(t+1) = \mathbf{w}(t) + \alpha\mathbf{x}(t)$$

where $\mathbf{x}(t)$ represents either the visual input, inhibitory input, or recurrent input depending on the connection type. The learning rate, α, was set independently for each connection type. Each connection type was normalized independently such that the weight vectors had unit length.

3. Training

In order to evaluate the quality of a given network, three training runs were made with the same parameter settings, but different random number seeds and training sequences.

Two training stimuli is the minimum number required to demonstrate columnar organization. Diamonds and squares were used in the experiments. Each stimulus was presented to the network for 4 consecutive cycles before randomly switching to a new stimulus. A blank image was presented to the network for two cycles between each stimulus. Each stimulus was presented to the network an equal number of times. Networks were trained for 100 total cycles during each training run. The network quality was evaluated based on the response of the network during the 100 training cycles as described next.

4. Searching

The search algorithm set the following network parameters. Search ranges are shown in brackets.

> *Neuron parameters*: threshold (θ) [0, 1] and persistence (p) [0,1]
>
> *Connection parameters*: connection radius [0, 6], learning rate (α) [0.01, 0.1], and connection influence (c) [0.1, 1].

In addition, an annulus region was created (radius of 2) for connections from excitatory neurons to inhibitory neurons.

Since the same parameters applied to all neurons of the same type, the number of parameters is independent of the number of neurons in IT. Therefore this method should apply to larger, two-dimensional, networks.

Automated search requires a method of generating new candidate networks and a function to evaluate candidates. New parameters were generated each iteration by randomly modifying the best network found thus far. A Cauchy random number generator was used in order to concentrate the search around the existing solution while allowing for occasional large jumps. In addition, the standard deviation of the distribution was inversely proportional to the search cycle (0.25 at the beginning, and 0.05 at the end).

Once a new candidate network was generated, a quality metric was applied. Only improvements in quality were accepted. The quality measure, *q*, for a single training run was defined as

$$q=\Sigma_{i\in S}\, m_i + \Pi_{i\in S}\, m_i$$

where S is the set of all stimulus patterns (2 in this case), and m_i is the winning margin for stimulus i:

$$m_i = g(\Sigma_{t\in I}\, f(t) - \Sigma_{t\notin I}\, f(t))$$

where I is the set of all time steps with input stimulus i, and g(x) is half-wave rectification. The function *q* encourages networks with cells which respond strongly to only a few stimuli, but yet respond to all of the stimuli. The second term is maximized when the winning margins for each category are equal, rewarding networks which divide up the neurons in proportion to the input stimulus distribution.

5. Results

The search algorithm was allowed to run until 2,000 networks had been evaluated. The best set of parameters was saved and used to train a new network shown below. This network exhibits a columnar organization, as shown in the figure 1.

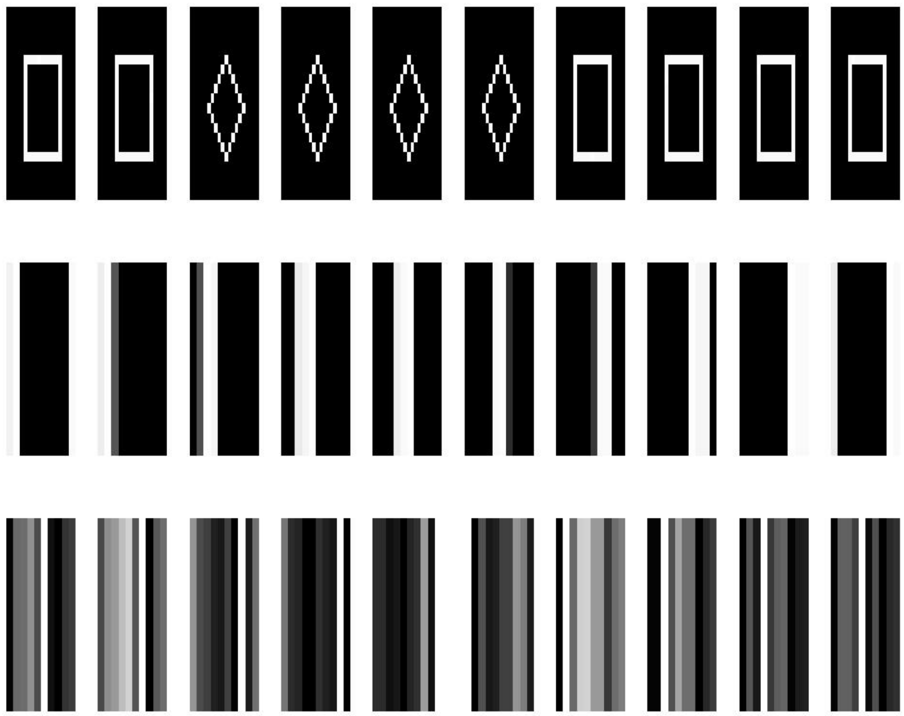


Fig. 1. Shown are weights for all input connections from V4 (first row), lateral connections from IT (second row), and inhibitory connections from ITi (last row) for each of the 10 excitatory neurons in IT, where white is maximum and black is the minimum value.

Note that neurons that prefer the same stimulus cluster together (see the first row showing connections to the visual input area), and that after training, each neuron responds to a particular input pattern. In addition, the network divides up the neurons such that some respond to each pattern rather than all responding to the same stimulus.

To assess the robustness of the parameters obtained, the search algorithm was run five times. For each run, the resulting parameters were used to train 10 different networks starting from different initial weights. The networks converged such that both objects were represented by different groups of neurons 8/10, 9/10, 9/10, 0/10 and 8/10 times with the five different parameter settings. The parameter settings obtained from all five runs are given in the following tables. Parameters which are similar across most of the runs are shaded.

Neuron parameters:

IT_e threshold	IT_e persistence	IT_i threshold	IT_i persistence
0.94	0.0	0.89	0.83
1	.0007	0	0
.91	.25	.64	1
1	0	.02	.68
1	.02	.17	.77

Connection parameters:

V4 to IT_e	IT_i to IT_e	IT_e to IT_e	IT_e to IT_i	IT_i to IT_i

α	C	rad.	α	c	rad.	α	c	rad.	α	c	rad.	α	c
0.08	0.64	5.97	0.1	-0.28	1.69	0.05	0.1	6	.01	.95	2.00	.02	-.55
0.07	0.70	5.56	.01	-0.48	0.74	0.01	0.97	5.27	.04	.30	.01	.01	-.75
0.10	0.63	3.76	.01	-1	0.71	0.08	0.12	0.80	.01	.99	5.79	.03	-.98
0.08	0.66	5.79	.10	-1	1.01	0.10	0.80	4.30	.06	.30	.78	.01	-.90
0.10	0.69	4.27	.06	-0.83	6.00	0.01	0.11	4.88	.05	1.0	6	.03	-.97

The parameters obtained by searching yielded significantly better results than 5 random sets of parameters ($p<.05$, Wilcoxan Rank Sum). Each of these random parameter settings converged zero out of 10 times with different initial weights. This demonstrates that parameter tuning was necessary for this problem.

Several patterns can be found in the parameters from the five different runs. The learning rate and connection influence from the input layer (V4) to the excitatory IT neurons are remarkably consistent. It also appears that the networks are basically winner-take-all networks, with large inhibitory radius values between the excitatory and inhibitory neurons in IT, causing strong competition between the neurons in the network. Finally, the lateral connections from the excitatory IT neurons tend to be local (small radius), which would help to make nearby neurons respond together. There is a reciprocal relationship between the connection influences of the lateral connections and the excitatory to inhibitory connections; strong local excitation is balanced by a reduction in the connection strength to inhibitory neurons thereby maintaining a similar overall level of inhibition. Hill-climbing was successful in discovering effective parameters in this moderately complex, recurrent network.

The two patterns used in the first experiment were nearly orthogonal, overlapping in only 4 pixels. Can this class of network solve a more difficult pattern discrimination problem in which the patterns are more similar? The experiments above were repeated with two patterns that shared approximately 50% of their features in common: the square from the first experiment and an hour-glass shape which shared the top and bottom lines with the square. All of the five sets of parameters found by the search procedure produced the same result: zero out of ten networks converged to a correct solution. All of the neurons in the networks ended up responding to either the same object (e.g. all were hour-glass detectors) or to both objects, instead of dividing the two patterns among the neurons. These results were significantly worse than the results of the first experiment with highly dissimilar patterns ($p<.05$, Wilcoxan Rank Sum). This does not prove that no parameters exist that will allow this network to discriminate similar objects, but it does suggest that it is difficult to find them. If one assumes that biological neural networks should be robust to changes in parameters (and that realistic input patterns will not necessarily be orthogonal), this result suggests that a different, perhaps more complex network architecture is required.

6. Conclusions

It was shown that a simple search strategy can find the parameter settings for a modestly complex network which discriminates between two nearly-orthogonal patterns. The constraint that nearby neurons should respond to similar patterns was enforced by the objective function optimized by the search procedure.

No parameters were found that could separate the patterns with a high degree of overlap. It should be noted that such patterns can be discriminated with networks which assume non-local signaling, such as the self-organizing feature map [1]. It would therefore be of interest to expand the complexity of the network, searching for solutions to this problem in the expanded search space.

The most serious drawback of the method is the computational complexity, which limited the complexity of the network. Larger networks and harder problems can be solved with the use of parallel processing and more efficient coding; this method will become more attractive as computing power continues to grow exponentially.

References

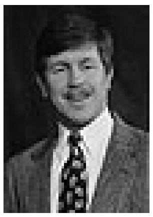
[1] T. Kohonen, *Self-Organization and Associative Memory*. (Springer-Verlag, New York, NY, 1988).

[2] E.T.Rolls and S. M. Stringer, On The Design Of Neural Networks In The Brain By Genetic Evolution. *Prog Neurobiol* 61 (2000) 557-79.

[3] S. Russell and P. Norvig, *Artificial Intelligence: a modern approach* (Prentice-Hall, Upper Saddle River, New Jersey, 1995).

[4] J. Sirosh and R. Miikkulainen, Self-organization and Functional Role of Lateral Connections and Multisize Receptive Fields in the Primary Visual Cortex, *Neural Processing Letters* 3 (1996) 39-48.

[5] G. Wang et al., Functional architecture in Monkey Inferotemporal Cortex Revealed by in vivo Optical Imaging, Neurosci Res 32 (1998) 33-46.



Jeffrey L. McKinstry received his B.A. in Computer Science from Point Loma Nazarene University (1987). He received an M.S. in Computer Science from USC in 1991. In 1999 he received a Ph.D. in Computer Engineering from the University of California, San Diego. His dissertation was entitled, “A model of Primary Visual Cortex: From Single Cells to Feature Maps”. He is currently working as a postdoc in theoretical neurobiology at The Neurosciences Institute in La Jolla, California and teaching computer science at Point Loma Nazarene University in San Diego.