Algorithm programmed by a neural network model for coordinate transformation

Sabrina J. Goodman
and
Richard A. Andersen
Department of Brain and Cognitive Sciences
Massachusetts Institute of Technology
77 Massachusetts Ave. – Room E25-236
Cambridge, MA 02139

Abstract

We have trained a neural-network model for coordinate transformations simulating the action of parietal area 7a (Andersen & Zipser, 1988; Zipser & Andersen, 1988; Goodman & Andersen, 1989). Examination of this network reveals that its planar responses extract the vector coordinates of the angles represented by the inputs, in order to add them together. The hidden unit responses were most planar when the number of units was minimal; the corresponding area of the brain evidently has several times as many hidden units for the task being performed. This cortical area may be using the same algorithm as the neural-network model, so examination of the model clarifies the function of the neurons.

Most criticisms of neural networks attack either their biological plausibility or the imprecision of having an algorithm trained instead of programmed. In recent studies we have attempted to address both of these issues. In support of neural networks as a plausible model of biological computation, we have compared the activity of individual units in one neural network simulation with the activity of individual neurons in animal brains, particularly area 7a of the posterior parietal cortex of monkeys (Andersen & Zipser, 1988; Zipser & Andersen, 1988; Goodman & Andersen, 1989). In this paper we examine the computerized network model to clarify the algorithm it performs once trained. Thus we can say more about the action of the brain than that cortical neurons have response properties similar to the units in the network; we can also say what this network does. This knowledge will then provide insight into the functioning of the neural structure being modeled.

We have examined variants on the neural network described by Zipser and Andersen (1988). The input layer units were programmed to behave like those neurons in area 7a of the posterior parietal cortex found to respond exclusively to either the angle of the eyes or the retinal location of a visual stimulus, on the theory that these neurons provide input to a mechanism for coordinate transformation in area 7a. The network was trained to produce an output representing the location of the visual stimulus relative to the head, i.e. the sum of the retinal and eye-angle locations. The resulting behavior of the hidden units could then be compared with those area 7a neurons which respond to both signals. This three layer network was trained by the method of back-propagation learning (see McClelland and Rumelhart, 1988).

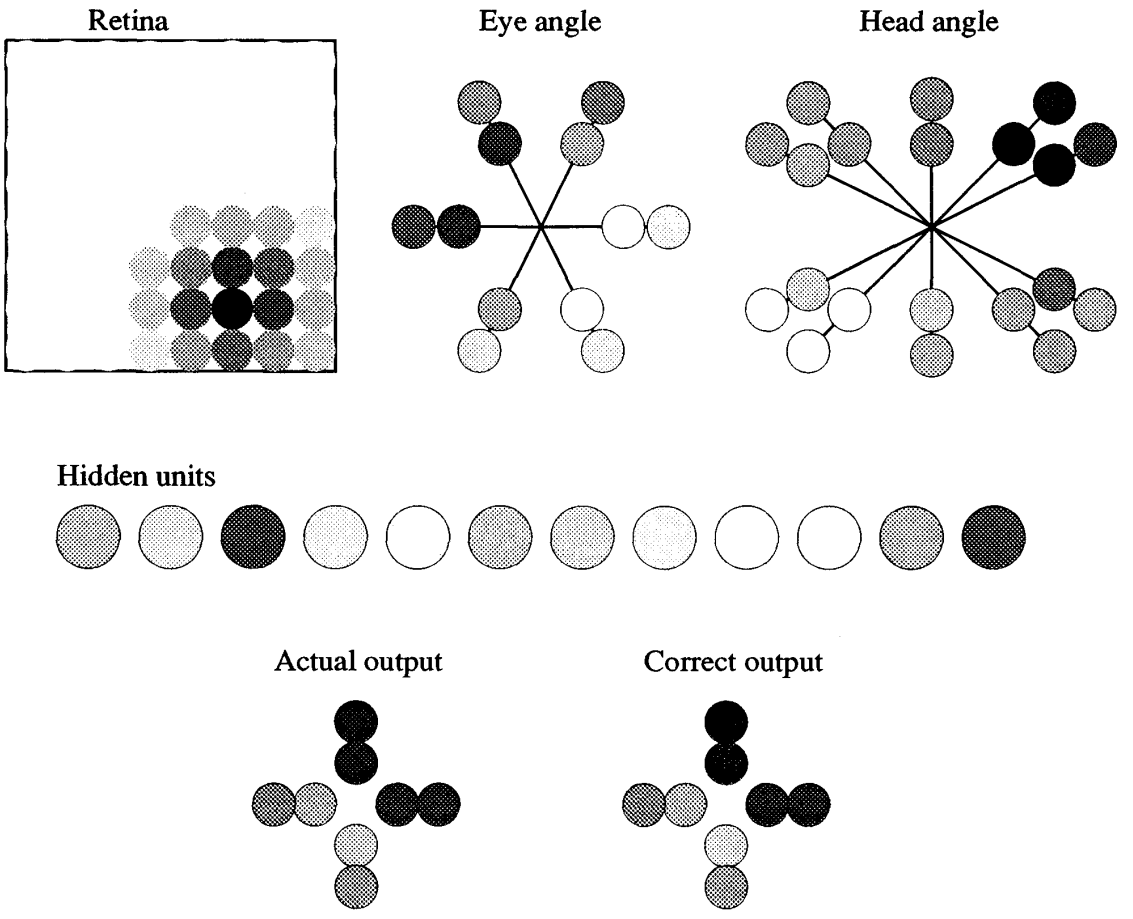

Figure 1: One of the neural networks used, with head as well as eye angle input. An example training pattern is shown to indicate the spread of a single point of light in the Gaussian array and the directions coded by the various gaze angle inputs, the actual hidden and output unit activations produced by this input in the trained network, and the desired output.

Each of the networks we trained had inputs representing the retina and the eye position, and, unlike previous models, some also had head position inputs. The output represented the gaze-independent location of the stimulus in space. The retinal input was comprised of an array of units with Gaussian shaped retinal receptive fields. The centers of these receptive fields were separated by 10 degrees of visual angle in an 8 by 8 grid, and the 1/e spread of the Gaussian functions was 15 degrees. The gaze angle inputs consisted of 32 units, of which the activity of each was a monotonic function of angle in one direction. These functions were linear for values between zero and one. The slopes and intercepts were randomly chosen for each unit within the range of values found for cells in area 7a. In the networks with no head input, half of the gaze angle inputs were functions of horizontal eye position only and half represented vertical eye

position only. In the networks which also added the angle of the head, the gaze inputs coded positive or negative angles along eight different lines, three for the "eyes" and five for the "head". Figure 1 shows a diagram of the network with the head angle input. Both types of network had a linear output format, like the eye or head angle input, but coding the location of the stimulus in body-centered (eye plus head plus retinal) rather than retinal coordinates. That is, it represented the angle to which the gaze should be moved to look directly at the stimulus point.

Andersen and Zipser (1988) examined the gain fields (see Figure 2) and receptive fields of the hidden units in these networks and found them similar to those found in area 7a of the monkey's parietal cortex. In particular, the gain fields tend to be planar, while the receptive fields may or may not be (see Figure 3). In both the brain and the networks, three patterns of gain tend to appear: the background activity (with no light stimulus) increases monotonically in one direction, while the increase in activity when a light is flashed (the gain) either increases, decreases, or first increases and then decreases with increasing background activity. In the network simulations on the computer, background means the activation of a hidden unit with all zeros on the retinal input, and the gain is the change in activation between that and the same "eye position" with a retinal input. The receptive fields also seem to have similar shapes in area 7a and the network model, covering most or all of the visual field with varying responses which peak to one side.

We have also simulated microstimulation experiments (Goodman and Andersen, 1989) by increasing the activation of individual hidden units after an input was presented and recomputing the output. This is meant to be equivalent to inserting an electrode in the animal's brain and stimulating a small number of neurons. The resulting changes in output, taken to represent eye position, were then compared with data on microstimulation-induced saccades (eye movements). Again, similarities were seen between the behavior of the network and the brain microstimulation data.

A particularly interesting aspect of the receptive fields, gain fields, and induced saccades from the network simulation is that for any one hidden unit, these three fields seem to go in the same direction. That is, the peak response to the retinal input, the direction of increasing background activity from eye position input, and the average saccade direction all correlate positively. This similarity of response directions for the different inputs was particularly striking when we added a head angle input. This was encoded in monotonic form like the eye angles, but with different specific linear functions of position, in a different set of directions. In spite of these differences between the two gaze angle inputs (eye and head), they are essentially the same way of representing a position. The monotonic input units are like vectors spanning the plane of the visual field, and the gaze angle is the sum of the vectors for head and eye positions. The network evidently used this fact to add the gaze angles at the hidden level, so that each hidden unit responds to the total gaze angle independent of the separate contributions of the eye and head inputs: The response of each hidden unit to head angle is virtually indentical to its response to eye angle. Figure 2 illustrates this by showing the gain fields, for the both monotonic inputs, of a typical hidden unit.

If the retinal input is also considered as a vector, specifying the (retinotopic) location of the point of light, the desired output of the whole network can likewise be calculated as a sum of vectors. A hidden unit extracts one coordinate (direction) of the stimulus if its activity is a planar function of the stimulus position. The activity of such a unit varies linearly with one component of the retinal-location vector. By extracting the same component from the gaze angle vector, with the same function of position, it can add them together automatically. This

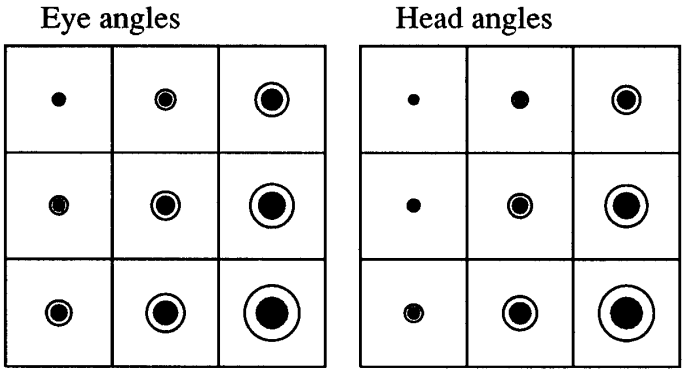

Figure 2: Gain fields for eye and head positions. The gain field was mapped by presenting a visual stimulus at the same retinal location, in the most responsive area of the receptive field, at nine different eye positions. Each set of circles represents the response at one eye position; these were spaced by 20 degrees in a 3 x 3 array with the central location (0,0). This protocol mimics the one used to map gain fields in the animal recording experiments (Andersen & Zipser 1988). The inner, dark circle of each set has a diameter proportional to the increase in activity generated by the visual input, and the annulus diameter is proportional to the eye position contribution to activity. These two gain fields are from one hidden unit in a network with non-orthogonal eye and head angle inputs.

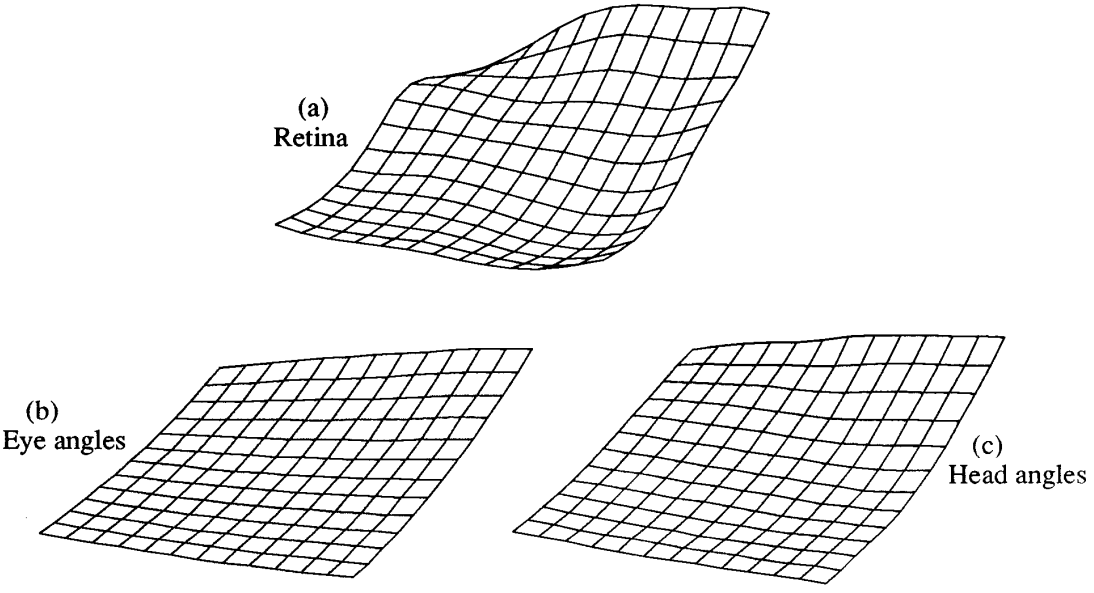

Figure 3: One hidden unit's responses to all three inputs. Height represents hidden unit activation, as a function of the position of (a) retinal stimuli (receptive field), (b) eye angles or (c) head angles.

network's output, the gaze-independent location of the stimulus (or the position to which the eyes must be moved to look at it), is in fact the sum of the two inputs if they are considered as vectors in this way. Thus a hidden unit with identical planar responses to each input position would have one component of the output solved, and could be an output unit in the linear-output form of the network.

In linear algebra, this vector addition could be described as follows, with vectors $\mathbf{r}$ = the retinal stimulus position, $\mathbf{e}$ = eye (or total gaze) angle, and $\mathbf{o}$ = output, and hidden unit activations α_i = the activation of the *i*th hidden unit. The direction of the planar response of a given hidden unit is also the vector onto which the input vectors are projected, when they are multiplied by the weights to that hidden unit; call this $\mathbf{a}_i$ for the *i*th hidden unit. Of course, these will be somewhat simplified. The network has 32 or 64 input units for each of the locations to be added together, but the vectors ($\mathbf{r}$, $\mathbf{e}$ and $\mathbf{a}$) are two-dimensional; this is a useful level of abstraction.

The desired result is

$$\mathbf{r} + \mathbf{e} = \mathbf{o}.$$

It follows that

$$\mathbf{r} \bullet \mathbf{a} + \mathbf{e} \bullet \mathbf{a} = \mathbf{o} \bullet \mathbf{a}.$$

The hidden unit activation is

$$\alpha_i = \mathbf{r} \bullet \mathbf{a}_i + \mathbf{e} \bullet \mathbf{a}_i,$$

so in effect

$$\alpha_i = \mathbf{o} \bullet \mathbf{a}_i.$$

Then the output weights need only act as the inverse of all the $\mathbf{a}_i$'s.

$$\mathbf{o} = \Sigma_i \, \alpha_i \mathbf{b}_i,$$

where $\mathbf{b}_i$ is the output vector (projection) of the *i*th hidden unit. The outputs of the hidden units were found to have the same direction tuning as the inputs to the hidden units. However, it is too simple to assume $\mathbf{b}_i = \mathbf{a}_i$, since the output is affected by twelve hidden units whose effects must all be balanced. Rather, the weight matrix formed by all of the $\mathbf{b}_i$'s must act like the inverse of the matrix of all of the $\mathbf{a}_i$'s.

The networks considered here do not give perfectly planar responses to any input, nor are their visual receptive fields identical to their background response to eye angle, except in one case: when the network had only two hidden units and a linear output. Under the constraint of only one unit for each dimension of the output, the network was forced to complete the computation at the hidden layer, and it produced completely planar responses. With our usual 12 hidden units, or with a gaussian output format, the network was more complicated, but still clearly doing the same thing. In the linear-output case in particular, it seems that the hidden units failed to solve the problem perfectly only because they didn't need to. The responses approximated the expected planes more closely when fewer hidden units were available. However, the non-linearity of the actual network behavior is also important, since that brings it closer to the behavior of the actual neurons.

A completely linear network would have constant gain, that is, the inner dark circles in Figure 2 would all be the same size (for any given hidden unit and retinal stimulus). As you can see in Figure 2, this is not the case in these networks; and it is not the case in the brain either (Andersen & Zipser, 1988). In the computer-simulated network, the non-linearity is caused by the sigmoid ("squashing") response function of the units. This function, $\alpha = 1/(1+e^{-(\text{netinput}+\text{bias})})$, was chosen to keep the activation of each unit between 0 and 1 in a

continuously differentiable way. The unit response function must be differentiable, unlike the threshholded-linear functions used in some previous neural network models, for the back-propagation training to work. The fact that the neurons show the same behavior suggests that they have a similar response function. The neural data most closely resembles the network properties from networks with more than twelve hidden units, in which the receptive fields do not look like planes.

With many hidden units, the activity of this neural network is complex and non-linear, but clearly resembles the activity of neurons in area 7a of the posterior parietal cortex in monkeys. With only two hidden units, the network keeps its activity in the near-linear region of the sigmoid response function, and the hidden units respond linearly to both inputs. The resulting linear algorithm is much easier to understand. Training networks with intermediate numbers of hidden units reveals the continuous variation between these, indicating that they use the same algorithm with varying degrees of noise and redundancy. This suggests that parietal area 7a performs the same coordinate transformation with essentially the same vector-addition algorithm, using a large redundancy of neurons to allow less precision in the responses of each.

References

Andersen, R.A., & Zipser, D. (1988). The role of the posterior parietal cortex in coordinate transformations for visual-motor integration. *Canadian Journal of Physiology and Pharmacology*, **66**, 488-501.

Goodman, S.J., & Andersen, R.A. (1989). Microstimulation of a neural-network model for visually guided saccades. *Journal of Cognitive Neuroscience*, **1**(4), 317-326.

McClelland, J.L., & Rumelhart, D.E. (1988). *Explorations in Parallel Distributed Processing*. Cambridge, MA: MIT Press.

Zipser, D., & Andersen, R.A. (1988). A back-propagation programmed network that simulates response properties of a subset of posterior parietal neurons. *Nature*, **331**, 697-684.