PROPERTIES OF CONNECTIONIST VARIABLE REPRESENTATIONS[1]

Deborah Walters

Department of Computer Science, State University of New York at Buffalo, Buffalo, NY

Abstract

A theoretical classification of the types of representations possible for variable in connectionist networks has been developed [1]. This paper discusses the properties of some classes of connectionist representations. In particular, the representation of variables in value-unit, variable-unit and intermediate unit representations are analyzed, and a course-fine concept of representation developed. In addition, the relation between the measurement of a feature and it's representation is discussed.

1. Connectionist Networks

Connectionist networks consist of a large number of very simple processing elements, which are highly interconnected, with each processor receiving input from and sending output to many other processors. In a broad sense of connectionism there are various types of connectionist networks: in cellular automata networks, the connections are generally limited to those between nearest neighbors, and the computations are generally deterministic [2]; in cooperative and competitive networks it is the dynamical analysis of feedback, shunting etc. within and between layers of processing units that is generally studied [3]; while in the "connectionist school" complete interconnectivity is permitted, and either local or distributed representations of features are used [4,5]. This paper is concerned with connectionist networks in this broad sense: it concerns data-parallel processing where each processing element within a group has the same program, and each processing element is connected to other processors which lie within its local neighborhood.

Connectionist networks have been studied for a variety of reasons. One motivation comes from cognitive science, where the desire is to understand the brain as a computational device. As the brain consists of large numbers of massively interconnected simple processing elements, the study of connectionist networks may ultimately aid us in understanding neural computations. However, this goal must be treated with caution; the analogy to neurons must be made at the proper level.

A more recent motivation for studying connectionism has arisen from the computer science emphasis on parallel processing, as connectionist networks are an example of fine-grain parallelism. The style of computing that is possible with such parallelism is very different from that possible with uniprocessor systems, or with parallel systems containing a small number of processors, and thus represent a distinct class of parallel computations. With the advent of the Connection Machine [6], a fine-grain parallel machine, there is increasing motivation to understand the types of computations possible with such hardware.

1.1. The Representation of Variables in Connectionist Networks

Just as a different style of computation is possible in a connectionist network, the styles of representation of variables that are natural for a connectionist network may be very different from the types of representation natural for serial, or coarse grain parallel processing. For example, Feldman and Ballard use explicit local representations of image features, which they refer to as parameter spaces or feature spaces. Shapes or objects are represented by a set of particular values of certain features, and the connectionist computations involved are basically indexing operations into the feature space, and constraint satisfaction between sets of features. Grossberg, however, does not use such a feature representation, but rather investigates the more global patterns of activity in a field of connectionist units. Similarly, Hinton, McClelland and Rumelhart [4], and Kohonen [7] use distributed

[1] This research is funded by NSF Grant IST 8409827 awarded to the author.

representations.

A general theoretical classification of the types of variable representations possible in connectionist networks, in which the different existing variable representations can be expressed, and new types of representations can be indicated, has been created [8]. The theoretical framework enables the properties of different connectionist representations to be formally analyzed which allows the principled choice of the optimal representation for a particular application. As the framework has been treated in detail elsewhere [1], only a brief description occurs here.

2. Preliminary Assumptions

The theoretical framework for connectionist representation of variables is based on certain general assumptions about connectionist networks, the processing elements which participate in connectionist networks, and the nature of variables that are to be represented in such networks. These assumptions are enumerated below.

2.1. Connectionist Units

Connectionist networks are to contain a large number of simple, identical processing units, each of which are capable of signaling a limited number of values. For example, if a connectionist unit consisted of an 8-bit memory word, it could represent or signal only 256 separate values. There are two means by which connectionist units can represent information: explicitly through it's level of activity or the value it stores; and implicitly, as each unit in a multi-unit system can represent a different value or range of vales. For example, in representing color, three units could be used; one each to implicitly represent red, blue and green. Each of these units could then explicitly represent the intensity value of it's implied color component.

Notice that it is not the connections between units which encode the information here, as we are primarily concerned with short-term rather than long-term encoding of visual information. Other researchers have used the connections between units to encode longer-term information.

2.2. Representation of Variables

The basic property of a variable is that it can take on a range of possible values. For each variable there is an n, and an injection g, such that g maps the set of values of that variable into euclidean n space, and only k distinct values lie along each dimension of a variable.

3. Definitions of Types of Variable Representations

The general theory of variable representations for connectionist networks assumes that representations can be classified in terms of the following three properties.

3.1. Conjunctive versus Disjunctive Representations

For an n dimensional variable, it is possible to have a completely conjunctive, a completely disjunctive or a partially conjunctive / partially disjunctive representation. A completely disjunctive representation could be thought of as containing n separate one-dimensional representations, one for each dimension of the variable. This contrasts with the completely conjunctive representation which would contain only a single n dimensional representation. A partially conjunctive / partially disjunctiove representation would consist of at least one one-dimensional representation, and at least one m-dimensional representation, with $m < n$.

One way to classify representations in terms of this property is to express the number of disjunctive dimensions present in a given representation. With no disjunctive dimensions, the representation would be completely conjunctive, and with n disjunctive dimensions a representation would be completely disjunctive, and intermediate values would be partial.

3.2. Variable-unit versus Value-unit Representations

Connectionist representations of variables can also vary in terms of the number of distinct values that each constituent unit can signal. One way to express this is in terms of the "memory size" of the connectionist units. It will be assumed here, without loss of generality, that connectionist units use a binary representation, and the memory size will be expressed in bits.

There are two extremes to the possible variable representations in terms of the memory size of connectionist units required to represent one value of a variable. In the variable-unit representation each unit has $\log_2 k'$ memory size, where k' is the number of distinct values of the variable to be represented. In the value-unit representation each unit has a memory size of 1. A single variable-unit can be used to signal the presence of any of the k' distinct values, while a single value-unit can only represent one of the k' values, and k' value-units would be required to represent any arbitrary value along a dimension.

The two types of coding referred to are the logical extremes of codings well known in the various disciplines which are concerned with connectionist computations. Neurophysiologists refer to the first type as a frequency code, and to the second type a a labeled-line code [9]. Ballard has discussed the implications of each type of coding, which he refers to as variable and value coding respectively [5].

The value unit and variable-unit encodings are the two possible extremes of variable representation in terms of the memory-size of units, and the number of units required to represent a variable. It is also useful to consider a representation which is intermediate between these extremes. In an intermediate representation the memory size of the units is b, $(1<b<\log_2 k)$. In both biological and machine systems, the intermediate-unit representation is often used.

3.3. Response Overlap

Representations can also vary in terms of response overlap. In a no response overlap representation a particular value of a variable along one dimension is represented by the activity of a single unit. In response overlap representations the activity of each processing unit represents a range of d discrete values of a variable along each dimension, and each particular value is encoded by the activity of a number of overlapping units. Connectionist units participating in a response-overlap representation are said to have a diameter of d.

4. Theoretical Classification of Variable Representations

Variable representations can be classified along the three dimensions described above, which form the connectionist variable representation space (VRS) illustrated in Figure 1 [1]. VRS provides a theoretical framework for describing the connectionist encodings used in previous research. For example, Hinton's coarse coding scheme is a conjunctive, overlapping, value-unit representation [10], while Ballard uses a conjunctive, non-overlapping value representation in his connectionist shape perception algorithm [5]. The neurons in the mammalian striate cortex that are selectively sensitive to a small range of spatial frequencies (or edge widths) and a small range of edge orientations are using a conjunctive, overlapping, intermediate-unit representation. In the MT region of visual processing, there is one set of neurons which are selectively sensitive to velocity, and another set which are selectively sensitive to the direction of motion. These neurons appear to be using a disjunctive, overlapping intermediate representation. In computer vision programs where edge information is represented as an intensity map, and an orientation map, the disjunctive, nonoverlapping variable representation is being used.

Each point in VRS represents a class of variable representations. In order to completely define a specific representation, more than it's location in VRS must be known; in addition the response mapping function must be specified. The response mapping function, f, defines the response of a given connectionist unit to the k values along each dimension of a variable.

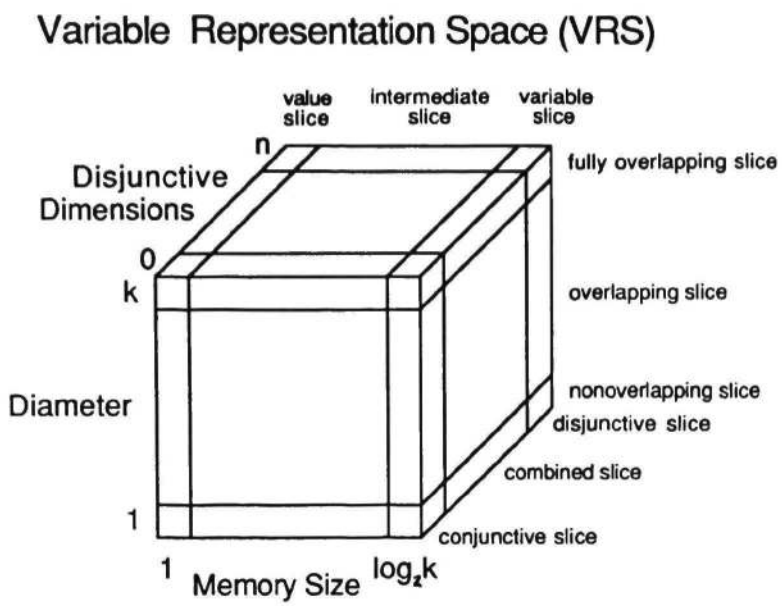

Figure 1

5. Measurement Issues: Response Function Analysis

By analyzing the properties of each region of VRS, the optimal class of representations for a given problem can be determined. The chosen representation can then be completely specified by defining the response mapping function, f. However, in an actual implementation such as computer vision, the real problem is to find a method for measuring the feature values present in an image. Thus it is not a question of choosing a variable representation and defining a mapping, because the measurement process defines the response mapping, and therefore constrains the choice of representation. In order to choose the correct representation, the response mapping function for a particular measurement process must be determined. This is an important point that has been previously neglected because much connectionist research has primarily emphasized intermediate and high-level visual processing and thus assumed that the input to the network had already been processed into the appropriate form.

5.1. Determining Response Mapping Functions

In many cases the methods for measuring a feature value directly from an image do not yield a simple, single dimensional mapping. The response of a detector can be a function not only of the value of the desired feature, but also of many related features. Thus the first step in representing variables measured from images is to determine just what is being measured by plotting the multidimensional response function for the measurement process, as a function of a priori image properties. For example, convolving an image with a template is one means of measuring the orientation and amplitude of an edge in an image. The question arises as to how well the output of the convolution correlates with the presence of an edge. To study this, the response function for an edge detector might be plotted against the following properties of edges: location, length, width, amplitude, orientation, curvature, image sampling, signal-to-noise ratio, orientation and/or curvature discontinuities, and edge profile. An example of orientation response functions for one set of oriented edge operators is shown in Figure 2a. Each curve is the response, as a function of the edge orientation, of one operator when convolved with a step edge. From these response functions it is clear that the edge operators create an overlapping, intermediate unit representation. But the response of an edge operator is not a single dimensional function: the response can vary as a function of edge amplitude, width, profile, orientation, and distance from an edge. For example, Fig. 2b shows the responses of one operator to step edges of different amplitudes. The response is obviously a function of both edge orientation and amplitude, which suggests a conjunctive representation of orientation and amplitude. Similar results would be obtained if the other dimensions were explored.

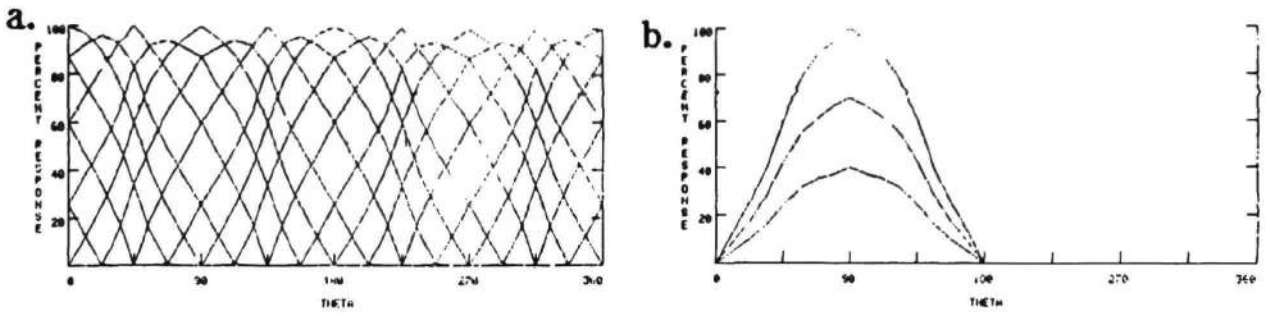

Figure 2

If a particular type of representation is required, then the measurement process must be designed to give the appropriate response mapping function. Thus response function analysis may provide useful constraints in the design of image based operators.

6. Properties of Feature Space Representations

At present only the partial analyzes of particular representations exist: Hinton [10] and Sullins [11] have both analyzed the distributed coarse coding representation; Ballard [5] has investigated some of the implications of value-unit versus variable-unit representations; and, Saund discusses a representation useful for dimensionality reduction [12].

In analyzing representations, several properties should be considered, including the following:

1) Representation of multiple values

The representations differ in terms of the number of distinct values of a feature that can be represented by one copy of the representation. For example, the variable code requires i copies to represent i values, while the value code requires only one copy, as long as there is a one-to-one mapping between values and units. Such differences effect overall coding efficiency.

2) Match between representation and implementation architecture

Connectionist units may have limited memory, which would influence the choice of a representation. For example, value-unit or intermediate-unit encoding is useful for units with small memory size, while variable unit representations are possible when units have enough memory.

3) Total Representation and Generalization

How many copies of a representation are required to represent all possible feature values? In general, the larger the diameter of a unit, the more copies required to simultaneously represent all possible feature values, but this also depends upon the internal code of the units. Thus a system with narrow diameter units has the advantage of being able to simultaneously represent multiple values. But in terms of generalization, the opposite is true. For example, if the mapping is not ordered, then a metric other than the simple distance metric must be used to determine the similarity between feature values. When the unit diameter is broad, a suitable simple metric exists: feature values which both activate the same unit are similar. Intermediate representations provide a useful compromise between total representation and generalizations.

4) Item density

The distribution of the feature values that will be encountered in a given situation is important, as representations differ in the density and distributions which can be handled.

5) Required degree of accuracy

Another property that is influenced by the characteristics of the variable to be represented is the required degree of accuracy, and related sampling issues, as are discussed in Section 6.1.

6) Diameter of response ranges and degree of overlap

These properties influence the generalization capabilities, the efficiency and the suitability for implementation in a particular architecture.

7) Probabalistic Representation
An efficient probabalistic representation is suitable for many applications.

These properties can be used to determine the type of representation best suited to represent particular types of information. For example, there are a variety of questions that could be asked about the represented information: "Is value x of feature y present?"; "How many instances of value x of feature y are present?"; "What is the value of feature y at a given spatial position?"; etc. Which types of information should be available thus depends on the nature of the features being represented, and on the types of computations in which the features will be involved.

6.1. Resolution of A Representation

Another property of representations is the resolution to which values of a variable can be encoded. This accuracy will obviously depend upon the sampling resolution of the representation. For example, in the variable-unit representation each unit can distinctly signal the k values of a variable, thus the resolution is k values/dimension, which is the best possible resolution. But even representations with coarse sampling, such as conjunctive, overlapping value units, can have a resolution equal to the variable representation.

Another way of stating the resolution issue is to discuss the degree of accuracy to which the value of a variable can be determined when it is encoded in a particular representation. In the remainder of this section the resolution of the various representations are discussed.

6.1.1. Estimation of the Value of a Variable from a Variable-unit Representation

In all types of variable-unit representations, the value of a variable is represented by the activity of a single processing unit. Thus if f is the response mapping function of a processing unit, then $f(x)$ is the representation of the value x of the variable. If f is a single-valued function, and if $f(x_1) = f(x_2)$, then $x_1 = x_2$, then x can be uniquely determined from $f(x)$ if f is known, and the resolution is k values/dimension.

6.1.2. Estimation of the Value of a Variable from a Value-unit Representation

In the non-overlapping value unit representation there is a separate processing unit, P_x to represent each value x of a variable, thus the activity in a unit uniquely represents the value of a variable, and the maximal resolution is achieved.

In the overlapping value-unit representation, the value of a variable is represented by the activity of a collection of processing units, and the resolution of each unit is coarse. Hinton has shown that the value of a variable represented in a coarse coding scheme (an overlapping, value-unit representation) can be uniquely determined if the number of values being represented is $\leq ((k/d)-1)^n$ [10].

6.1.3. Estimation of a Value of a Variable from an Intermediate-unit Representation

In an intermediate-unit representation with no overlap, each dimension can be broken down into $k/(2^b)$ sections, with a variable unit representation in each section. In this case, the resolution is maximal, and the value of a variable is represented both by the activity-level of a given processing unit, and which unit is active.

In terms of the estimation of a value, the most interesting representation is the overlapping, intermediate-unit encoding. The following analysis applies to the disjunctive class of this type of representation for sparse data.

Assume each dimension of a variable is periodic, with period P, thus $x+P = x$. Further assume response functions are strictly monotonically decreasing for $x \geq r$, and are strictly monotonically increasing for $x \leq r$, where r is the peak value of f. Figure 3a shows an example response function, f_{r_1}, which satisfies this assumption. Assume a given representation consists of m identical response functions, thus for all r_1, r_2, x; $f_{r_1}(x) = f_{r_2}(x+(r_2-r_1))$. Figure 3b shows a protion of a

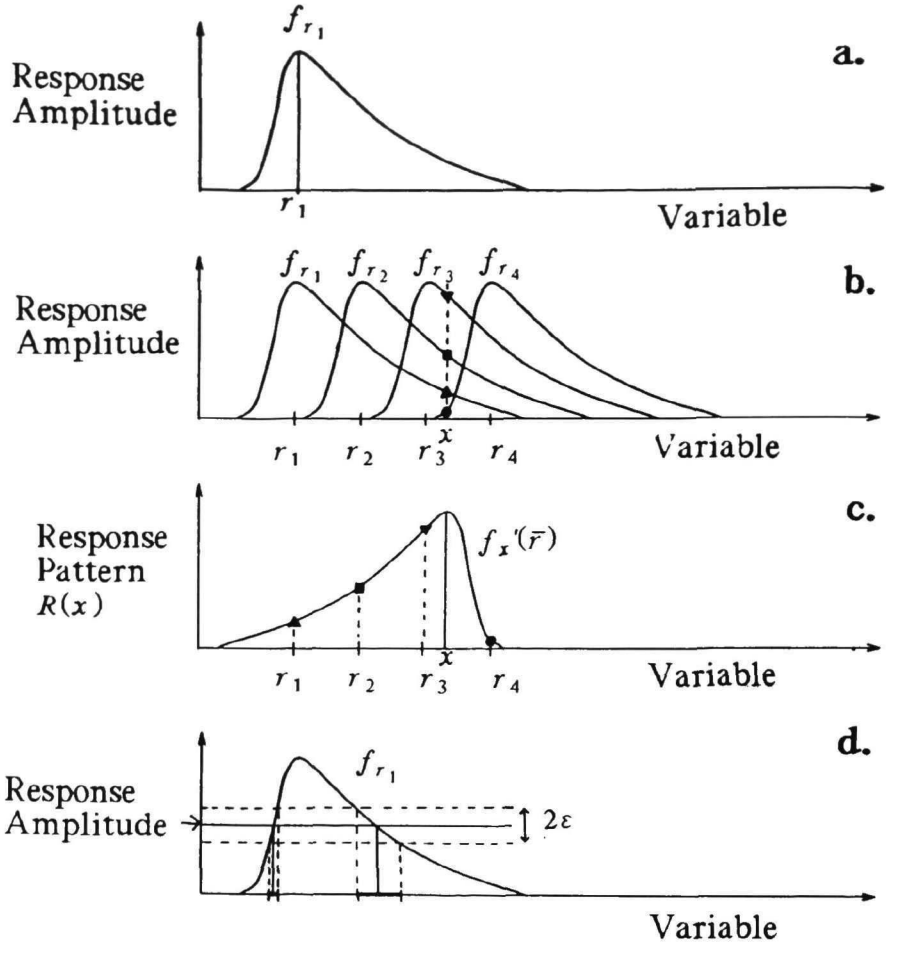

Figure 3

representation, with four response functions, one for each of the four units r_1, r_2, r_2 and r_4.

The total response pattern, $R(x)$, is the set of responses of the m different response functions, thus $R(x) = f_{r_1}(x), f_{r_2}(x), \cdots, f_{r_m}(x)$. The small shapes in Figure 3c are the response pattern which represents x, the value of the variable indicated by the arrow in Figure 3b. Note the response pattern is simply the response of each unit to the value x, replotted at the maximum of the response function (r_i) for each unit.

The mirror-image response of response function f_x to the peak values of the set of response functions is $f_x'(\bar{r}) = f_x'(r_1), f_x'(r_2), \ldots, f_x'(r_m)$, where $f_x'(r) = f_x(2x - r)$.

From these assumptions, it is possible to show that: $R(x) = f_x'(\bar{r})$, because by assumption 3, with $x = r_i$, it is seen that each element of $R(x)$ is equal to an element of $f_x'(\bar{r})$. This means that $R(x)$ contains all the information needed to obtain x, and that x is found by taking a mirror-copy of the response function, and sliding it along $R(x)$ until the best match is found. Then the maximum value of the mirror-copy will occur at x. Figure 3c show the best fit of $R(x)$, and it's subsequent indication of the value of x.

This analysis shows that only two distinct response units are required, and that they need not be orthogonally spaced over the variable space. However, the minimum permissible distance between r_1 and r_2 is a function of the measurement error, ε, and thus of γ (as defined below), with $|r_1 - r_2| \geq 2\gamma$.

The second assumption can also be relaxed, with the same general results still holding. It is only necessary that the f_i's be strictly monotonically decreasing for $r_i \leq x < y$, for y such that $f_i(y)=0$. The only additional requirement is that there is still a minimum of two nonzero responses for each value of x.

6.1.4. Accuracy of Estimate of x

The accuracy to which x can be found is a function of the accuracy to which the f_i's are defined, and the accuracy with which the $f_i(x)$'s are measured. So if it is assumed that all $f_i(x)$'s are continuously defined, there is no error in the definition of f_i. So assume the error in the measurement of $f_i(x)$ is ε.

Given the above, it is possible to determine x to within $\pm\gamma$, where $\gamma = \min_{i=0,r} (\max(y_{(i,1)}, y_{(i,2)}))$, for $y_{(i,1)}$ such that $f_i(y_1) = f_i(x) + \varepsilon$, and $y_{(i,2)}$ such that $f_i(y_2) = f_i(x) - \varepsilon$. Figure 3d shows the disjoint range of possible estimates of x that results from one particular measured response of f_{r_1}. Note that where the slope of f_{r_1} is steep, γ is small, but the shallow slope yeilds a large γ. This suggests that in order to give the most accurate estimates in the region of it's maxima, the response function should be steep in the region of it's maxima. Thus gaussian shaped response functions [12] are not desirable.

Alternatively, if f_i is sampled at intervals of δ, and $\varepsilon = 0$, then it is possible to determine x to within $\pm\delta$. In other words, under these assumptions, $\gamma = \delta$.

6.1.5. Coarse versus Fine Estimation of x

The previous discussion assumes that the goal of the computation is the accurate estimation of x, given the transfer function f_i and the total response pattern $R(x)$. However, another goal might be the rapid estimation of x, given only the total response pattern $R(x)$. One possible method for estimating x from $R(x)$ alone is: $x = r_i$, such that $f_{r_i}(x) \geq f_{r_j}(x)$, for all $j = 1, m$. This method yields only a coarse estimate of x, there being only m possible values of the estimate. The original method for estimating x yields a much more accurate or fine estimate, but at the cost of requiring more information, and a more complex computation.

When variables are represented by overlapping encodings, there are two modes in which the information can be used: the explicit, coarse representation of a variable can be used to yield a rapid estimate of the value of a variable at a resolution of k/d; while more accurate estimates must be based on the implicit fine representation, which requires more intensive processing, and yields estimates at a resolution of k. Coarse estimates of the values of a variable can be made in parallel across an image, allowing the next stages of processing to proceed in parallel. Fine estimates may require serial processing, and thus not occur automatically over all regions of an image.

As biological systems appear to use overlapping representations of feature variables, they may use the coarse mode to make rapid judgements, such as those in the preattentive, parallel stage of visual perception, while using the more complex, and perhaps serial mode to make fine judgements such as those involved in hyperacuity. For example, humans appear to be able to perform certain visual tasks in parallel, such as the discrimination of two texture regions which differ in terms of the orientation of line elements [13]. Humans can also make very fine discrimination judgements of the orientation of line segments [14]. It may be that such fine judgements require more complex processing and cannot necessarily be made in parallel.

References

1. D. Walters, "The Representation of Variables in Connectionist Networks," Proceedings of First International Conference on Computer Vision, London, England (June, 1987).
2. K. Preston and M.J.B. Duff, *Modern Cellular Automata: Theory and Applications*, Plenum, New York (1984).
3. S. Amari and M.A. Arbib, *Competition and Cooperation in Neural Nets*, Springer-Verlag, New York, NY (1982).
4. G.E. Hinton, J.L. McClelland, and D.E. Rumelhart, "Distributed Representations," pp. 77-109 in *Parallel Distributed Processing*, ed. J.L. McClelland,MIT Press, Cambridge. MA (1986).

5. Dana H. Ballard, "Cortical connections and parallel processing: Structure and function," *The Behavioral and Brain Sciences* **9** pp. 67-120 (1986).

6. W.D. Hillis, *The Connection Machine*, MIT Press, Boston (1985).

7. T. Kohonen, E. Oja, and P. Lehtio, "Storage and Porcessing of Information in Distributed Associative Memory Systems," in *Parallel Models of Associative Memory*, ed. J.A.Anderson,Earlbaum, Hillsdale, NJ (1981).

8. D.K.W. Walters, "Variable Representation in Neural Networks," Proceedings of the First International Conference on Neural Networks, San Diego, CA (June, 1987).

9. H.B. Barlow, "Single units and sensation: A neuron doctrine for perceptual psychology? ," *Perception* **1** pp. 371-394 (1972).

10. G.E. Hinton, "Shape representations in parallel systems'," Proceedings 7th IJCAI, Vancouver, BC (1981).

11. J. Sullins, "Value cell encoding strategies," U. Roch. TR 165 (1985).

12. Eric Saund, "Abtraction and representation of continuous variables in connectionist networks," Proceedings of AAAI (1986).

13. J.Beck, "Textural Segmentation," in *Organization and Representation in Perception*, ed. J.Beck,Erlbaum, Hillsdale, NJ (1982).

14. J.P. Thomas and J. Gille, "Bandwidths of orientation channels in human vision," *J. Opt. Soc. Am.* **69** pp. 652 660 (1979).