RHO-SPACE:
A NEURAL NETWORK FOR THE DETECTION AND REPRESENTATION OF ORIENTED EDGES[1]

D.K.W.Walters, Computer Science Dept., State University of New York at Buffalo, NY 14260

Abstract

This paper describes a neural network for the detection and representation of oriented edges. It was motivated both by the inherent ambiguity of convolution-style edge operators, and the processing of oriented edge information in biological vision systems.

The input to the network is the output of oriented edge operators. The computations within the network are based on orientation dependent, three-dimensional, excitatory and inhibitory neighborhoods in which computations such as lateral inhibition and linear excitation can occur.

Rho-space has a variety of interesting properties, which have been investigated. These include:
1) Both coarse and fine representation of the orientation information is possible.
2) No global thresholding is required, and the local adaptive thresholding is localized in orientation, as well as in spatial position.
3) The filling-in of dotted and dashed lines readily occurs.
4) There is a natural representation of connectivity, which agrees with human perception.
5) Illusory contours, of one type produced by the human visual system are produced.
6) All processing is completely data-driven, and no domain dependent knowledge or model based processing is used.

1. Introduction

The most universally applied stage of low-level visual processing is the detection of image edges, be they intensity edges, motion edges or texture edges. Yet there are some basic theoretical problems which complicate the detection and representation of image edges. One such problem is the inherent ambiguity in the response of any single convolution-style edge operator; the operator responds to the conjunction of edge location, orientation, amplitude, etc., and the values of these contributing factors can not be untangled from a single response. In this paper it is argued that the ambiguity problem lies not with edge operators themselves, but with how the operator responses are being interpreted in current computer vision systems. For example, neural-based biological vision systems use convolution-style, oriented edge operators and appear to have solved the ambiguity problem. This suggests that a solution exists using the style of representation and computation possible in neural networks.

This paper explores the use of a neural network for the detection and representation of oriented edges. Grossberg and Mingolla [1] have also addressed the edge detector ambiguity problem through the use of networks which model neural computations at the level of the dynamic, cooperative and competitive interactions of feedback, shunting, etc. Our research differs in two main ways: first, the neural networks studied perform static, noniterative, discrete computations of the type that could be easily implemented in a clocked, discrete, parallel digital architecture; and second, emphasis is placed on the type of distributed representation that is possible in such networks.

2. Representations for Oriented Edges

Although there is considerable debate amongst vision researchers as to the existence of an optimal edge detector [2, 3, 4, 5], there is general agreement about how oriented edges should be represented. The standard representation consists of an amplitude or gradient image, $A(x,y)$, in which each point represents the amplitude or gradient of the edge at spatial position (x,y); and an orientation image

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

$B(x,y)$, in which the value of each point represents the orientation or gradient direction at spatial position (x,y). The implicit assumption is that the local edge amplitude and orientation at each image point can be measured. But the ambiguity problem invalidates this assumption for the response of individual edge operators. One potential solution is to look at the response of a set of edge operators at each image location, such as set of oriented operators. It will be shown that the response of the set as a whole contains significant structural information. But most techniques using oriented edge operators use either thresholding or local averaging to produce a single estimate of edge amplitude and orientation from the set, and this loses the structural information. This point has been made by Zucker [6], who proposes a model-matching scheme for reconstructing the edge information from the responses. A model-free approach is taken here which uses a distributed representation of oriented edge information, ρ-space, in which the responses of oriented edge operators for a single image point are not combined through thresholding or other techniques, which allows the subsequent computations to disambiguate the orientation, position and amplitude information.

3. Rho-Space Representation

Rho-space is a three dimensional space, where the x and y dimensions represent the spatial dimensions of an image, and the third dimension, ρ, represents the orientation of image contours (intensity edges, texture boundaries, lines, etc.). The space is discretized in all dimensions.

The input to ρ-space is currently produced by convolving an image with a set of Canny-type oriented edge operators [2], of either 8 or 18 separate orientations. Figure 1 shows a diagram of the ρ-space. Each orientation plane shown in Fig. 1 can be thought of as the result of convolving the image with an edge operator of a given orientation. The value at each location of a single orientation plane is then the amplitude of the output of that particular operator at that image location. Figure 1 shows 6 orientation planes, but in the computer implementation either 8 or 18 orientation planes were used.

The algorithms developed for the ρ-space representation assume that there is a simple processor associated with each point or pixel in the space, and each processor is locally connected only to those processors in its three-dimensional neighborhood (as defined below). Each simple processor is actually a network of neural-type units, but can be simulated as single processor which runs a simple internally stored program. The ρ-space processors perform local, noniterative computations such as discrete forms of lateral inhibition, short and long range linear excitation, and short and long range linear inhibition. These computations put the oriented edge information into a form that is usable by a wide range of

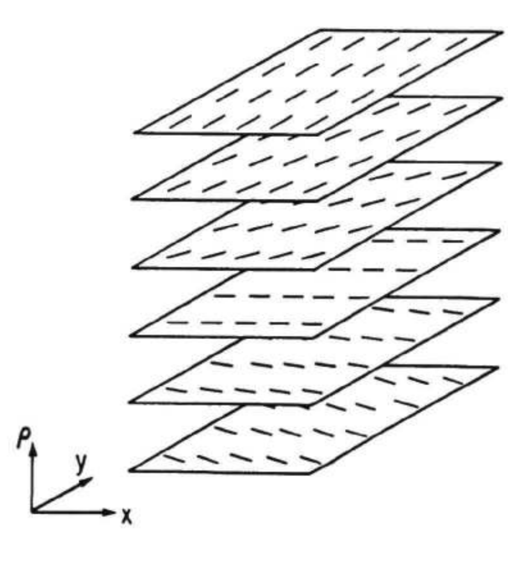

Figure 1

low and intermediate level visual computations such as perceptually based enhancement, segmentation and grouping [7]. Further descriptions of the ρ-space representation can be found in [8,9].

3.1. Definitions

Assume there are k processors in ρ-space, and define x_j to be the activity of processing unit j.

3.2. Excitatory and Inhibitory Neighborhoods in Rho-Space

All ρ-space computations, and the definitions of oriented lines in ρ-space are based on the concept of the local neighborhood of each point in ρ-space. The excitatory neighborhood, E_j, of point j includes all points which are directly connected to j, and which participate in excitatory operations. The inhibitory neighborhood, I_j, of point j consists of all points which are directly connected to j, and which participate in inhibitory computations.

All E_j and I_J are defined as functions of the edge operators used to generate the input to ρ-space. The two spatial dimensions of the neighborhoods, n by m, are the spatial dimensions of the convolution kernels of the edge operators. The orientation dimension, d, of the neighborhoods are determined by the number of separate orientations represented in the convolution kernels. For example, for sixteen oriented 7 by 7 operators of the type illustrated in Figure 2a, the excitatory neighborhood for the central solid horizontal pixel is that shown in Figure 2b, where the non-empty circles represent the locations in the excitatory neighborhood. Figure 2b shows just a small portion of ρ-space, part of each of three consecutive orientation planes. The middle plane corresponds to horizontal edges (0 degrees), while the top and bottom planes contain information about orientations of +22.5 and -22.5 degrees respectively. The excitatory neighborhood of each point in ρ-space lies within a small rectangular box shaped region of ρ-space.

The inhibitory neighborhood, I_j, of point j is the compliment of the excitatory neighborhood, E_j, defined over the n by m by d space centered on the point j.

Each neighborhood can be divided into two halves by a plane which passes through the central point, and which is orthogonal to the orientation direction of the central point. Let $E_{j,1}$ and $E_{j,2}$ refer to the two halves of E_j.

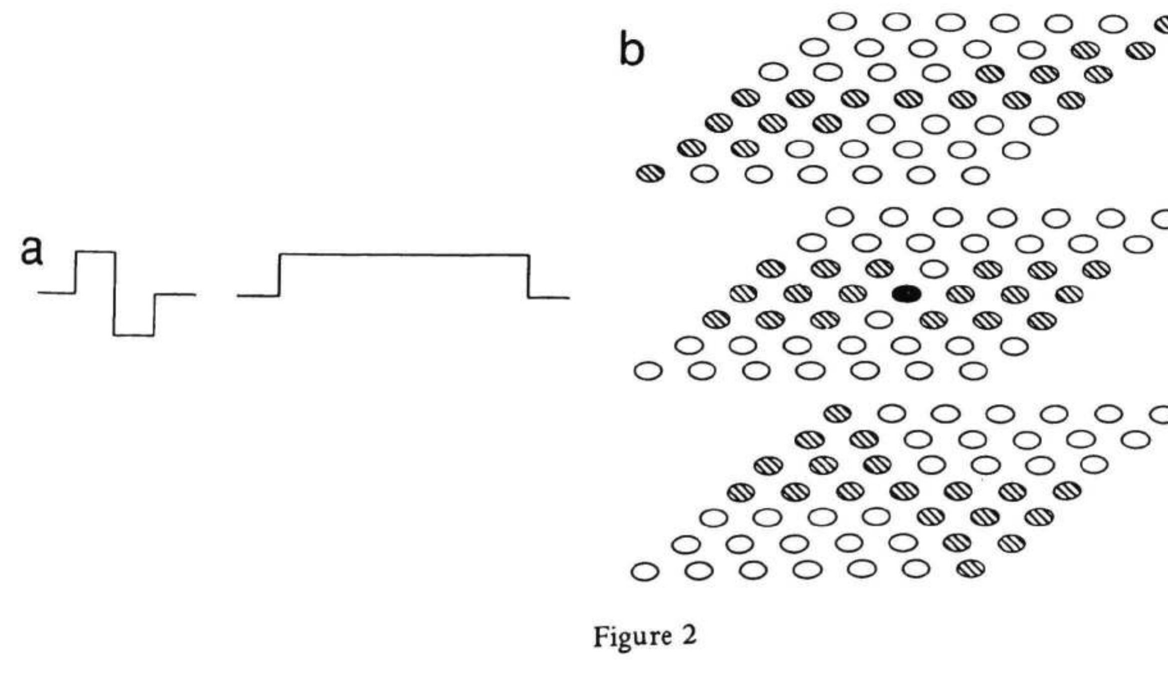

Figure 2

4. Computations Using the Rho-Space Representation

The computations in ρ-space are all local computations which involve only a single processor and it's local neighbors. Four such computations are lateral inhibition (LI), short-range linear inhibition (SRLI), short-range linear excitation (SRLE), and mid-range linear inhibition (MRLI). Each processor, x_j, is composed of 5 neural units: O_j, is the output of the oriented edge operators; T_j is the output of LI, L_j is the output of SRLI; F_j is the output of SRLE; and R_j is the output of MRLI. For example, LI is defined :

$$T_j = O_j \,(\max((O_j - M_{O,I_j}), 0)/(O_j - M_{O,I_j}))$$

where $M_{O,I_j} = \max_i (O_i \textit{ st } O_i \textit{ is an element of } I_j)$.

SRLI is defined as :

$$L_j = T_j \,(\max(-M_{T,E_j}, 0)/-M_{T,E_j})$$

where, $M_{T,E_j} = \max_i (T_i \textit{ st } T_i \textit{ is an element of } E_j)$.

All four computations are illustrated in Figure 3 where (a) shows a black-white checkerboard image in which uniform noise has been added to each pixel, and part (b) shows the nonzero pixels present in the input image. In part (c) each white pixel indicates that at least one of the oriented operators had a non-zero response at that image point. Thus, the operators are indicating that all but one of the points in this image could be an edge point. A common method for interpreting such responses is to use either a global or a local threshold to remove unwanted responses, but it is generally not possible to find a threshold which removes all of the noise-generated responses, and none of the edge-generated responses. Part (d) shows the results after LI, which is a kind of local adaptive thresholding, but which has the added advantage that it is orientation selective. Thus a high amplitude horizontal edge does not inhibit a neighboring low amplitude vertical edge. Part (e) shows the results after SRLI which has the function of removing potential edge points which are not connected to other edge potential points, and thus could not have arisen from true edges. Part (f) shows the small gaps in the lines representing edges being filled in by SRLE, while part (g) shows the short, unconnected lines removed after MRLI, yielding the connected edges of the checkerboard. (More details of these excitatory and inhibitory interactions can be found in [10]).

5. Definitions of Lines in Rho-Space

As one of the goals for computations in ρ-space is to group local edges into a form which represents more global image edges, a means of defining an image edge is required. If we were working in euclidean space, then we might define an image edge as a connected line of local edges, but in ρ-

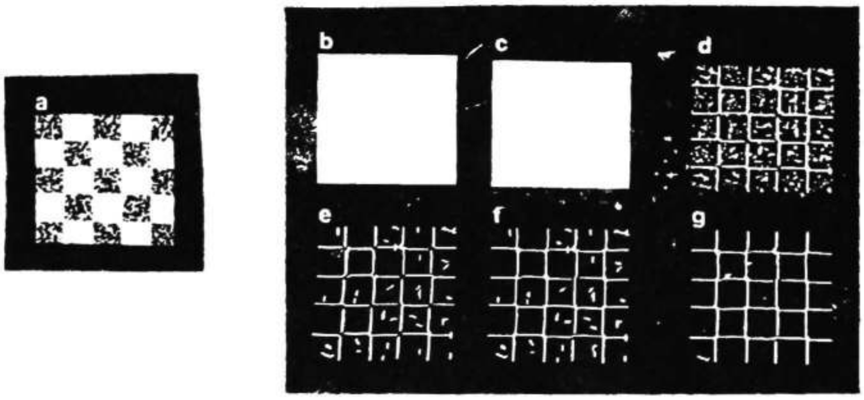

Figure 3

space the euclidean definitions of connectivity and of lines do not hold. Thus the following ρ-space definitions are required.

A point j is connected to another point j' if and only if both are nonzero, and j lies within $E_{j'}$ and j' lies within E_j.

(Note that two processors in rho-space can be connected, without the edge points associated with them being connected.)

A line is a set of connected points which lack neighbors in their inhibitory neighborhoods.

A line end is a line point, j, which has neighbors in either $E_{j,1}$ or $E_{j,2}$, but not in both.

6. An Example of Grouping and Segmentation

From the above definitions, it is possible to group local edge points in ρ-space into more global image edges or lines, and from structural information about line ends, it is also possible segment an image into sets of lines which are likely to have arisen from a single object [7]. An example of this process is seen in Figure 4, where parts (a) through (f) show the results of the ρ-space computations described above, while parts (g) and (h) each show one of the two segments generated by the segmentation algorithm. Note that although the circle and the rectangle are connected in the image space, they are not connected in ρ-space, which aids in their segmentation.

7. Rho Space Properties

Rho-space has a variety of interesting properties, which have been investigated. These include:
1) Both coarse and fine representation of the orientation information is possible.
2) No global thresholding is required, and the local adaptive thresholding is localized in orientation, as well as in spatial position.
3) The filling-in of dotted and dashed lines readily occurs.
4) There is a natural representation of connectivity, which agrees with human perception.
5) Illusory contours, of one type produced by the human visual system are produced.
6) All processing is completely data-driven, and no domain dependent knowledge or model based processing is used.

As an example, consider the illusory contour property. One of the properties of the current implementation of ρ-space is that the response to a single line, is the line itself, and two orthogonal end lines. This behavior was pointed out by Marr and Hildreth as being an undesirable side effect of oriented edge operators [11]. However, the orthogonal end lines might play a positive role in perception the formation of illusory contours. For example, when the pattern in Fig.5a is viewed from the

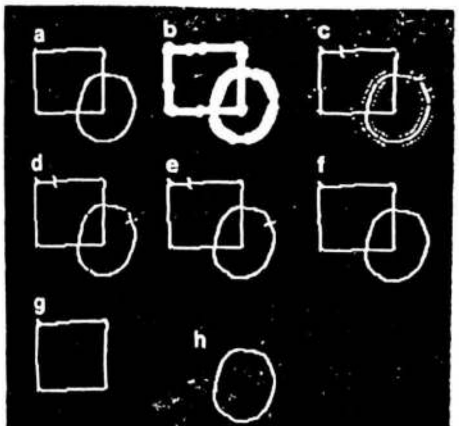

Figure 4

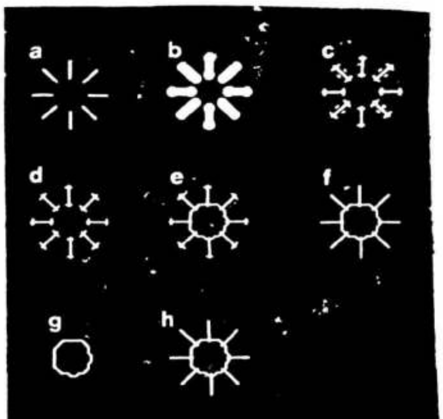

Figure 5

appropriate distance, the central region may appear "darker" than the rest. This can be explained in part in terms of an illusory circular contour being formed, and the orthogonal end lines can provide such a contour. To illustrate this, an example ρ-space computation is shown in Figure 5. Part (a) shows the input pattern. Part (b) shows the nonzero convolution responses, and part (c) shows the nonzero responses after LI. Part (d) shows the nonzero responses after SRLI, and the orthogonal end lines are apparent. Part (e) shows the nonzero responses after the SRLE, and a roughly circular contour which may correspond to the illusory contour is formed. In part (f) the short unconnected line segments have been removed by MRLI to form the final percept. This is an interesting result because an illusory contour has been formed in a completely data-driven manner, without reference to models, or inferring depth.

Grossberg has shown that the illusory contour formed by this pattern disappears when the individual lines are rotated by 45 degrees about their interior end points. Under such rotation the orthogonal end lines of the lines would not join to form a closed contour, and no illusory contour would be formed, thus supporting the orthogonal end line hypothesis for this illusion.

References

1. S. Grossberg and E. Mingolla, "Neural Dynamics of surface perception: Boundary-webs, illuminants and shape from shading," *Comp. Vision, Graph. & Image Proc.* **37**(1) pp. 116-165 (1987).
2. John F. Canny, "Finding Edges and Lines in Images," MIT Master's thesis (1983).
3. R.M. Haralick, "Digital step edges from zero crossing of second directional derivatives," *IEEE Trans. Pattern Anal. Machine Intell.* **PAMI-6** pp. 58-68 (1984).
4. S.E.L. Grimson and E.C. Hildreth, " "Comments on "Digital tesp edges from zero crossings of second directional dirivatives," *IEEE Trans. Pattern Anal. Machine Intell.* **PAMI-7** pp. 121-127 (January, 1985).
5. V. Torre and T.A. Poggio, "On edge detection," *IEEE T-PAMI*, pp. 147-163 (March, 1986).
6. S. Zucker, "Early Process for Orientation Selection and Grouping," pp. 170-200 in *From Pixels to Predicates*, ed. A.P. Pentland,Ablex, Norwood, NJ (1986).
7. D.K.W.Walters, "Selection of image primitives for general-purpose visual processing," *Computer Vision Graphics & Image Processing*, ((in press)).
8. D.K.W.Walters, "Selection and use of image features for segmentation of boundary images," Proceedings IEEE CVPR (1986).
9. D.K.W.Walters, "A Computer Vision Model Based on Psychophysical Experiments," pp. 88-120 in *Pattern Recognition by Humans and Machines*, ed. H.C.Nusbaum,Academic Press, New York, NY (1986).
10. D.K.W.Walters, *Parallel Computations in Rho-Space*. In preparation.
11. D. Marr and E. Hildreth, "Theory of edge detection," *Proceedings of the Royal Society of London B* **207** pp. 187 217 (1980).