MASSACHUSETTS INSTITUTE OF TECHNOLOGY
ARTIFICIAL INTELLIGENCE LABORATORY

and

CENTER FOR BIOLOGICAL INFORMATION PROCESSING
WHITAKER COLLEGE

A.I. Memo 751
C.B.I.P. Paper 011

June, 1985

Analog "Neuronal" Networks in Early Vision

Christof Koch, Jose Marroquin and Alan Yuille

Abstract: Many problems in early vision can be formulated in terms of minimizing an energy or cost function. Examples are shape-from-shading, edge detection, motion analysis, structure from motion and surface interpolation (Poggio, Torre and Koch, 1985). It has been shown that all quadratic variational problems, an important subset of early vision tasks, can be "solved" by linear, analog electrical or chemical networks (Poggio and Koch, 1985). In a variety of situations the cost function is non-quadratic, however, for instance in the presence of discontinuities. The use of non-quadratic cost functions raises the question of designing efficient algorithms for computing the optimal solution. Recently, Hopfield and Tank (1985) have shown that networks of nonlinear analog "neurons" can be effective in computing the solution of optimization problems. In this paper, we show how these networks can be generalized to solve the non-convex energy functionals of early vision. We illustrate this approach by implementing a specific network solving the problem of reconstructing a smooth surface while preserving its discontinuities from sparsely sampled data (Geman and Geman, 1984; Marroquin, 1984; Terzopoulos, 1984). These results suggest a novel computational strategy for solving such problems for both biological and artificial vision systems.

This report describes research done within the Artificial Intelligence Laboratory and the Center for Biological Information Processing (Whitaker College) at the Massachusetts Institute of Technology. The Center's support is provided in part by the Sloan Foundation and in part by the Whitaker College. Support for the A.I. Laboratory's research is provided in part by the Advanced Research Projects Agency of the Department of Defense under Office of Naval Research Contract N00014-80-C-0505. JM is supported by the Army research Office under contract ARO-DAAG29-84-K-0005. CK is supported by a grant from the Office of Naval Research, Engineering Psychology Division.

1. Introduction

The study undertaken in this manuscript has its origin in two different areas: computational vision and neuronal networks. Problems in early vision, such as computing depth from two stereoscopic images, reconstructing and smoothing images from sparsely sampled data, computing motion etc. are inherently difficult to solve, although to us they seem effortless. Within the last years, computational studies have provided promising — although far from complete — theories of the computations necessary for early vision (for partial reviews see Marr, 1982; Brady, 1982; Ballard, Hinton and Sejnowski, 1983; Poggio, Torre and Koch, 1985). Early vision consists of a set of processes that recover physical properties of the visible three-dimensional surfaces from the two-dimensional intensity arrays (Marr, 1982). A number of these tasks can be described within the framework of standard regularization theory (Poggio and Torre, 1984; Poggio *et al.*, 1985): edge detection, smooth surface interpolation and computing the smoothest velocity field. Standard regularization analysis can be used to solve them in terms of quadratic energy functionals which must be minimized, subject to certain constraints. Previous work by Poggio and Koch (1985; see also 1984) showed how to design linear, analog networks for solving regularization problems with quadratic energy functions. The domain of applicability of standard regularization theory is limited, however, by the convexity of the energy functions which makes it impossible to deal with problems involving true discontinuities without introducing new concepts (Poggio *et al.*, 1985). Such problems can be described by non-convex energy functions involving line processes (Geman and Geman, 1984; Marroquin, 1984; see Blake, 1983). Methods proposed for minimizing these include simulated annealing (Kirkpatrick, Gelatt and Vecchi, 1983; Geman and Geman, 1984) and graduated non-convexity (Blake, 1983). More recently Marroquin (1985b) has proposed a different approach, based on the use of Markov Random Fields (MRF) models and Bayes estimation theory, in which the solution to early vision problems is not expressed in terms of minimizing an energy functional. The resulting algorithms, however, can also be implemented in analog and hybrid networks such as the ones we describe here.

There has been considerable interest in recent years in the computational properties and capabilities of networks of simple, neuronal-like elements (for instance, Kohonen, 1977; Marr and Poggio, 1977; Ullman, 1979; Hopfield, 1982, 1984; Palm, 1982). More recently, Hopfield and Tank (1985) have shown that analog neuronal networks can provide fast, next-to-optimal solutions to a well characterized, but difficult, optimization problem, the Travelling Salesman Problem. In this paper we show that highly interconnected networks of simple, analog processing elements can be used to give fast solutions to a number of early vision problems. Apart from their intrinsic interest as neuronal models, these networks may have important practical applications for computer vision. Their ability to perform computations by analog elements implies fast convergence times with respect to the basic cycle time of digital hardware. For many problems in early vision, a fast approximate answer is more valuable than a slow correct one. The use of such networks in combination with massive parallel computers offers the possibility of real time artificial vision systems. One caveat. We do not, of course, equate our impoverished model of a "neuron" with the complexities of "real" neurons. Our neuronal networks do, however, share several important properties with nerve cells: high connectivity, analog mode of operation and tolerance to hardware failures.

2. Smooth Surface Reconstruction

Surface reconstruction is a typical problem of early vision that can be formulated in terms of minimizing a quadratic energy function (Grimson, 1981, 1982; Terzopoulos, 1983). It occurs in several situations. For example, if a stereo algorithm computes depth values only at specific locations in the image, for instance along edges (*i.e.*, at zero crossings of the convolution of the image with the Laplacian of a Gaussian Operator), the surface must be interpolated between these points. Another instance occurs when the data is given everywhere but is noisy and needs to be smoothed.

Grimson (1981, 1982) studied surface interpolation in the context of stereo matching. He considered a stereo algorithm (Marr and Poggio, 1979) in which isolated primitive features, zero crossings — corresponding to significant events in the images — were matched yielding a depth value at the feature points. He then proposed an interpolation scheme to obtain depth values throughout the image. A partial justification for the use of interpolation schemes in human vision comes from studies of random dot stereograms. Here data is only given at isolated dots and yet the perception is of a smooth surface, even when the density of dots is very low. Grimson's interpolation scheme involves minimizing a quadratic energy function and can be described as fitting a thin flexible plate through the observed data. Both Grimson's and Terzopoulos's (1983) interpolation scheme (1983) can be described in terms of regularization theory. The energy or cost function $E(x)$ to be minimized — subject to certain constraints which derive from a physical analysis of the problem under consideration — is given by:

$$E(x) = ||Bx - b||^2 + \lambda||Sx||^2 \tag{1}$$

where x is the vector representing the image points and B and S are matrices. The first term gives the distance of the solution to the data and the second term corresponds to the regularizer needed to make the problem well-posed. For surface interpolation, the elements of B are equal to 1 at those locations where the depth is known and 0 at all others.

The stabilizer S corresponds to the operator associated with a membrane or thin plate, depending on the kind of smoothing desired. E can be reformulated as

$$E(x) = x^T(B^TB + \lambda S^TS)x - 2x^TBb + b^Tb. \tag{2}$$

This expression can be transformed into

$$L(V) = \frac{1}{2}\sum_{ij} T_{ij}V_iV_j + \sum_i V_iI_i \tag{3}$$

by identifying the matrix T with $2(B^TB + \lambda S^TS)$, V_i with x, I_i with $-2Bb$ and dropping the constant term b^Tb. We can interpret this expression as the Lyapunov function of a neuronal network, with linear input-output characteristic. Thus, the "voltage" V_i corresponds to the output of the processing element i, henceforth termed neuron i, I_i is the current being injected into neuron i and T_{ij} is the strength of connectivity between neurons i and j. If no connection exists between two neurons, the appropriate entry in T_{ij} is set to zero. If every neuron has an associated capacitance C_i, its output V_i will be updated according to:

$$C_i\frac{dV_i}{dt} = -\frac{\partial L}{\partial V_i}. \tag{4}$$

Note that this update corresponds to finding the minimum of a convex function using steepest descent. This rule ensures that L always decreases with time and hence corresponds to a Lyapunov function. For the case of quadratic regularization principles, this function is positive definite quadratic and so the system will always converge (except in some pathological cases) to the unique energy mimimum. In other words, every quadratic variational principle of the type shown in equation (2) can be solved with an appropriate neuronal network, where the connections can be implemented by linear Ohmic resistances and the data is given by injecting currents. A similar result, using slightly different circuit components, was derived by Poggio and Koch (1985).

3. Line Processes

However, quadratic variational principles have limitations. The main problem is the degree of smoothness required for the unknown function that is to be recovered. For instance, the surface interpolation scheme outlined above smoothes over edge discontinuities and often leads to unrealistic results (figure 1).

Marroquin (1984) has proposed a scheme to overcome this difficulty (see also Blake, 1983; Terzopoulos, 1985). Following Geman and Geman (1984), he used a probabilistic

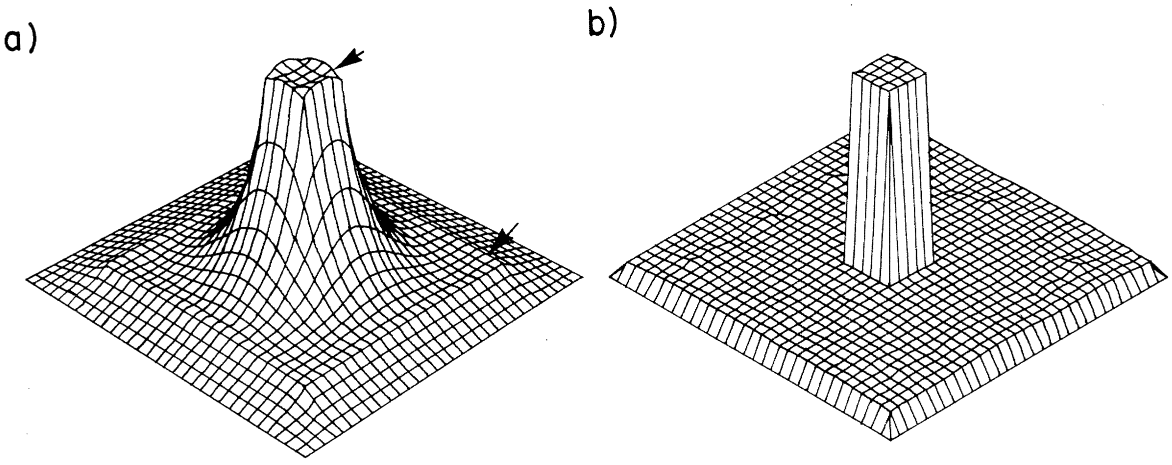

Figure 1. Smooth and piecewise smooth surface reconstruction from noisy and sparse data. (a) Three dimensional representation of a reconstructed and smoothed surface from sparse observations using a quadratic energy expression corresponding to interpolating with a membrane. Depth measurements are only available along the ridge in the main plane of the figure and along the rim of the "tower" (see arrows). Along these contours, on the average every second point is sampled. The observations are assumed to be corrupted by a Gaussian noise ($\sigma_i = 0.25$). The state of the system is show at 0.5τ. (b) Piecewise smooth surface reconstruction from the same data set at time τ. The image is clearly segmented into two distinct domains. Both surfaces are computed using our network with constant coupling. For parameters, see text.

formulation of the surface reconstruction problem; the behavior of a piecewise smooth surface is modeled using two coupled Markovian Random fields (Kindermann and Snell, 1980): a continuous-valued one that corresponds to the depth, and a binary one, whose variables are located at sites between the depth lattice (see figure 2a). The function of the unobservable "line process" is to indicate the presence or absence of a discontinuity between two adjacent depth elements. Using Bayes theory, it is found that the maximum *a posteriori* estimate of the surface corresponds to the global minimization of the "energy" function:

$$E(f,l) = \sum_{i,j}(f_i - f_j)^2(1 - l_{i,j}) + c_l \sum_i (f_i - d_i)^2 + \sum_i V_C(l), \qquad (5)$$

where the term $V_C(l)$ (the "potential" of the line process l) measures the cost that has to be paid for the introduction of specific local configurations of lines and embodies the prior knowledge about the geometry of the discontinuities (for instance, the fact that they occur along piecewise smooth curves that only rarely intersect).

In one dimension the energy function is given by

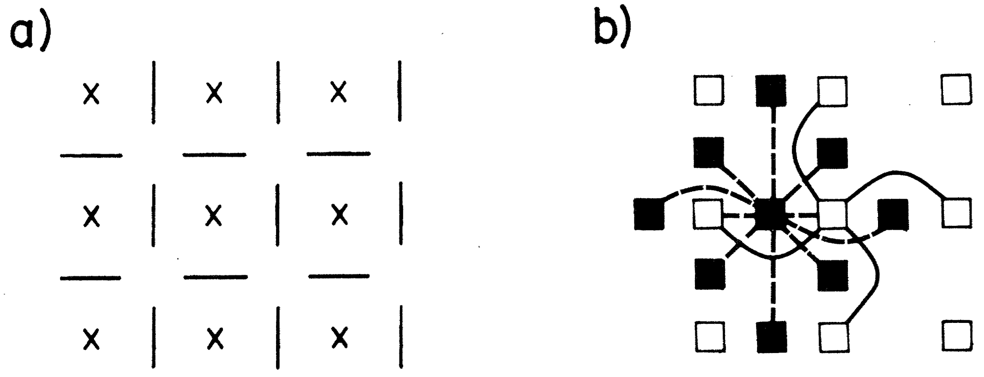

Figure 2. (a) The two-dimensional lattice of line processes (lines) and depth points (crosses). Each depth value is enclosed on all four sides by a line process. (b) The local connections between neighouring line processes (filled squares) and the depth lattice (open squares). Notice that the quadratic part of the network, i.e. the depth lattice, is only connected to its four neighbours, while every line process, except at the boundaries, is connected to 10 neighbouring locations.

$$E(f,l) = \sum_i (f_{i+1} - f_i)^2(1 - l_i) + c_d \sum_i (f_i - d_i)^2 + c_l \sum_i l_i. \tag{6}$$

Here the l_i correspond to the line process. Observe that if the gradient of f becomes too large (i.e. $(f_{i+1} - f_i)^2 > c_l$), it becomes cheaper to break the surface, and put in a line — paying the "price" c_l — rather than to interpolate smoothly. This line process l_i introduces local minima into the energy function, making the problem highly non-linear. The term $(f_i - d_i)^2$, describing the difference between the measured data d_i and the approximated surface value f_i, is weighted by c_d, which depends on the signal-to-noise ratio. If d_i is very reliable, then $c_d >> 1$. For two-dimensional images more terms are required in equation (6) (see, equation (11)). In Marroquin's study (1984), the potentials of the binary line process were implemented by a table lookup procedure and the minimization of the resulting combinatorial optimization problem was carried out using simulated annealing.

Here we sketch another method based on Hopfield results (1982, 1984; see also Hopfield and Tank, 1985; for more details see Marroquin, 1985a). Hopfield's idea was to solve combinatorial optimization problems by allowing the binary variable to vary continuously between 0 and 1 and to introduce terms in the energy function that forced the final solution to one of the corners of the hypercube $[0,1]^N$.

Briefly, let the output variable V_i for neuron i have the range $0 < V_i < 1$ and be a continuous and monotonic increasing function of the internal state variable u_i of the neuron i: $V_i = g_i(u_i)$. A typical choice is

$$g(u_i) = \frac{1}{1+e^{-2\lambda u_i}}. \tag{7}$$

The neurons are highly interconnected. The strength of the connection between i and j is given by the matrix element $T_{i,j}$. Furthermore, each neuron has its own input capacitance C_i and transmembrane resistance R_i. The resulting charging equation that determines the rate of change of u_i is

$$C_i \frac{dV_i}{dt} = \sum_j T_{i,j} V_j - \frac{u_i}{R_i} + I_i, \tag{8}$$

where I_i can be considered as fixed input to neuron i. Hopfield introduces the quantity

$$E = -\frac{1}{2} \sum_{i,j} T_{i,j} V_i V_j + \sum_i \frac{1}{R_i} \int_0^{V_i} g_i^{-1}(V) dV + \sum_i I_i V_i, \tag{9}$$

and shows that E is a Lyapunov function of the system, as long as $T_{i,j}$ is symmetric. In other words, using the update of equation (8):

$$C_i \frac{dV_i}{dt} = -\frac{\partial E}{\partial V_i}, \tag{10}$$

the time evolution of the system is a motion in state space that seeks out minima in E and comes to a stop at such points. The relation between the stable states of the continuous model and those of the binary ones, in which the output of every neuron can be either 0 or 1, is governed by λ. For $\lambda \to \infty$, g_i tends to either 0 or 1.

In order to be able to use these analog networks for piecewise smooth surface reconstruction, we have to map the binary line processes l_i into continuous variables bounded by 0 and 1. One possibility for choosing an appropriate continuous energy expression is outlined below. The energy function has four contributors: the interpolation term, E_I, the data term, E_D, the line potential term E_L (corresponding to the potential $V_C(l)$ in (5)) and the gain term E_G:

$$E_I = \sum_{i,j} \left((f_{i,j+1} - f_{i,j})^2 (1 - v_{i,j}) + (f_{i+1,j} - f_{i,j})^2 (1 - h_{i,j}) \right) \tag{11a}$$

$$E_D = \frac{c_D}{2} \sum_{i,j} (f_{i,j} - d_{i,j})^2. \tag{11b}$$

$$E_L = c_V \sum_{i,j} \left(v_{i,j}(1 - v_{i,j}) + h_{i,j}(1 - h_{i,j}) \right) + c_P \sum_{i,j} (v_{i,j} v_{i,j+1} + h_{i,j} h_{i+1,j}) + c_C \sum_{i,j} (v_{i,j} + h_{i,j})$$

$$+c_L \sum_{i,j} v_{i,j} \cdot \left((1 - v_{i+1,j} - h_{i,j} - h_{i,j+1})^2 + (1 - v_{i-1,j} - h_{i-1,j} - h_{i-1,j+1})^2 \right) \qquad (11c)$$

$$+c_L \sum_{i,j} h_{i,j} \cdot \left((1 - h_{i,j+1} - v_{i,j} - v_{i+1,j})^2 + (1 - h_{i,j-1} - v_{i,j-1} - v_{i+1,j-1})^2 \right)$$

$$E_G = c_G \sum_{i,j} \Big(\int_0^{v_{i,j}} g_i^{-1}(v) dv \Big) + \Big(\int_0^{h_{i,j}} g_i^{-1}(h) dh \Big) \qquad (11d).$$

Here $f_{i,j}$, $v_{i,j}$ and $h_{i,j}$ correspond to the depth, the vertical line process and the horizontal line process respectively. The first term in E_L forces the line process to the corners of the hypercube, i.e. to either 0 or 1. The second term penalizes the formation of adjacent parallel lines while the third term represents the cost that need be paid for the introduction of every single line. The fourth term is an interaction term which favours continuous lines and penalizes both multiple line intersections and discontinuous line segments. The gain term forces the line process inside the hypercube $[0,1]^N$. Figure 2b illustrates the connections required to implement this energy function within the line and the depth lattices. Following Hopfield (1984), we choose the following update rule:

$$\frac{df_{i,j}}{dt} = -\frac{\partial E}{\partial f_{i,j}} \qquad (12a)$$

$$\frac{dm_{i,j}}{dt} = -\frac{\partial E}{\partial v_{i,j}}, \qquad (12b)$$

$$\frac{dn_{i,j}}{dt} = -\frac{\partial E}{\partial h_{i,j}}, \qquad (12c)$$

where $m_{i,j}$ and $n_{i,j}$ are the internal state variables for the processing elements corresponding to the vertical and horizontal line processes; that is $v_{i,j} = g(m_{i,j})$ and $h_{i,j} = g(n_{i,j})$. It is easy to see that for this update the total energy will always decrease.[1]. The system will evolve in such a manner as to find a minimum of E. Our energy function E differs from the energy function chosen by Hopfield and Tank (1985) in two important aspects. Firstly, our energy function contains cubic terms (e.g. $f_{i,j+1} \cdot f_{i,j} \cdot v_{i,j}$) which implies quadratic terms in the update rule, different from Hopfield's linear update rule. Secondly, our network consists of two networks: the first one corresponding to the inherently continuous surface depth and the second one associated with the inherently binary line processes. We will demonstrate empirically in the next section that this system seems to find next-to-optimal solutions to the surface reconstruction problem, even though no formal proof showing the convergence to the global minimum of E exists.

[1] Note that the energy gain term E_G makes a simple contribution proportional to $-m_{i,j}$ and $-n_{i,j}$ to the right hand side of (12b) and (12c) and no contribution at all to (12a).

4. Simulation Results and Heuristics

An analog network for a 32 by 32 image was simulated on a digital computer. Preliminary explorations of parameter values showed that the performance of the system depends on the relative weight of the different terms in the energy function. In particular, two sets of distinct parameters are important. First, the weight of the term describing the interaction among the line processes, E_L, versus the weight of the smoothing term, in our case equal to 1, and secondly, the relative weight of the different components of E_L and E_G, *i.e.*, c_V, c_L, c_P, c_C and c_G. The first ratio ultimately determines the limiting depth gradient beyond which no more interpolation takes place. Decreasing the importance of the line interaction term E_L versus the smoothing term encourages the formation of lines at smaller and smaller depth gradients. Thus, this number requires some rough estimate of the limiting depth gradient for which no smooth interpolation should occur. The second set of parameters determines to what extent adjacent parallel lines form, intersection of lines occur etc. We determined a parameter set giving reasonable solutions. Fortunately, the choice of these parameters does not seem to vary from image to image. Results in this paper refer to parameters set at $c_V = 0.5, c_L = 4.0, c_P = 5.0, c_C = 1.0$ and $c_G = 0.5$. As boundary condition we choose to set horizontal lines at the two horizontal boundaries of the square image and vertical lines at the vertical boundaries. Thus, the image is effectively decoupled from the outside. As initial conditions we set the internal state variable of the line process neurons, $m_{i,j}$ and $n_{i,j}$, to 0 and all horizontal and vertical lines to 0.5. In other words, the initial starting point is the middle of the hypercube $[0,1]^N$, explicitly biasing no single position.

The final state of the network should approximate as closely as possible the state of lowest energy. Since f and l are independent variables, the solution set can be found by minimizing $E(f,l)$ for a given fixed arrangement of the line processes by varying f. These considerations dictate the following strategy. After initializing the depth lattice with the sparsely sampled depth data, the network computes the smoothest surface assuming all line processes set to 0. Thus, the initial state of the network is an everywhere smooth surface. This process converges in about two to three time constants. Subsequently, the depth network is updated ten times for every single update of the line process network. Functionally, this is equivalent to assuming that the depth network is stationary with regard to the line process network. In other words, the time constant of the depth network is a tenth of the time constant of the line process network $\tau = C_i R_i$. In the following, we will always refer to the elapsed time in terms of τ. The time step for the differentiation was set to 0.01τ.

Figure 1 illustrates dramatically the difference between smooth and piecewise smooth surface interpolation. Note, that depth measurements are available only at every second

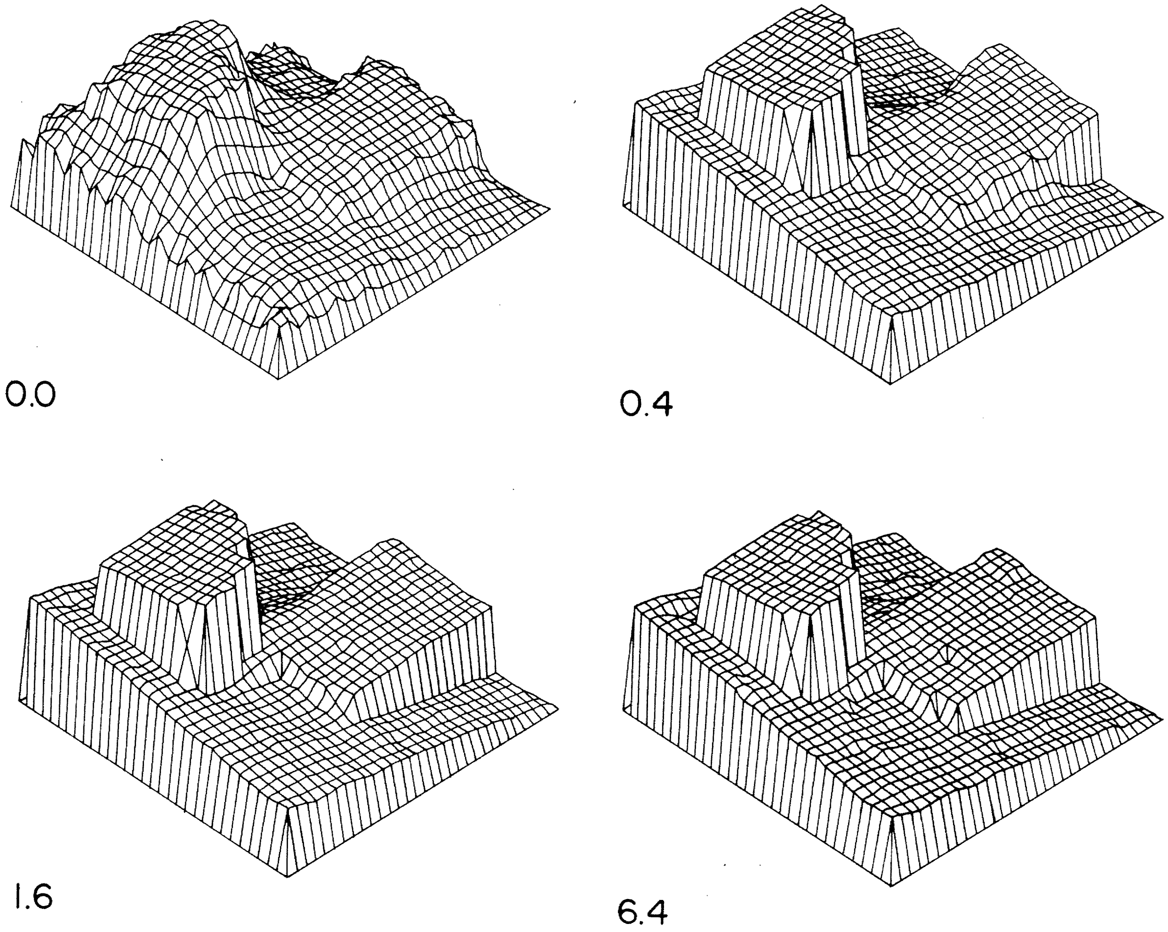

Figure 3. Temporal evolution of the states of the network for a sparsely sampled synthetic scene containing three slanting rectangles. Interpolation using a membrane-type energy function. On the average every third point is sampled. These measurements are corrupted by Gaussian noise ($\sigma = 0.25$). The top left figure shows the initial state of the network after smoothing. The following illustrations show the changing states of the network, clearly revealing three rectangles. Note, that in order to reconstruct such objects, there is a critical number of sampling points *per* object, below which the reconstruction yields very ambiguous results. Time is specified in terms of τ. A variable coupling was assumed.

point (on the average) along the contours marked with arrows. Thus, only about 5% of all points in the image are sampled. The results speak for themselves.

How does the choice of λ affect our results? For $\lambda >> 1$, that is in the high-gain limit, the l_i's will be almost always either 0 or 1. Conversely, for $\lambda << 1$, values of l_i will be evenly distributed between 0 and 1. Experimentally, the only difference between a run with low or high gain is the convergence time. Runs with low λ usually took much longer to converge, since the values of l_i were distributed within the hypercube $[0, 1]^N$ and the first term in E_L explicitly penalizes these values. The final solution set appeared to be the same, no matter what value of λ was chosen. For all our simulations $\lambda = 16$. Under these conditions, $v_{i,j}$ and $h_{i,j}$ are almost always either 0 or 1.

Figures 3 and 4 show more complicated synthetic images. To reconstruct the original, fully sampled, image as closely as possible, we introduced the following procedure. Most images contain, unlike figure 1, more than a single depth scale. That is, the depth gradient may vary greatly throughout an image, reflecting the fact that objects are located at different depths. As we discussed before, the relative weight of the line interaction energy E_L to the interpolation term E_I governs the scale at which no more smooth interpolation takes place and the surface breaks. Therefore, one way to scan for different depth values is to change the weight of E_L during a simulation run. We multiplied E_L by a factor $1/K(t)$, where $K(t)$ starts out small — typically at 0.1 — and increases linearly until a given saturation threshold. In other words, initially the formation of lines is strongly penalized, encouraging a smooth interpolation everywhere except at very steep disparity gradients. Subsequently, by paying a smaller and smaller price for the formation of lines, the surface will break at smaller and smaller depth gradients. $K(t)$ is bounded from above, since noise will otherwise lead to the creation of lines everywhere. The final state of the network is independent of the speed at which $K(t)$ changes, as long as it increases slowly enough. Interestingly, increasing $K(t)$ gives rise to new lines while older lines persisted. In no case did previous established lines fade away, although we are unable to prove this assertion. Note that the evolution of this nonlinear dynamical system can best be described as a process involving the minimization of an energy functional at each stage. This energy functional varies from stage to stage (see also Terzopoulos, 1985).

5. Discussion

The results we have presented indicate the plausibility of using graded networks of simple neuron-like processing elements to "solve" constraint satisfaction problems in early vision that can be formulated as minimization of a convex or a non-convex energy function. We have simulated on a digital computer a network for one particular well characterized problem, reconstructing surfaces given noisy and sparse depth measurements. Although this particular task has no "optimal" solution, since reconstructing a complex surface from insufficient data allows a potentially infinite number of solutions, our simulations indicate that the reconstructed image appears to be at least "very good". In particular, the solutions are similar to the solutions obtained using simulated annealing or other algorithms derived from estimation theory (Marroquin, 1984, 1985b).

Many early vision problems can be formulated in terms of minimizing an energy function or as finding optimal Bayesian estimates. Although these methods are by no means the only approaches to early vision, they do offer advantages, including elegant ways of

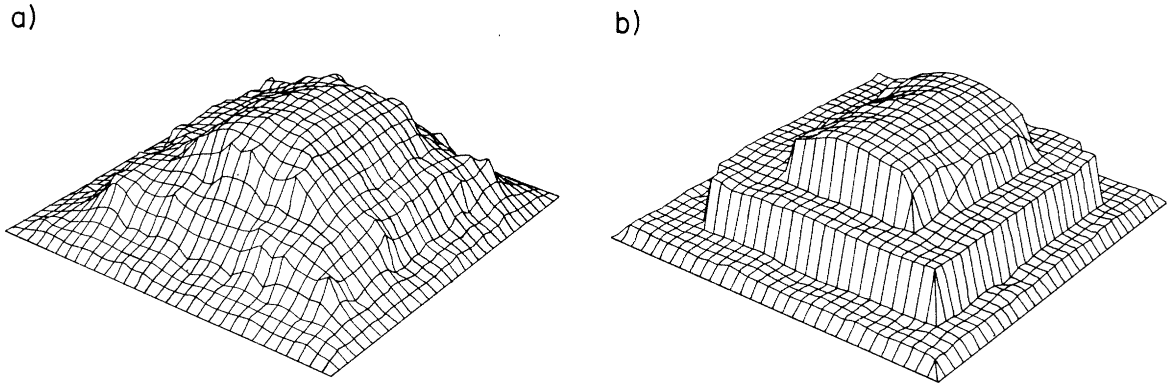

Figure 4. Smooth and piecewise smooth surface reconstruction of a sparsely sampled synthetic scene containing both flat and curved surfaces. Smoother surface interpolation can be obtained by the use of a higher-order stabilizer, such as the thin plate stabilizer (Terzopoulos, 1985). The network is shown after 1.6τ (for more details see figure 3).

representing constraints, or *a priori* knowledge of the world. Vision problems which have been formulated in this way include surface interpolation (Grimson, 1981, 1982; Terzopoulos, 1985; Marroquin, 1984), edge detection (Poggio, Vorhees, Yuille 1984), shape from shading (Horn and Brooks 1984), velocity field estimation (Horn and Schunck, 1981; Hildreth, 1984) and color (Hurlbert, 1985; Poggio *et al.*, 1985).

The single most important advantage of analog, parallel networks, independent of their implementation, is their speed. Typical convergence times are on the order of several system time constants. Thus, the convergence times will be of the order of 10 to $100ms$ for neuronal hardware and of the order of 10 to $100ns$ for semiconductor circuits. An image similar to figure 3 but on a 128 by 128 grid yields similar convergence times. The convergence time does not depend *per se* on the size of the image array but rather on the size of the largest patch of smooth surface in the image. Since the quadratic smoothing term mimicks a diffusion, it takes on the order of n^2 time constants for information to propagate across the smooth patch, n pixels across. This behavior contrasts favorably with simulated annealing (Kirkpatrick *et al.*, 1983; Hinton and Sejnowski, 1983). This latter technique, although guaranteed to converge asymptotically to the global minimum, is often very slow.

It is interesting that our attempts to enhance the final solution by changing λ during a single run did not have any apparent effect, except to increase the convergence time. In the Ising spin interpretation of Hopfield's binary networks (Huang, 1963; Hopfield and Tank, 1985) λ can be interpreted as temperature T of the system and changing λ during a run

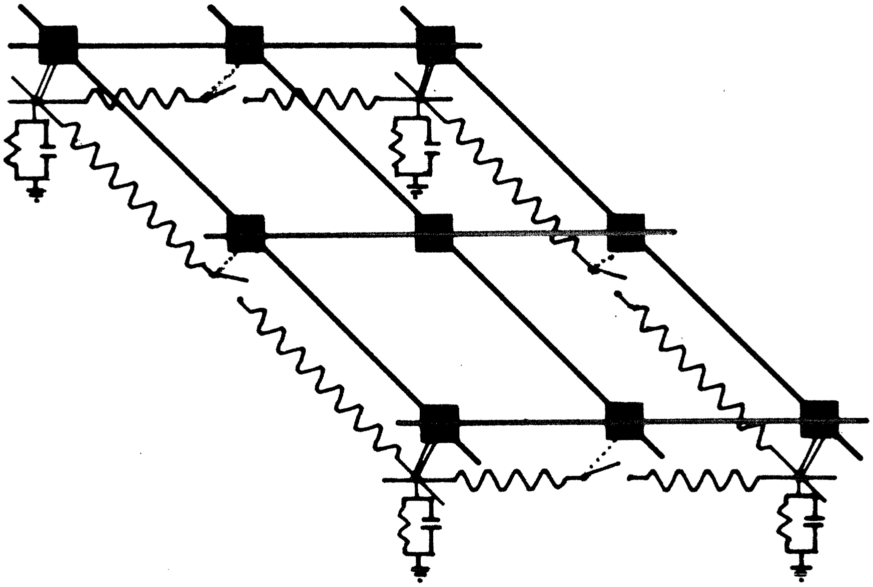

Figure 5. Schematic diagram of a *hybrid* massive parallel machine for a real-time artificial vision system. The analog network — built using constant Ohmic resistances — minimizes a piecewise quadratic energy expression while the digital processors (indicated as solid squares) compute an arbitrary, nonlinear energy expression. The digital processors corresponding to the line process set or break the resistive connections in the analog network with the help of one or two transistors. The input data for the resistive network, positive currents, and the output of the network — positive voltages — are read in/out through the digital processors after suitable A/D or D/A conversion. Note that the spatial resolution of the analog network is half of the resolution of the digital network. This scheme combines the speed of a simple analog network with the versatility of a digital processor and would permit real time execution of vision algorithms.

would be paramount to simulated annealing.

The principal drawback of our method is that there is no guaranty that the network will converge to a state of lowest energy. Similar to the network solutions to the Travelling Salesman Problem (Hopfield and Tank, 1985), we can only show experimentally that the computed solutions seem reasonable compared with solutions obtained with other algorithms. Thus, it appears that our network seems to find a state of low — if not the lowest — energy. As Hopfield and Tank (1985) have pointed out, the main reason for this good behavior is the smoothing of the solution space upon transforming the problem from a discrete, binary space into a continuous one.

The energy expression for the surface interpolation problem (equation 11a) contains, unlike Hopfield's corresponding energy function (equation 9), cubic terms; that is, the corresponding update equation (12) contains a multiplication of two variables. A standard way of implementing a multiplication within analog microelectronic circuits is to change the conductance of a transistor in a voltage dependent manner (variable transconductance multiplier; see for instance Grebene, 1972). As first proposed by Torre and Poggio (1978; see also Koch, Poggio and Torre, 1982) there is experimental evidence that a similar mechanism may be implemented within the dendritic trees of nerve cells. An inhibitory synapse with a reversal potential close or equal to the resting potential of the neuron (a so-called *shunting* or *silent* inhibition) interacts with an excitatory synapse in a multiplicative manner.

What are the implications of our results for the architecture of artificial vision systems? How feasible is the construction of truly analog integrated network circuits? The network we have proposed for solving the surface reconstruction problem is a simple one, in as far as the weights between the different processing elements are positive, global variables and do not depend on the location in the network. This is, unfortunately, not always to be expected. For instance, the resistances in the linear network Poggio and Koch (1985) have proposed for computing the smoothest velocity field (Hildreth, 1984) do depend on the location, making an integrated circuit implementation difficult. Both linear and nonlinear transfer functions can be built — within limits — using standard circuit technologies. The technology of choice would most likely be a variant of the MOS technology. A variable weighting term, such as $K(t)$, would be difficult to implement within any degree of accuracy.

Purely analog networks do have one major drawback with regard to conventional programmable processors. Once a particular analog network has been build, it is difficult to change its parameters — such as the specific form of the penalizing terms — or to adapt the network to a different use. Thus, every task would require a dedicated analog network. A second problem is handling the massive dataflow from and to the individual processors. A possible solution to this dilemma is a mixed, *hybrid* architecture, where full advantage is taken of the speed of analog circuit components and the versatility of programmable processors.

The observation that a great number of early vision problems can be formulated as either minimizing a quadrational energy or as minimizing an energy containing a quadratic term (Poggio *et al.*, 1985) leads to the following scheme for such a hybrid machine. Given a regular grid with a number of simple serial and programmable processors at every node, such as the Connection Machine currently under development at MIT and TMC (Hillis, 1984). Every processor is connected to its four neighbours (figure 5). Superimposed onto this Manhattan-like geometry is a regular network with constant Ohmic resistances, with half the spatial resolution of the digital network. The processors corresponding to the line elements have the ability to break the resistive connection between two neighbouring processors with the aid of a simple switch. The processors "above" the nodes of the linear network have at any time the option to read in/out from the analog network.

The hybrid machine has two basic cycles. In its analog cycle, the processors inject (positive) current — corresponding to the depth measurements — into the analog network. Subsequently, the resistive network finds the smoothest surface, *given a certain distribution of lines* — mimicked by the breaking of resistances between nodes. It is guaranteed to find the smoothest surface — for a given line distribution — since the network will converge to a state of least power dissipation, a quadratic functional (Poggio and Koch, 1985). Convergence is expected to be fast in comparison with the typical execution times of simple operations in the processors. In the digital cycle, the processor network will read out the current voltage at every node in the resitive network — corresponding to the reconstructed depth value — and compute a new estimate for the now *binary* line process using conventionally programmed digital software. Subsequently, the processors will set or break the appropriate connections in the analog network. This hybrid system thus switches between the analog and the digital mode, essentially implementing our solution strategy. Changing the weight or the form of the energy expression can be done with the help of the digital processors, offering greater versatility than a purely analog system. Since the digital processors are "where the data is", such a scheme alleviates or strongly reduces the data flow problem. Note that the conversion of the data from digital to analog and vice versa implies delays between cycles, probably on the order of several microseconds if the accuracy of the depth variable is no more than 6 to 8 bits.

Acknowledgments: Foremost we wish to thank John Hopfield, who, during his sabattical at the Center for Biological Information Processing, motivated us to study the use of neuronal networks in early vision. Eric Saund contributed greatly to the early stages of the project. Tomaso Poggio provided the intellectual environment which made everything possible and helped formulate some of the basic questions. Tom Knight gave us insights into the technology of building analog circuits. Linda Ardrey drew the figures. We thank Tom Knight, Tomaso Poggio and Demitri Terzopoulos for critically reading the manuscript.

6. References

Blake, A., "The least-disturbance principle and weak constraints", *Patt. Recogn. Lett.*, **1**, 393–399, 1983.

Brady, J.M., "Computational approaches to image understanding", *Computing Surveys*, **14**, 3–71, 1982.

Ballard, D.H., Hinton, G.E., and Sejnowski, T.J. "Parallel visual computation", *Nature*, **306**, 21–26, 1983.

Geman, S. and Geman, D., "Stochastic Relaxation, Gibbs Distributions and the Bayesian Restoration of Images", *IEEE Trans. Pattern Analysis & Machine Intelligence*, **6**, 721–741, 1984.

Grimson, W.E.L., *From Images to Surfaces: A computational study of the human early visual system*, MIT Press, Cambridge, Ma., 1981.

Grimson, W.E.L., "A computational theory of visual surface interpolation", *Phil. Trans. R. Soc. Lond., B*, **298**, 395–427, 1982.

Grebene, A.B., *Analog integrated circuit design*, Robert Krieger Publishing Company, Florida, 1972.

Hildreth, E.C., *The Measurement of Visual Motion*, MIT Press, Cambridge, 1984.

Hillis, W.D., "The connection machine: a computer architecture based on cellular automata", Physica, **10D**: 213–228, 1984.

Hinton, G.E., and Sejnowski, T.J., "Optimal perceptual inference", *Proc. IEEE 1983 Conf. Computer Vision and Pattern Recognition*, Washington, DC, 1983.

Hopfield, J.J., "Neural networks and physical systems with emergent collective computational abilities", *Proc. Natl. Acad. Sci. USA*, **79**: 2554–2558, 1982.

Hopfield, J.J., "Neurons with graded response have collective computational properties like those of two-state neurons", *Proc. Natl. Acad. Sci. USA*, **81**: 3088–3092, 1984.

Hopfield, J.J. and Tank, D.W., "Neural computation in optimization problems", *Biol. Cybern.*, in press, 1985.

Horn, B.K.P. and Brooks, M. "The variational approach to shape from shading",*Artificial Intelligence Lab. Memo*, **No. 815**, MIT, Cambridge, 1985.

Horn, B.K.P. and Schunck, B.G., "Determining optical flow", *Artificial Intelligence*, **17**, 185–203, 1981.

Huang, K., *Statistical mechanics*, John Wiley & Sons, New York, 1963.

Hurlbert, A., "Color computation in the visual system", *Artificial Intelligence Lab. Memo*, **No. 814**, MIT, Cambridge, 1985.

Kindermann, R. and Snell, J.L., "Markov Random Fields and their Application", American Mathematical Society, Vol. 1, 1980.

Kirkpatrick, S., Gelatt, C.D., Jr., and Vecchi, M.P., "Optimization by simulated annealing", *Science*, **220**, 671-680, 1983.

Koch, C., Poggio, T. and Torre, V., "Retinal ganglion cells: a functional interpretation of dendritic morphology", *Phil. Trans. R. Soc. Lond. B*, **298**, 227-264, 1982.

Kohonen, T. *Associative Memory*, Springer Verlag, Heidelberg, 1977.

Marr, D., *Vision*. Freeman, San Francisco, 1982.

Marr, D. and Poggio, T., "Cooperative computation of stereo disparity", *Science*, **195**, 283-287, 1977.

Marr, D. and Poggio, T., "A theory of human stereo vision", *Proc. R. Soc. Lond. B*, **204**, 301-328, 1979.

Marroquin, J. "Surface reconstruction preserving discontinuities", *Artificial Intelligence Lab. Memo*, **No. 792**, MIT, Cambridge, 1984.

Marroquin, J. "Probabilistic solution of inverse problems", *Ph.D. Thesis*, Department of Electrical Engineering and Computer Science, MIT, Cambridge, 1985a.

Marroquin, J. "Optimal bayesian estimators for image segmentation and surface reconstruction", *Artificial Intelligence Lab. Memo*, **No. 839**, MIT, Cambridge, 1985b.

Palm, G., *Neural assemblies*, Springer Verlag, Heidelberg, 1982.

Poggio, T. and Koch, C., "An analog model of computation for the ill-posed problems of early vision", *Artificial Intelligence Lab. Memo*, **No. 783**, MIT, Cambridge, 1984.

Poggio, T., and Koch, C., "An analog model of computation for the ill-posed problems of early vision", *Proc. R. Soc. Lond. B*, in press, 1985.

Poggio, T. and Torre, V., "Ill-posed problems and regularization analysis in early vision", *Artificial Intelligence Lab. Memo*, **No. 773**, MIT, Cambridge, 1984.

Poggio, T., Torre, V. and Koch, C., "Computational vision and regularization theory", *Nature*, in press, 1985.

Poggio, T., Voorhees, H. and Yuille A., "Regularizing Edge Detection", *Artificial Intelligence Lab. Memo*, **No. 776**, MIT, Cambridge, 1984.

Terzopoulos, D., "Multilevel computational processes for visual surface reconstruction", *Comp. Vision. Graph. & Image Proc.*, **24**, 52–96, 1983.

Terzopoulos, D., "Multiresolution computation of visible-surface representations", *Ph.D. Thesis*, Department of Electrical Engineering and Computer Science, MIT, Cambridge, 1984.

Terzopoulos, D., "Regularization of ill-posed visual reconstruction problems involving discontinuities", *IEEE Trans. Pattern Analysis & Machine Intelligence*, in press, 1985.

Torre, V. and Poggio, T., "A synaptic mechanism possibly underlying directional selectivity to motion", *Proc. R. Soc. Lond. B*, **202**, 409–416, 1978.

Ullman, S., "Relaxation and constrained optimization by local processes", *Comp. Graphics Image Proc.*, **10**, 115–125, 1979.