INTERNATIONAL ATOMIC ENERGY AGENCY
UNITED NATIONS EDUCATIONAL, SCIENTIFIC AND CULTURAL ORGANIZATION

INTERNATIONAL CENTRE FOR THEORETICAL PHYSICS
34100 TRIESTE (ITALY) - P.O.B. 586 - MIRAMARE - STRADA COSTIERA 11 - TELEPHONE: 9240-1
CABLE: CENTRATOM - TELEX 460392-1

SMR/302-29

COLLEGE ON NEUROPHYSICS:
"DEVELOPMENT AND ORGANIZATION OF THE BRAIN"
7 November - 2 December 1988

"Visual Reconstruction and the GNC Algorithm"

Andrew BLAKE
Department of Engineering Science
University of Oxford
Oxford, U.K.

Visual Reconstruction and the GNC Algorithm [1]

Andrew Blake and Andrew Zisserman

1 Introduction

Piecewise continuous reconstruction is a generic problem in vision. Consider the following visual sub-tasks:

1. Estimating the shape of a textured surface, viewed stereoscopically
2. Detection of discontinuities in intensity
3. Estimating surface shape from a depth array obtained from an active rangefinder
4. Segmentation and description of curves, either in an image or in 3D space
5. Estimation of reflectance (or of lightness) from intensity data
6. Estimating the shape of a textured surface from its optical flow field
7. Segmentation of textures

Each involves reconstruction of some piecewise continuous function from noisy data. In the case of task 1, the data is a set of correspondences and the reconstructed data is a depth map - a function encoding distance x to the visible surface, for each image point. In case 2 the input is noisy intensity data and the output is a set of edges together with a smoothed intensity field. (Smoothing is inhibited across edges however.)

There has been a good deal of research interest in "cooperative" algorithms to perform reconstruction tasks. A cooperative algorithm is one that can be executed by a number of interconnected processing cells working in parallel.

[1] A revised extract from *Visual Reconstruction*, A.Blake and A.Zisserman, MIT Press.

Early work on stereoscopic matching was done by Julesz (1971) and by Marr (1976). Subsequent algorithms however have viewed stereoscopic matching as a combinatorial problem, rather than as a problem of reconstruction (Marr and Poggio 1979, Mayhew and Frisby 1981, Baker 1981, Ohta and Kanade 1985) - although Pollard et al. (1985) retained some cooperativity.

Cooperative solutions to tasks 1,2,3,4 above are discussed by us in some detail in a book (Blake and Zisserman 1987). In each case, piecewise continuity of a reconstruction is achieved by imposing "weak continuity constraints". These are constraints that impose continuity almost everywhere, but can be broken when forced to do so by the data. The mechanism for actually imposing the constraints is the "GNC" (Graduated Non-convexity) algorithm. This will be described briefly here, but see (Blake and Zisserman 1987) for details.

Our approach to reconstruction originates from earlier work on edge detection (Blake 1983) and has been developed in (Blake and Zisserman 1985, 1986, Blake et al. 1986). Mumford and Shah (1985) clarified the relationship between piecewise continuous reconstruction and edge detection by linear filtering. Geman and Geman (1984) introduced a powerful statistical approach to reconstruction, based on "simulated annealing". Related algorithms have since been used for reconstruction of stereoscopically viewed surfaces (Marroquin 1984), a development of work by Grimson (1981) and Terzopoulos (1984), for segmentation of optic flow fields (Murray and Buxton 1987) and for segmentation of texture (Derin and Cole 1986).

For illustrative purposes, in this paper, we consider the simplest form of reconstruction problem: detection of step discontinuities (edges) in 1D data. The aim is to construct a piecewise smooth 1D function $u(x)$ which is a good fit to some data $d(x)$. This is achieved by modelling $u(x)$ as a "weak elastic string" - an elastic string under weak continuity constraints. Discontinuities are places where the continuity constraint on $u(x)$ is violated. They can be visualised as breaks in the string. The weak elastic string is specified by its associated energy; the problem of finding $u(x)$ is then the problem of minimising that energy.

2 Detecting step discontinuities in 1D

The behaviour of the elastic string over an interval $x \in [0, N]$ is defined by an energy, which is a sum of three components:

P: the sum of penalties α levied for each break (discontinuity) in the string.

D: a measure of faithfulness to data.

$$D = \int_0^N (u - d)^2 dx$$

S: a measure of how severely the function $u(x)$ is deformed.

$$S = \lambda^2 \int_0^N u'^2 dx$$

This is the elastic energy of the string itself that is stored when the string is stretched. The constant λ^2 is a measure of elasticity or "stretchability" or willingness to deform[2].

The problem is to minimise the total energy:

$$E = D + S + P \tag{1}$$

- that is, for a given $d(x)$, to find that function $u(x)$ for which the total energy E is smallest. Without the term P (if the energy were simply $E = D + S$) this problem could be simply solved using the calculus of variations. For example fig 1c shows the function u that minimises $D + S$ given the data $d(x)$ in fig 1a. It is clearly a compromise between minimising D and minimising S - a trade-off between sticking close to the data and avoiding very steep gradients. The precise balance of these 2 elements is controlled by λ. If λ is small, D (faithfulness to data) dominates. The resulting $u(x)$ is a close fit to the data $d(x)$. In fact, λ has the dimensions of length, and it will be shown that λ is a characteristic length or scale for the fitting process.

When the P term is included in E, the minimisation problem becomes more interesting. No longer is the minimisation of E straightforward mathematically. E may have many local minima. For example for the problem of fig 1, b) and c) are both local minima. Only one is a global minimum; which one that is depends on the values of α, λ and the height of the step in a). If the global minimum is b) then the reconstruction $u(x)$ contains a discontinuity; otherwise, if it is c), $u(x)$ is continuous. (See Blake and Zisserman 1987) for variational analysis that precisely characterises the effect of parameters α, λ.)

3 The computational problem

The "finite element method" (Strang and Fix 1973) is a good means of converting continuous problems, like the one just described, into discrete problems. In the case of the string it is relatively easy. The continuous interval $[0, N]$ is divided into N unit sub-intervals ("elements") $[0, 1], \ldots [N-1, N]$, and nodal values are defined: $u_i = u(i), i = 0 \ldots N$. Then $u(x)$ is represented by a linear piece in each sub-interval (fig 2). The energies defined earlier now become:

$$D = \sum_0^N (u_i - d_i)^2 \tag{2}$$

[2]Really, interpreting S as a stretching energy is only valid when the string is approximately aligned with the x axis. Another way to think of S is that it tends to keep the function $u(x)$ as flat as possible.

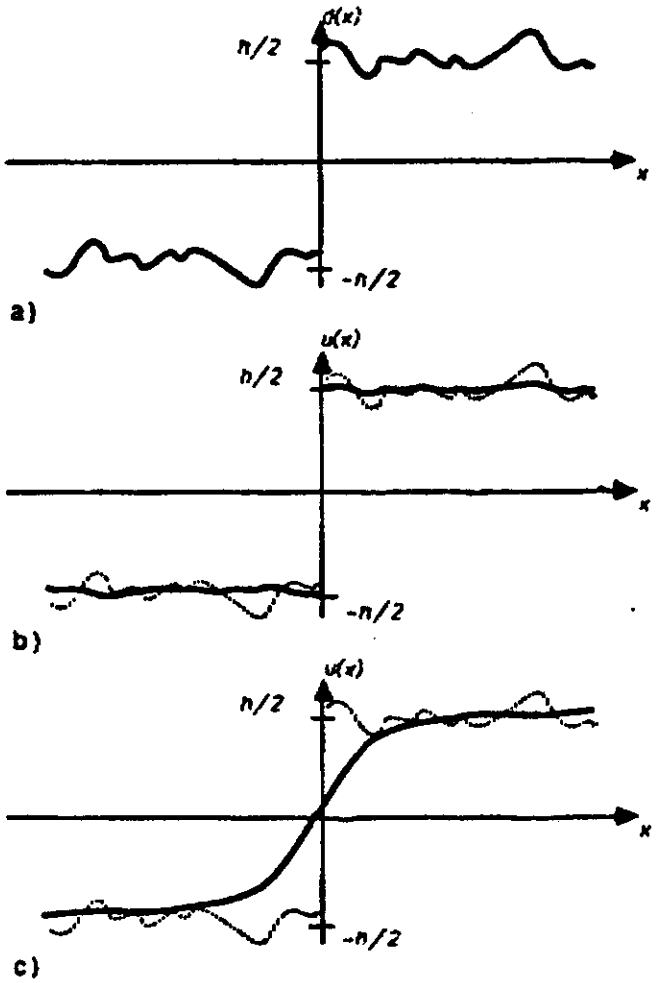

Figure 1: Calculating energy for data consisting of a single step. (a) Data. (b) A reconstruction with one discontinuity. (c) A continuous reconstruction.

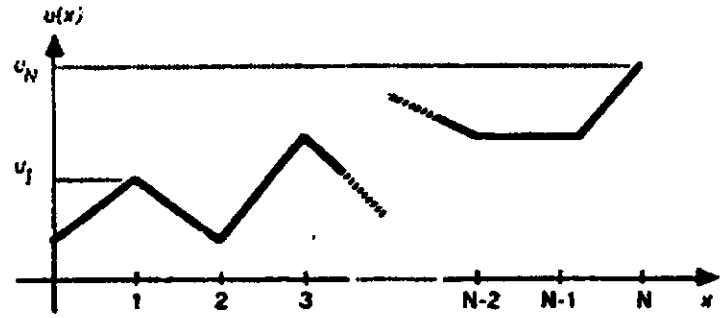

Figure 2: Dividing a line into sub-intervals or "elements".

$$S = \lambda^2 \sum_1^N (u_i - u_{i-1})^2 (1 - l_i) \tag{3}$$

$$P = \alpha \sum_1^N l_i \tag{4}$$

where l_i is a so-called "line-process". It is defined such that each l_i is a boolean-valued variable.

Either: $l_i = 1$ indicating that there is a discontinuity in the sub-interval $x \in [i-1, i]$.

or: $l_i = 0$ indicating continuity in that subinterval - u_i, u_{i-1} are joined by a spring.

Note that when $l_i = 1$ the elastic string is "broken" between nodes $i-1$ and i and the relevant energy term in (3) is disabled. (Geman and Geman (1984) coined the term "line-process" as a set of discrete variables describing edges in 2D; here we have a simple case, appropriate in 1D.)

4 Eliminating the line process

The problem, now in discrete form, is simply:

$$\min_{\{u_i, l_i\}} E. .$$

It transpires that the minimisation over the $\{l_i\}$ can be done "in advance". The problem reduces simply to a minimisation over the $\{u_i\}$. Exactly how this is achieved is explained in (Blake and Zisserman 1987). The reduced problem is more convenient for two reasons:

- The computation is simpler as it involves just one set of real variables $\{u_i\}$, without the boolean variables $\{l_i\}$.
- The absence of boolean variables enables the "graduated non-convexity algorithm", described later, to be applied.

It will be shown that once the line-process $\{l_i\}$ has been eliminated, the problem becomes

$$\min_{\{u_i\}} F, \text{ where } F = D + \sum_{1}^{N} g(u_i - u_{i-1}). \tag{5}$$

The neighbour interaction function g will not be defined precisely here but to give some idea of how it acts, it is plotted in figure 3. The term $S+P$ in (1) has

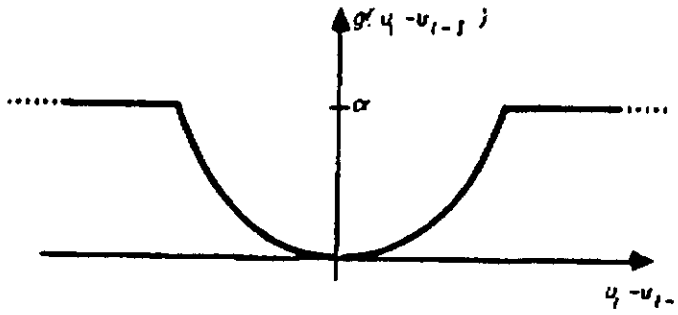

Figure 3: Energy of interaction between neighbours in the weak string. The central dip encourages continuity by pulling the difference $u_i - u_{i-1}$ between neighbouring values towards zero. The plateaus allow discontinuity: the pull towards zero difference is released, and the weak continuity constraint has been broken.

been replaced by the $\sum g(..)$ term in (5). Note that nothing of value has been thrown away by eliminating line variables. They can very simply be explicitly recovered from the optimal $\{u_i\}$ (Blake and Zisserman 1987).

5 Convexity

The discrete problem has been set up. The task now is to minimise the function F; but that proves difficult, for quite fundamental reasons. Function F lacks the mathematical property of "convexity". What this means is that the system u_i may have numerous stable states, each corresponding to a local minimum of energy F. Such a state is stable to small perturbations - give it a small push and it springs back - but a large perturbation may cause it to flip suddenly into a state with lower energy.

There may be very many local minima in a given F. In fact there may be one local minimum of F corresponding to each state of the line process l_i - 2^N local minima in all! The goal of the weak string computation is to find the *global* minimum of F; this is the local minimum with the lowest energy. Clearly it is infeasible to look at all the local minima and compare their energies.

How do these local minima arise? The function F can be regarded as the energy of a system of springs, as illustrated in figure 4a. Vertical springs are attached at one end to anchor points, representing data d_i which are fixed, and to nodes u_i at the other end. These springs represent the D term in the energy F (5). There are also lateral springs between nodes. Now if these springs were just ordinary springs there would be no convexity problem. There would be just one stable state: no matter how much the system were perturbed, it would always spring back to the configuration in figure 4a. But in fact the springs are special; they are the ones that enforce weak continuity constraints. Each is initially elastic but, when stretched too far, gives way and breaks, as specified by the energy g in figure 3. As a consequence, a second stable state is possible (figure 4b) in which the central spring is broken. In an intermediate state (figure 4c) the energy will be higher than in either of the stable states, so that traversing from one stable state to the other, the energy must change as in figure 4d. For simplicity, only 2 stable states have been illustrated, but in general each lateral spring may either be broken or not, generating the plethora of stable states mentioned above.

No local descent algorithm will suffice to find the minimum of F. Local descent tends to stick, like the fly shown in figure 4, in a local minimum, and there could be as many as 2^N local minima to get stuck in. Somehow some global "lookahead" must be incorporated. The next section explains how the Graduated Non-Convexity (GNC) algorithm does this.

6 Graduated non-convexity

A method of minimising F is needed which avoids the pitfall of sticking in local minima. Stochastic methods such as "Simulated Annealing" (Kirkpatrick et al. 1982) avoid local minima by random fluctuations, spasmodic injections of energy, to shake free of them (figure 5a). Although this guarantees to find the global minimum eventually, the amount of computation required may be very large (Geman and Geman 1984). It would appear, however, to be in the interests of computational efficiency to use a non-random method. GNC, rather than injecting energy randomly, uses a modified cost function (fig 5b).

In the GNC method, the cost function F is first approximated by a new function F^* which is *convex* and hence can only have one local minimum, which must also be a global minimum[3]. Descent on F^* (descending, that is, in the $(N+1)$-dimensional space of variables $\{u_i\}$) must land up at that minimum. Now, for certain data d_i this minimum may also be a *global* minimum of F - which is what we were after. There is a simple test to detect whether or not this

[3] Actually there are some details to take care of here, distinguishing between convexity and strict convexity.

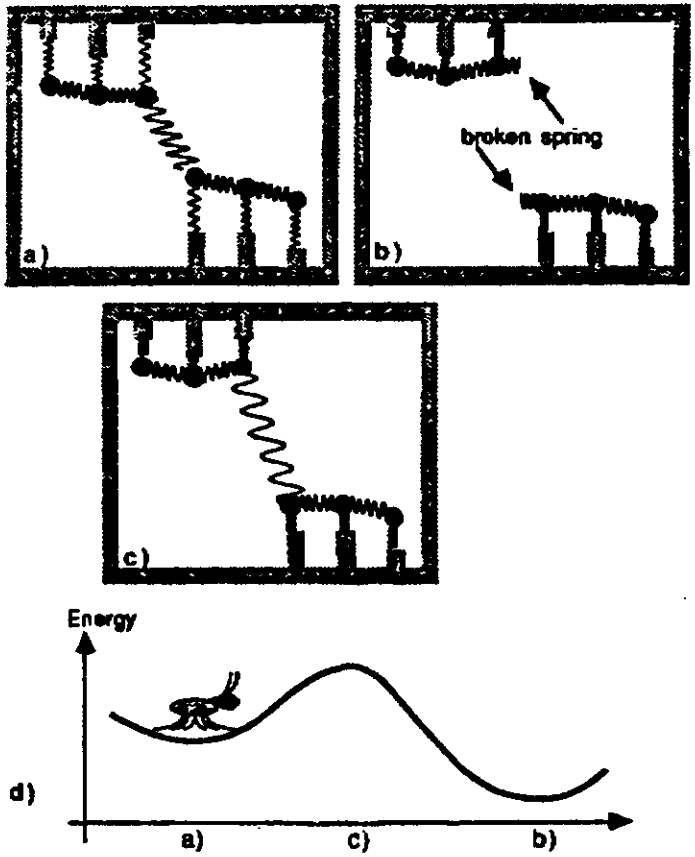

Figure 4: Non-convexity: the weak string is like a system of conventional vertical springs with "breakable" lateral springs as shown. The states (a) and (b) are both stable, but the intermediate state (c) has higher energy than either (a) or (b). Suppose the lowest energy state is (b). A myopic fly with vertigo, crawling along the energy transition diagram (d) thinks state (a) is best - he has no way of seeing that, over the hump, he could get to a lower state (b).

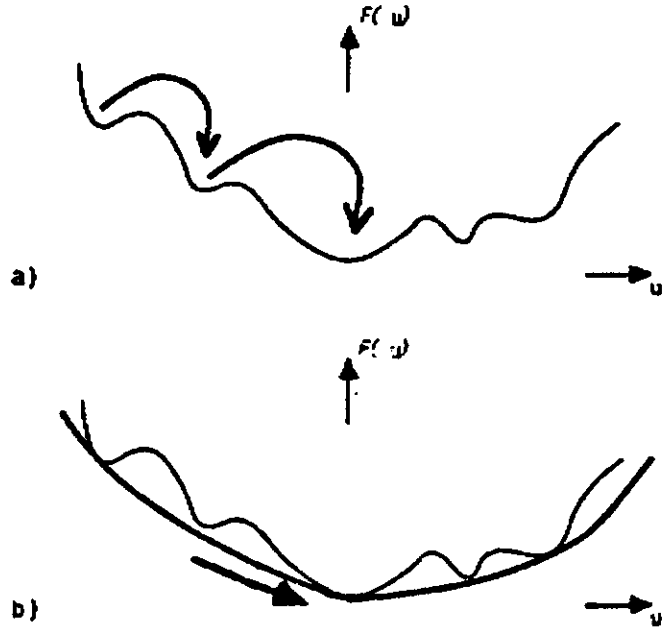

Figure 5: a) Stochastic methods avoid local minima by using random motions to jump out of them. b) The GNC method constructs an approximating convex function, free of spurious local minima.

has succeeded (fig 6a,b). In fact (Blake and Zisserman 1987) success is most likely when the scale parameter λ is small.

A more general strategy, that works for small or large λ, is to use a whole *sequence* of cost functions $F^{(p)}$, for $1 \geq p \geq 0$. These are chosen so that $F^{(0)} = F$, the true cost function, and $F^{(1)} = F^*$, the convex approximation to F. In between, $F^{(p)}$ changes, in a continuous fashion, between $F^{(1)}$ and $F^{(0)}$. The GNC algorithm is then to optimise a whole sequence of $F^{(p)}$, for example $\{F^{(1)}, F^{(1/2)}, F^{(1/4)}, F^{(1/8)}, F^{(1/16)}\}$, one after the other, using the result of one optimisation as the starting point for the next. As an example, optimisation of a non-convex F, using a sequence of just 3 functions, is illustrated in fig 6c. Initially, optimisation of $F^{(1)} \equiv F^*$ produces u^* but (let us suppose) this happened not to be the global optimum of F. (Note that any starting point will do for optimising $F^{(1)}$. That is because $F^{(1)}$, being convex, has only one minimum, which will be attained by descent, regardless of where descent starts.) But successive optimisation of $F^{(p)}$ as p decreases "pulls" towards the true global optimum of F. Exactly how the functions $F^{(p)}$ are constructed is beyond the scope of this paper. It all depends on making F^* a good convex approximation to F. Suffice it to say that, like F in (5), F^* and all the $F^{(p)}$ are sums of local functions:

$$F^{(p)} = D + \sum_{1}^{N} g^{(p)}(u_i - u_{i-1}), \qquad (6)$$

and this is important when it comes to considering optimisation algorithms. Of course, the trick is to choose the right neighbour interaction function $g^{(p)}$.

There are numerous ways to minimise each $F^{(p)}$, including direct descent and gradient methods. Direct descent is particularly straightforward to implement and runs like this: propose a change in one of the nodal values u_i, see if that leads to a reduction in $F^{(p)}$ (this only involves a local computation); if it does then make the change. A simple program which implements GNC by direct descent is outlined in fig 7. It can be made to run quite satisfactorily with fixed point arithmetic. As $F^{(p)}$ is expressed as a sum over $i = 0, \ldots, N$ of local terms (6), the effect of altering a particular u_i (tested in the if and else if statements) can be computed from just a few of those terms. For example u_i appears only in $g^{(p)}(u_i - u_{i-1})$, $g^{(p)}(u_{i+1} - u_i)$ and, in D, in the term $(u_i - d_i)^2$. Not only does this simplify the computation of the effect on $F^{(p)}$ of changing u_i, but it is also possible to perform such computations on many u_i in parallel. More efficient algorithms, based on gradient descent methods, are described in (Blake and Zisserman 1987).

Figure 8 shows the GNC method in operation, solving the 1D weak elastic string problem. A successive gradient descent scheme (non-linear SOR) is used. For reasonably small values of the scale parameter λ the total time for execution[4] is about $0.001N$ seconds, where N is the length of the data vector. This works out at about 50 arithmetic operations per data element.

[4]On a SUN3 computer, with SKY floating point board

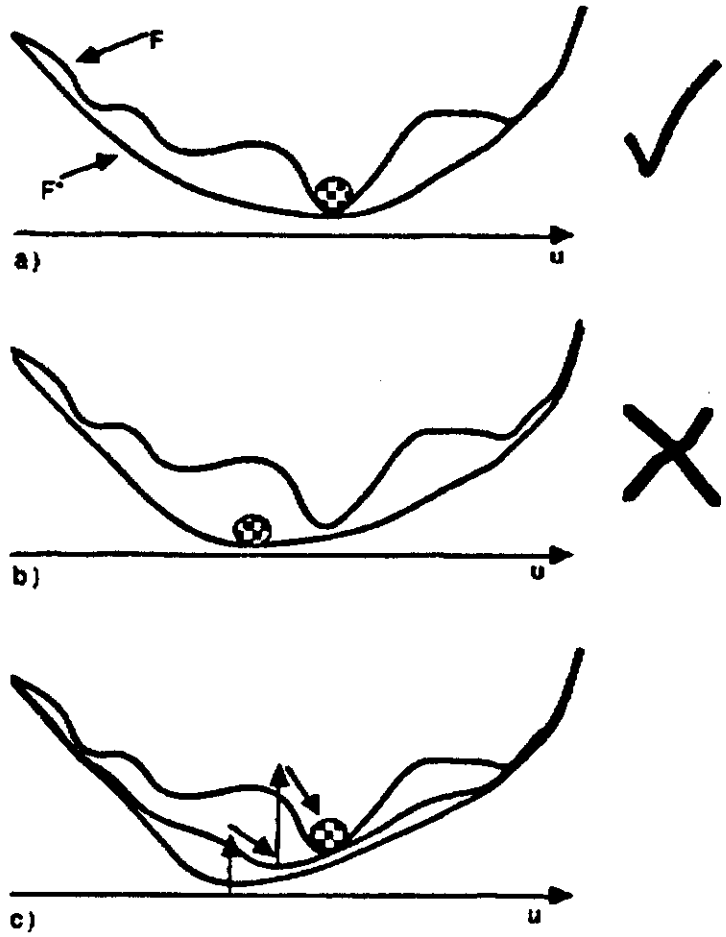

Figure 6: The minimum of a non-convex function F may be found by minimising a convex approximation F^* (a). If that does not work (b), the minimum may still be found by the GNC algorithm, which runs downhill on each of a sequence of functions (c), to reach the true global optimum.

```
for p ∈ {1, 1/2, 1/4, 1/8, 1/16, 1/32} do
 for δ := 1; δ ≥ δ_min; δ := δ/2 do
  changed := true
  while changed do
   changed := false
   for i = 0...N do
    if F^(p)(u_1, .., u_i + δ, .., u_N) < F^(p)(u_1, .., u_i, .., u_N) then
     u_i := u_i + δ
     changed := true
    else if F^(p)(u_1, .., u_i − δ, .., u_N) < F^(p)(u_1, .., u_i, .., u_N) then
     u_i := u_i − δ
     changed := true
```

Figure 7: A direct descent algorithm for GNC - see text for details

7 Analogue hardware

Horn (1974), Terzopoulos (1984) and others have pointed out that linear cooperative systems can be implemented as analogue networks. Simple, continuous reconstruction for example can be implemented by a resistive network, as in figure 9. In the case of reconstruction with discontinuities by the GNC algorithm, analogue implementation may also be possible, using a mixture of resistors and suitably chosen non-linear elements. The complete system would consist of a series of buffered stages, on e stage corresponding to each value of p used in the GNC algorithm. Provided the spatial scale (λ) of reconstruction is sufficiently small, just one or two stages are needed. For larger values of λ the number of stages increases as $\log(\lambda)$.

An outline circuit diagram is shown in figure 10. The non-linear element used in the network (figure 10a) operates in three modes

- a resistive mode
- an open-circuit mode
- a transitional mode

switching between them at built-in current trigger levels. Work is currently in progress to build a prototype system of this sort. Such a system has the attraction that in can perform the reconstruction computation very rapidly - perhaps as much as two orders of magnitude faster than a fully parallel, digital system. It could also, in principle, be suitable for implementation as an integrated circuit.

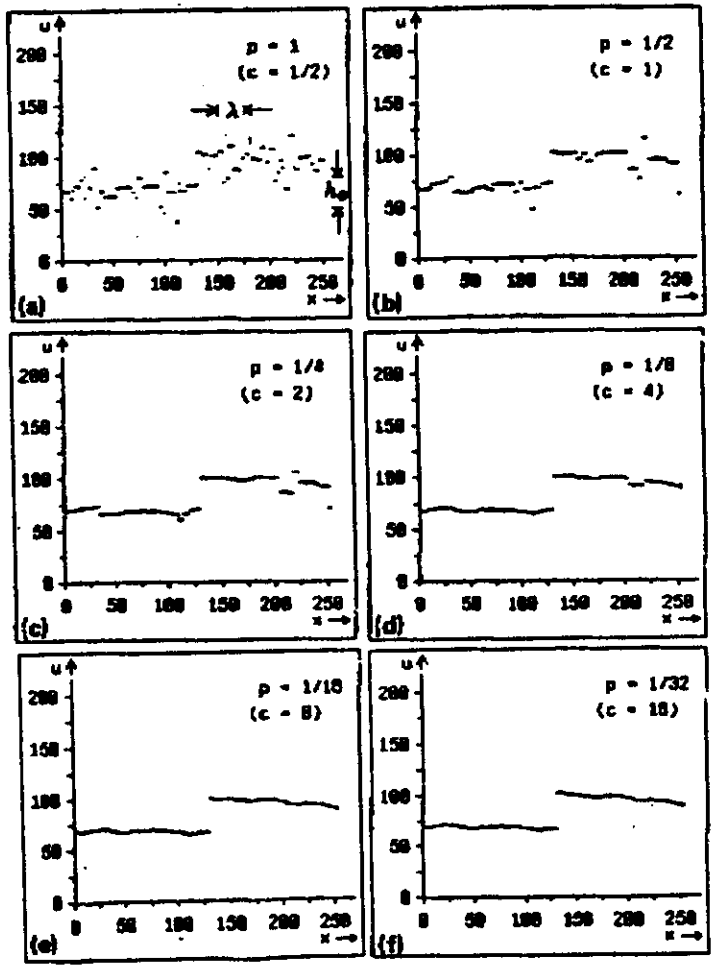

Figure 8: Snapshots of GNC: Initial data (a). As GNC progresses, parameter p is decreased (b)-(f). Parameters λ, h_0 (marked on (b)) are measures of characteristic scale and sensitivity to discontinuities, respectively.

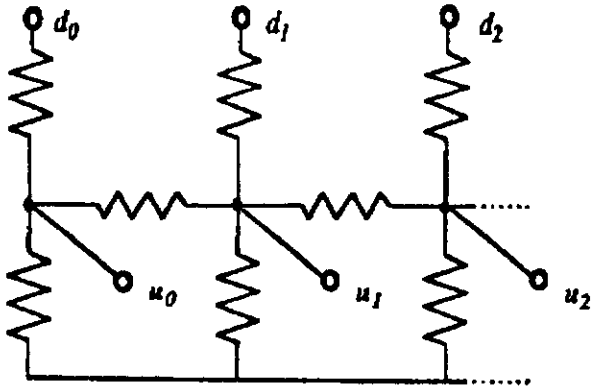

Figure 9: Resistive network for continuous reconstruction.

Acknowledgments

This work was supported by SERC grant GR/D 1439.6 and by the University of Edinburgh. The Royal Society of London's IBM Research Fellowship supported A. Blake. We are very grateful to Bernard Buxton and David Willshaw for helpful comments.

References

[1] Baker,H.H. (1981). Depth from edge and intensity based stereo. *IJCAI conf. 1981*, 583-588.

[2] Blake, A. (1983). The least disturbance principle and weak constraints. *Pattern Recognition Letters, 1, 393-399.*

[3] Blake, A. and Zisserman, A. (1985). Using weak continuity constraints. Report CSR-186-85, Dept. Computer Science, Edinburgh University, Edinburgh, Scotland. Also in *Pattern Recognition Letters*, 1987.

[4] Blake, A. and Zisserman, A. (1986). Weak continuity constraints in computer vision. Report CSR-197-86, Dept. Computer Science, Edinburgh University, Edinburgh, Scotland.

[5] Blake, A., Zisserman, A. and Papoulias, A.V. (1986). Weak continuity constraints generate uniform scale-space descriptions of plane curves. *Proc ECAI, Brighton, 1986.*

[6] Blake,A. and Zisserman,A. (1987). *Visual Reconstruction.* MIT Press, Cambridge, USA.

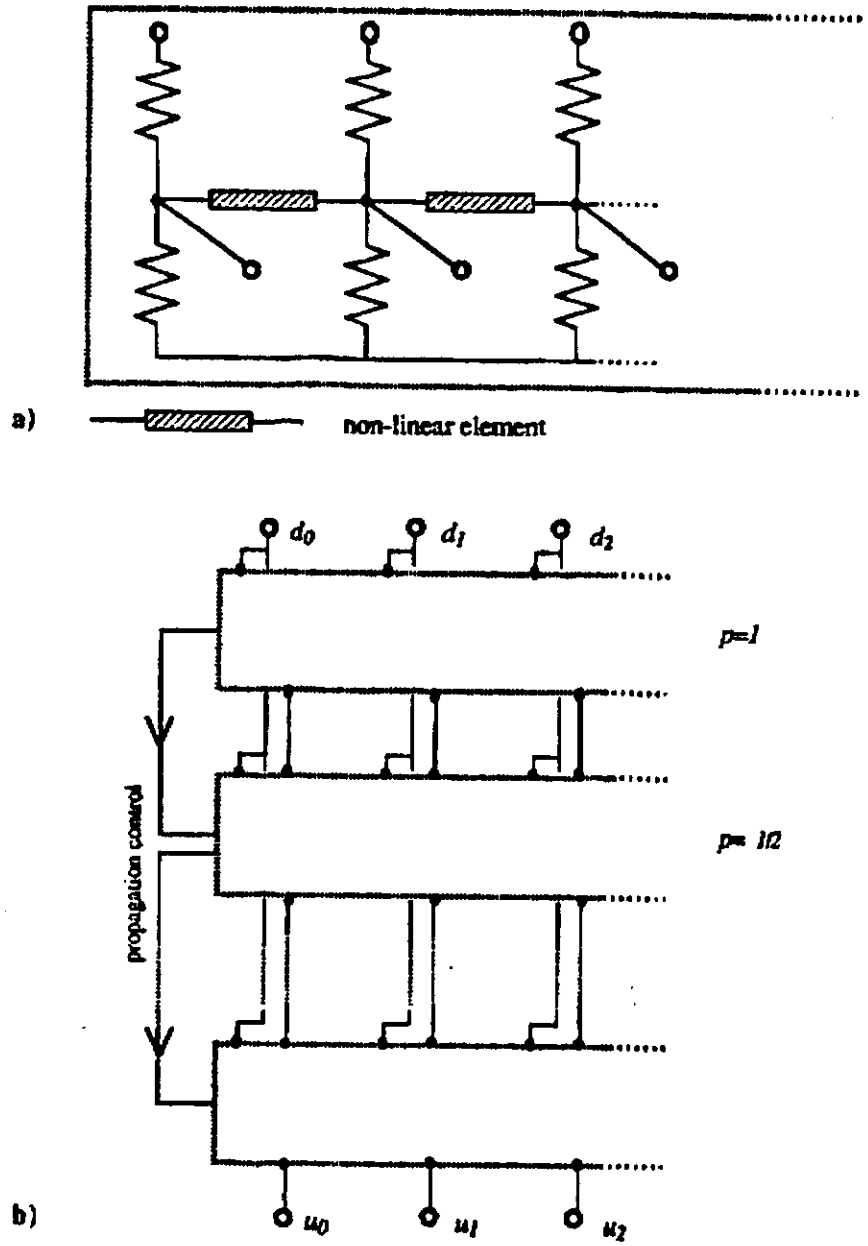

Figure 10: A non-linear network for reconstruction with discontinuities. (a) A single stage. (b) A cascade of stages forms the complete network.

[7] Derin,H. and Cole,W.S. (1986). Segmentation of textured images using Gibbs random fields. *CVGIP*, 35, 1, 72-98.

[8] Geman, S. and Geman, D. (1984). Stochastic Relaxation, Gibbs distribution, and Bayesian restoration of images. *IEEE PAMI*, 6, 721-741.

[9] Grimson,W.E.L. (1981). *From images to surfaces.* MIT Press, Cambridge, USA.

[10] Julesz,B (1971). *Foundations of cyclopean perception.* University of Chicago Press.

[11] Kirpatrick, S., Gellatt, C.D. and Vecchi, M.P. (1982). Optimisation by simulated annealing. IBM Thomas J. Watson Research Centre, Yorktown Heights, NY, USA.

[12] Marr,D. (1976). Cooperative computation of stereo disparity. *Science*, 194, 283-287.

[13] Marr,D. and Poggio,T. (1979). A computational theory of human stereo vision. *Proc. R. Soc Lond. B*, 204, 301-328.

[14] Marroquin, J. (1984). Surface reconstruction preserving discontinuities. Memo 792, AI Laboratory, MIT, Cambridge, USA.

[15] Mayhew,J.E.W and Frisby,J.P. (1981). Towards a computational and psychophysical theory of stereopsis. *AI Journal*, 17, 349-385.

[16] Mumford, D. and Shah, J. (1985). Boundary detection by minimising functionals. *Proc. IEEE CVPR conf.*, 22.

[17] Murray, D.W. and Buxton, B. (1986). Scene segmentation from visual motion using global optimisation. *IEEE PAMI*, 9, 2, 220-228.

[18] Ohta,Y. and Kanade,T. (1985). Stereo by intra- and inter-scanline search, using dynamic programming. *Proc. IEEE PAMI*, 7, 2, 139-154.

[19] Pollard,S.P., Mayhew,J.E.W. and Frisby,J.P. (1985). PMF: a stereo correspondence algorithm using a disparity gradient limit. *Perception, 14, 449-470.*

[20] Strang,G. and Fix,G.J. (1973). *An analysis of the finite element method.* Prentice-Hall, Englewood Cliffs, USA.

[21] Terzopoulos,D. (1984). Multi-resolution computation of visible-surface representations. Ph. D. thesis, MIT, Cambridge, USA.