Active Contours in Medical Image Processing using a Networked SIMD Array Processor

G D Sullivan, A D Worrall, R W Hockney and K D Baker

Department of Computer Science,
University of Reading, UK RG6 2AX
e-mail: G.Sullivan@reading.ac.uk

An SIMD parallel implementation of Kass et al's "snake" algorithm is reported, and demonstrated in use for interpreting medical images.

The SIMD machine acts as a "compute server" to a general-purpose AI environment, which provides and interactive user interface, or controls the server autonomously.

INTRODUCTION

This paper reports work carried out within the BIOLAB consortium of the CEC Advanced Informatics in Medicine (AIM) programme. The overall purpose of the work is to assess the potential for commercially available parallel computer systems to enhance practical applications of medical image analysis. Here, we report methods developed for using the Active Memory Technology (AMT) DAP as a "compute server" to a network of workstations. We report three technical developments.

- A parallel implementation of the "snake" algorithm of Kass, Witkin & Terzopoulos, extended to deal with closed contours ("bubbles").

- A software interface between a development environment on the remote SUN workstations and the DAP, providing either autonomous or interactive control of the "bubbles".

- A demonstration of the use of the DAP as a more conventional "compute server" to the workstations, giving access to a set of conventional low-level algorithms, forming part of the AMT image-processing library.

The configuration is intended to assess the opportunities for using specialised parallel computers within a network of biomedical workstations. The DAP acts as a remote-access engine giving considerably improved performance over a single workstation in the network. The workstation displays the images to the user and provides an interactive interface, allowing high-level control of the system.

The DAP facilities have been integrated within the Interactive Vision Environment (IVE, initially developed under Alvey grant MMI-007), which runs within the AI development environment POPLOG. The numerical computing power of the DAP is thereby coupled to the general-purpose reasoning power only available in serial machines. The benefits and costs of the SIMD machines such as the DAP in this context are examined and reported.

THE AMT-DAP 510: OVERVIEW

The DAP is a commercially available array processor, whose basic design has not changed since the 1970s. It represents a cost-effective way of increasing raw computing power for some applications.

The DAP 510 processor consists of an array of 32 by 32 processor elements (PEs), connected in a fixed square grid (AMT,1988). Each PE is a 1-bit computer, running at a clock rate of 10MHz. This gives each processor about the power of a PC or a SUN 3, but all arithmetic is carried out bit-serial, so the realisable performance greatly depends on the algorithms employed.

All PEs carries out instructions under the control of a single program. Each PE has 4Kbytes of local memory which is also used to store the images. There is no general shared memory. The DAP has a hi-resolution, colour graphics monitor, served by a frame-buffer that can be read or written in parallel by the PEs.

The DAP is interfaced to a host SUN 3/50, by means of a SCCSI port, which carries all communications between the DAP and the host (and hence the SUN network).

The DAP is normally programmed in an extended form of Fortran which allows automatic use of vector and matrix parallelism. A library of standard control software on the host can be called from either Fortran or C. This includes software for passing data to and from the DAP, loading, starting and interrupting the DAP, and for passing SUN mouse information to the DAP.

ACTIVE CONTOURS: RESUMÉ

Deformable models have been used for a variety of purposes in vision (e.g. Burr, 81, Kass et al 87, Mowforth et al 1989). The central idea is to define structures which distort under the influence of an image, whilst conforming to pre-defined constraints on shape. Iterative methods are usually employed, so that the model deforms dynamically. When stable, *it adopts a position within the image which is optimal* (in the sense defined by the deformation forces, and for given initial conditions).

The work reported here defines a one-dimensional elastic structure, similar to a string of beads, looped into a closed band. The elastic band is subject to three internal forces:

(i) An elastic force, tending to bring all control points (the beads) together

(ii) A stiffness force, tending to round off abrupt corners, and distributing any change in tangent around the band.

(iii) An internal pressure, which tends to inflate the band outwards.

The internal pressure, and the elastic forces reach equilibrium

at a particular size, to form a 2D "bubble". In addition, the elastic forces may be distributed inhomogeneously around the bubble, as if the elasticity were greater in places. This causes it to stretch selectively under the effect of the internal pressure, and thus prefer to take up a particular shape.

A fourth force, is an external force due to the image, which pulls the control points towards higher grey-values. The original images may be pre-processed to create intermediate images which encode any desired feature as a scalar. In our work to date we have typically used as intermediate images either a gaussian smoothed image, or the modulus of the gradient of a smoothed image; these emphasise bar-like or edge-like features respectively.

By simulating the effects of forces at the control points, the bubble is free to move in the image to minimise the aggregate force. It converges over time to a position which can be used to identify regions in the image which conform to the pre-defined shape.

There are two major interests of using deformable models in medical image analysis.

- The final position adopted by the bubble represents a global best fit between the expected model, and the image data (for given initial conditions). Missing data or edge drop-out due to imperfect low level image processing can thereby often be corrected.

- Bubbles (or other deformable models) can be biased to select structures of specific shape. This allows expectations about the form and position of specific anatomical structures to be used in the search process.

The major disadvantages of the technique are

- It is very sensitive to initial conditions. In problems such as often confront medical image processing, the initial conditions can be specified by consideration of the clinical context.

- It relies on iterative methods which are usually computationally expensive. However, the algorithm is essentially parallel, and can readily be implemented on the DAP.

The parallel implementation brings about a significant improvement in processing speed. The algorithm then becomes a practical tool, which is likely to be of particular benefit when used interactively in medical image processing.

IMPLEMENTATION

The DAP510 comprises an array of 32x32 processors. We therefore define each bubble as one or more sequences of 32 control points, each point being assigned to one processor. The full array is then able to compute 32 sequences in parallel, which can be assigned to arbitrary combinations of bubbles each having multiples of 32 points (e.g. 32 x 32-point bubbles, 4 x 256-point bubbles, 1 x 1024-point bubble etc).

The positions of each control point is advanced stepwise in time, and each timestep comprises three phases. During the first or FORCE phase, the local gradient of the image is found at each control point. This requires random access to the 512x512 byte image, stored amongst the PEs. This is achieved by a tightly coded scalar loop in APAL assembler code.

In the next or MOVE phase, the net force at a point is computed from the local image force, internal forces due to

the positions of its neighbours (tension and stiffness), and the internal pressure term. The equations of motion are:

$$\frac{dx}{dt} = \frac{d}{ds}\left(T\frac{dx}{ds}\right) - S\frac{d^4x}{ds^4} + P_x - \frac{dI}{dx}$$

$$\frac{dy}{dt} = \frac{d}{ds}\left(T\frac{dy}{ds}\right) - S\frac{d^4y}{ds^4} + P_y - \frac{dI}{dy}$$

In these equations, s is a parametric variable advancing by unity between successive control points. The first term on the RHS gives the effect of an elastic string with an elastic constant T, which acts as a variable tension in the string proportional to the separation between bubble points. With this term alone, the exact solution between two fixed points is a straight line with equidistant bubble points, as one would expect for an elastic string. The second term on the RHS provides the bubble with stiffness determined by the value of s, and a tendency to resist sharp changes of curvature. The third term is a constant internal pressure force which acts along the line joining the centre of gravity of the bubble points to the surface bubble point in question. In order to prevent the pressure force from translating the bubble in the absence of an image, the average value of the pressure force over all the bubble points is made zero. The last term is the gradient of the image intensity I, or the image force term previous calculated in the FORCE phase.

The solution of the equations of motion is obtained by finite differencing, and stability for any sized timestep is ensured by using an implicit differencing scheme (see [1] for details). This leads to a five-diagonal set of equations for the new positions of each bubble, that is to say the need to solve 32 five-diagonal sets of equations, first for x then for y. The parallel solution is obtained using the method of cyclic reduction, actually an extension to a five-diagonal matrix of the PARACR algorithm described in reference [2]. Finally, the last or GRAPHICS phase of the timestep uses the directly connected DAP display monitor to plot the new positions of all snake points on the image.

Having found all the new bubble positions in parallel, the timestep is repeated continuously, and the bubbles relax towards positions tending to lie along the bubble-shaped objects in the image. However, even when the points are stationary, the DAP is continuously computing them; that is to say the bubbles are 'alive' at all times, and will respond instantly if any points are interactively moved.

Control of Shape and Size

In the absence of an image, the steady state solution to the above bubble equations with periodic (or cyclic) boundary conditions in the s-variable, and constant values for tension and stiffness, is a circle. If the stiffness is zero then the radius of the circle is proportional to the ratio of pressure to tension, just as it would be for a soap bubble. The introduction of stiffness slightly decreases the radius of the circle. The pressure can therefore be conveniently used to vary the size of the bubble in a natural way. If the tension is varied around the bubble surface (i.e. as a function of the variable s), then the shape of the bubble can be varied.

Different harmonic functions may be applied to the elastic tension and summed. In principle this provides a simple means for defining a wide range of closed, convex shapes. Limited experiments have been carried out using the second
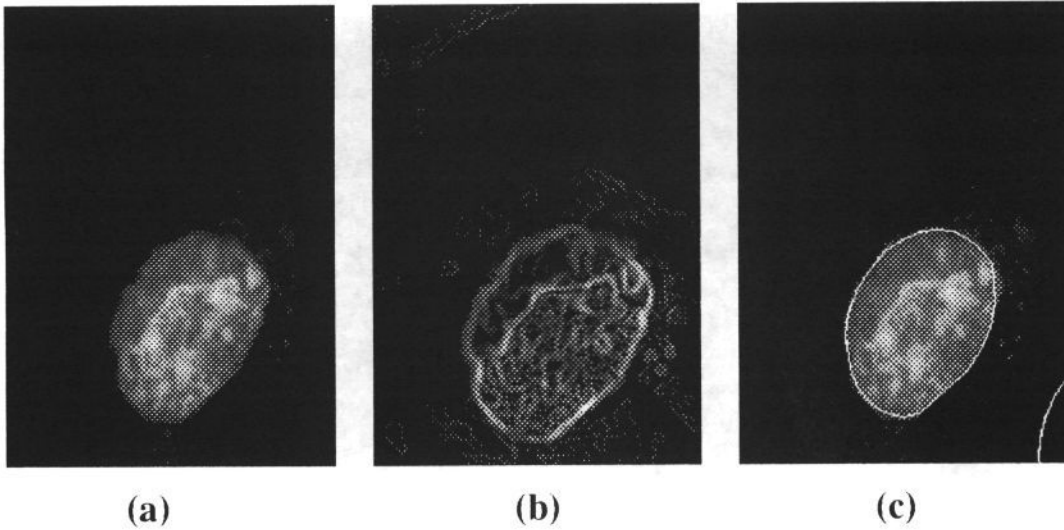
(a)   **(b)**   **(c)**

Figure 1. A single 32 node bubble interacting with an isolated cell.
(a) original image, (b) modulus of the smoothed gradient,
(c) the bubble overlaid on (a).

and third harmonics, biassing the bubble towards given elliptical or ovoidal shapes.

EXAMPLES

Figure 1 demonstrates a simple interaction between a 32-point bubble and an image of an isolated cell. The image, Figure 1(a), was first smoothed by a 2-D gaussian distribution (σ = 1 pixel), and an image corresponding to the modulus of the discrete gradient vector was computed, Figure 1(b).

A small bubble was placed inside the cell in Figure 1(b). Under the influence of the pressure term, the bubble grew, until it was caught and held by the image maxima. When stable, the bubble's position outlines the cell boundary fairly accurately - Figure 1(c). The time to stabilise depends on the match between the initial shape parameters of the bubble, and the image distribution. It is strongly affected by the time-step used in the algorithm. A large time-step causes the bubble to respond rapidly to the forces on it, but this can lead to instability and oscillation. In the examples shown here, the convergence time (for each bubble) was approximately 2-4 seconds. Better convergence schemes, based on simulated annealing methods can improve performance (see below).

The power of the parallel implementation of bubbles is more evident in images consisting of multiple cells. Figure 2 shows 32 bubbles interacting with an edge-differentiated cell image. The centres of suspected cells were first identified by finding peaks in a smoothed Laplacian image (where the scale of the smoothing is adjusted to suit the expected cell size.). Bubbles were then placed in the image at these points. In most cases the bubbles have adjusted their shape to that of the actual cells and have become attached to them. The settling time for all 32 bubbles is again about 2 seconds.

Performance

The performance of the AMT DAP relative to a typical workstation was measured by having the host computer perform the same calculation at the same time as the DAP. Timings were obtained individually for the different phases of the timestep for both the DAP and the SUN, and the overall
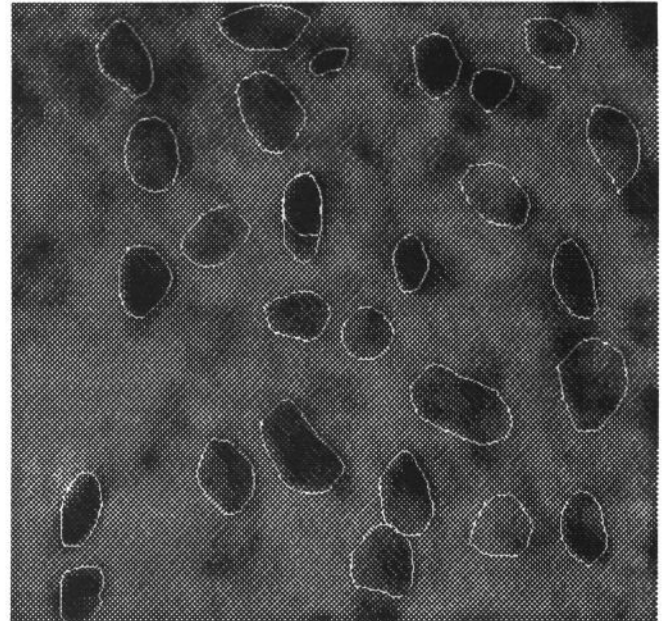

Figure 2. 32 bubbles placed automatically in parallel

speedup ratio of the calculation was calculated. Timings given below refer to the SUN 3/50 host machine.

PROCESSING PHASE	Times in msec		Speed-up Ratio
	SUN 3/50	DAP510	
FORCE	2639	70	37
MOVE	1020	18	56
GRAPHICS	4000	95	41
TIMESTEP	7659	183	41

These timings show that on the DAP only 10% of the time is spent solving the motion equations - the bulk of the time is consumed in the FORCE calculations and the GRAPHICS output. The FORCE phase is dominated by the essentially serial random image access routines, using carefully coded assembler routines in APAL which picks up one byte from
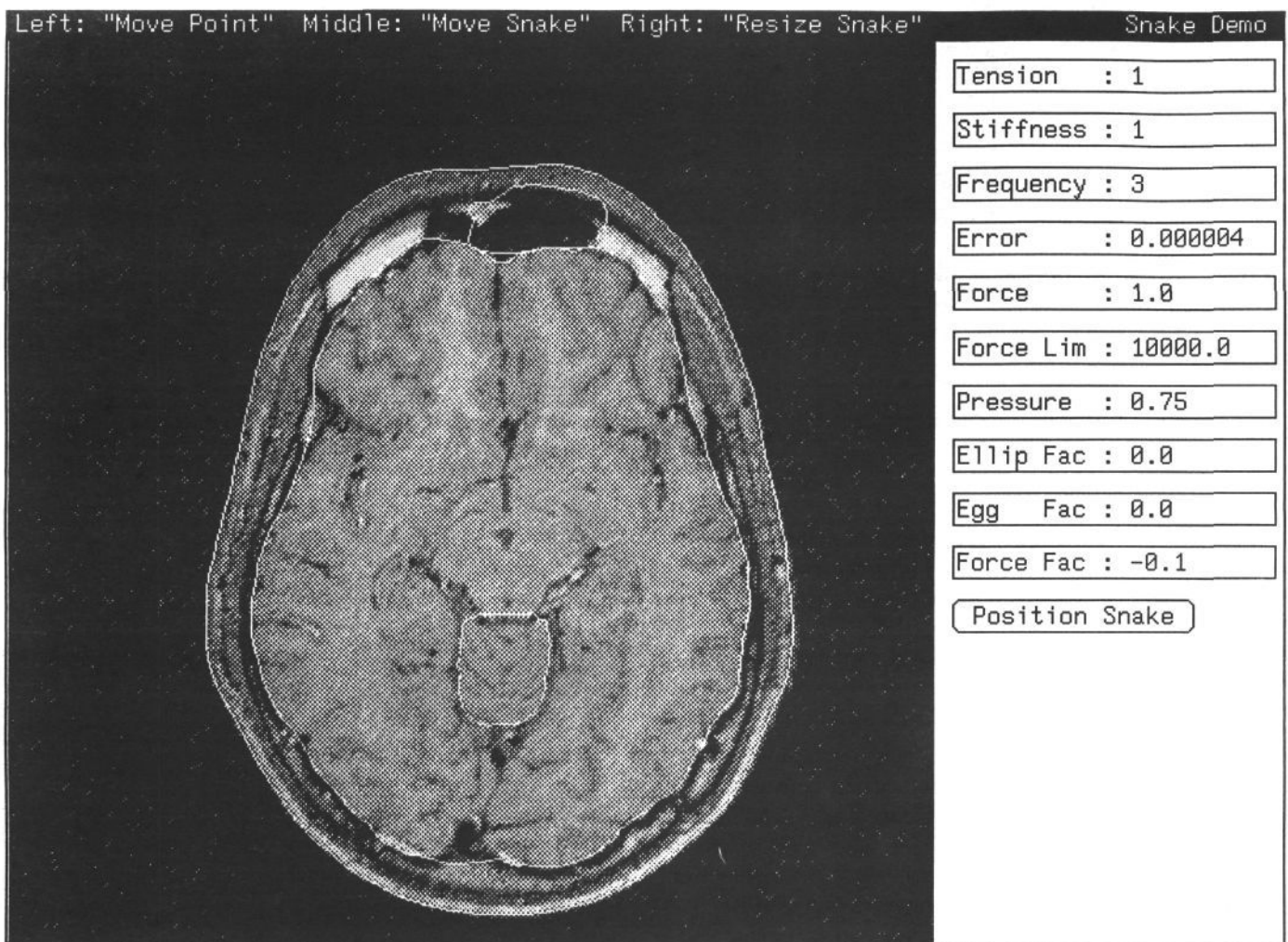
Figure 3. Interactive control panel on the SUN workstation

the image at a time.

Comparisons using a SUN 4/20 indicate that (as expected), the SUN 4 performs about 10 times as fast as the SUN3. This gives a speed-up advantage for the DAP of merely 4. However, when used in the networked configuration (see below) the graphics phases and the control parameter processing of the DAP program can be dropped. This restores the advantage to about a factor of 10.

It should be noted that the DAP-510 has fairly low processing power: a larger DAP can be configured with 64*64 PEs, and a faster machine, with 8-bit arithmetic is scheduled. In this work the DAP is used to represent SIMD compute servers; the bubble algorithm could easily be adapted to other, faster SIMD machines. More modern machines have mechanisms for shared memory, with indirect addressing in the PEs, which would greatly speed up the calculations in the FORCE phase of the program.

CONTROL MECHANISMS

Bubbles can only act to refine the localisation of structure in the image, since they must initially be placed close enough to an appropriate attractor. Unguided, they merely fall into arbitrary local minima. One potential practical role for deformable models in medical image processing applications is to provide an "intelligent" tool to assist the clinician in making interactive measurements of image features, and our initial DAP deformable model algorithm was developed for such direct use.

Direct control

Basic interactive facilities have been incorporated into the DAP program so that individual control points, or groups of points (including whole bubbles), may be picked up under mouse control and moved anywhere on the image.

The values of the Tension, Stiffness and Pressure parameters, the second and third harmonic shape parameters, and the Timestep, may also be changed interactively under mouse control.

These tools allow the broad positions and shapes of bubbles to be directed by the user, while the DAP program is continuously refining the approximation to seek the optimal local minimum of a well-defined energy function. Hand-segmentation of images is common in medical image analysis, usually by getting the clinician to out-line regions by means of a light-pen or mouse. This process is very fast, but poorly defined, and inaccurate. The parallel implementation iterates at about 4 Hz, and converges to local minima in a few seconds - this is fast enough to provide valuable support to a radiographer. The result is better controlled (e.g. it could be repeated by another operator with the same effect), and the image is likely to need less detailed initial interpretation by an operator.

Workstation control

The DAP graphics screen only allows single-user access. At present, such under-use of an array processor cannot be
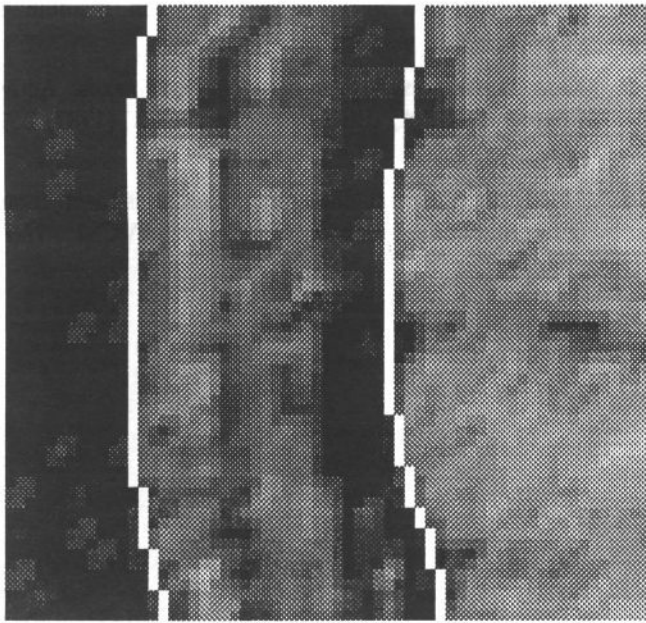
Figure 4. Close-up view of part of Figure 3, showing two bubbles, interacting with the scalp outline and the cortical surface
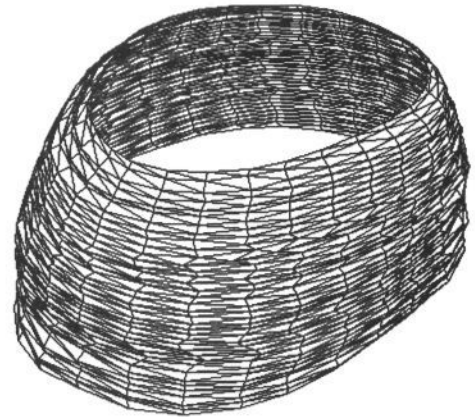

Figure 5. Example of a surface in 3D, created from bubbles applied to separate slices in MR images of the brain.

justified on cost grounds. Medical image processing applications typically occur intermittently. It is therefore attractive to seek to network the DAP as a "compute server" for many workstations. This has the additional advantage that the GRAPHICS part of the algorithm can be delegated to the workstation, thus increasing the performance by about a factor of 2 (see table).

The DAP-SUN SCCSI link gives the host computer full control of the DAP. The DAP program repeatedly checks the host SUN for messages, requesting a read or write of either the system parameters or the position of the control points. This allows a program written on the host machine to control the compute engine.

To provide for user interaction with the program on the SUN workstations, we have developed a simple link between the DAP and POPLOG running on SUN 3 or 4 (or other) machines within the network. Within POPLOG, the DAP engine has been integrated into a vision systems development tool developed and used extensively in house (the Interactive Vision Environment - IVE). Figure 3 shows a screen dump from a typical session.

This facility allows the DAP programs to be used on a single-user basis across the network. Multiple users are managed by the existing multi-tasking operating system of the DAP. In a dedicated system, a remote demon is envisaged to improve the efficiency of time sharing between multiple workstations.

LOW-LEVEL IMAGE PROCESSING

An extensive range of software has already been developed to run on the DAP, and carefully optimised calls exist in the DAP Image Processing Library to carry out many standard filtering operations. These include most of the pre-filtering used in the examples shown here. Further work has also been carried out to integrate the DAP as a compute server to the IVE for these functions. The DAP is particularly well suited to simple filtering algorithms, since each PE mainly accesses a portion of the image stored in its local memory.

DISCUSSION

We have described a SIMD implementation of the algorithms reported by Kass, Witkin and Terzopoulos. The use of the parallel computer makes new methods for interactive image processing feasible. Such processing is likely to be of particular importance in the domain of medical image processing, where the clinician often needs to control the analysis of an image by hand, yet would greatly welcome increased assistance from the computer in outlining important structures.

Much time is currently spent in radiological laboratories to identify structures in images manually, to identify regions which require further processing. The example shown in Figure 1 is of a cell from a cervical smear: the degree of irregularity in the boundary of the cell is a primary indicator of a pre-cancerous condition. The cells in Figure 2 have been stained by means of mono-clonal antibodies which attach to the active sites involved in rheumatoid arthritis; the type of staining, and the size and shape of the stained cells are of importance for contemporary research into the causes and treatment of rheumatism.

Active deformable models can form a variety of structures. The smooth, closed "bubbles" described here have particular use in cytological applications, but other structures are currently being investigated. Figure 4 shows the outline scalp extracted from MR slice images. Each slice was pre-filtered to create an image of the modulus of the gradient. A bubble was placed near the edge of the image, outside the head region, with low pressure. It rapidly shrank and was trapped onto the outer gradient peak. The elastic term was then reduced, so that the position of the bubble was dominated by the image data.

Figures 3 and 4 were captured during this operation. The cortical boundary has also been approximated in these images by a second bubble, half the size of the first bubble, placed inside the scalp and then expanded by increasing the pressure. This reliably avoids many of the artifacts associated with the

use of edge maps, as reported in Attwood et al (1990).

In principle it should be possible to process multiple slices simultaneously, though software to do this is still under development. This would allow 32 slices such as are shown in Figure 4 to be computed in parallel, and outlines similar to these to be computed in a few seconds. The further use of DAP graphics routines for viewing and shading the results are also intended.

Two other extensions of this work are of special interest:

(i) The use of higher order terms in the definition of the shape of the bubble. Bubbles are biassed by the harmonics in the elasticity, and by the choice of starting conditions. It is therefore possible to model more complex structures, such as the ventricles of the heart in cine-angiography, or organs of the brain in cranial radiography. These models can be stored in a database of anatomical shapes, called up according to the clinical context.

(ii) Methods for forming 2-D rubber sheets, in a 32*32 array. Many medical imaging methods produce series of slices such as in Figure 4, which capture the 3-D structure of organs being imaged. It is straightforward to extend the 1D snakes (moving in a 2D image) to 2D membranes (moving in 3D data). This will provide the ability to "shrink-wrap" 3D structures directly, rather than building them from independently processed slices.

CONCLUSION

The very high speed implementation on a parallel machine makes the techniques of deformable models feasible for clinical and research applications. The speed-up achieved transforms algorithms for deformable models from a computational curiosity, to one which would appear to have immediate practical significance. This implementation is based on an SIMD machine, available commercially for the cost of a typical mini-computer. This order of cost can be justified in a networked, multi-tasking system.

ACKNOWLEDGEMENTS

REFERENCES

1. AMT Technical Overview: The AMT DAP 500. 1988

2. Attwood, CI, Sullivan GD, Robinson, G., Baker KD and Colchester ACF. BMVC90, 1990 (These Proceedings)

3. Burr, D.J. Elastic Matching of Line Drawings. IEEE PAMI-3, 6, 708-713 (1981)

4. Hockney R W and Jesshope C R, "Parallel Computers 2", Adam Hilger/IOPP, Bristol and New York (1988) 479-483. Distributed in USA by American Institute of Physics, NY.

5. Kass M, Witkin A, Terzopoulos D, "Snakes: Active Contour Models", IEEE Proc. First ICCV (1987) 259-268

6. Mowforth, P.H and Zhengping, J. Model-based tissue Differentiation. Proceedings of the AVC89, 67-73 (1989).