A Video Based Tracker for Use in Computer Aided Surgery

Neil Maitland and Chris Harris

Roke Manor Research Limited, Romsey, Hampshire

Recent improvements in medical equipment and data visualisation tools have made it possible for surgeons to use medical images of patients during operations as well as during the preparation phase. In order to use the data effectively in the theatre the surgeon must be given live information about his current location inside the patient relative to the pre-operative images. This paper describes a vision-based system that can locate and track a passive hand-held instrument in the operating theatre to provide the real-time information required.

A target acquisition system is described which locates a pre-defined object in an arbitrary scene and calculates its 3D location and attitude. Enhancements to the RAPiD model-based tracking system are also detailed including multiple camera operation, improvements to edge identification in the image and combining robust tracking with high accuracy. Experimental results are also provided.

1 INTRODUCTION

VISLAN

This paper concerns one component of VISLAN, a system being developed for computer-assisted neurosurgery (Colchester, 1994; Evans, 1994). VISLAN aims to assist surgeons in two stages. First, the system will help in the registration and segmentation of the various pre-operative data sets produced by modern 3D scanning methods such as X-ray computed tomography and magnetic resonance. This will ease the planning stage of the operation. Secondly, once the patient is in the theatre the surgeon will be assisted by the system in implementing his plans by interactive displays.

In order to provide these intra-operative displays, the pre-operative data must be aligned with the patient in theatre. This will be achieved using a vision system that will locate visible landmarks on the patient during surgery - eg skin surface shape; blood vessels in the brain - with matching features in the pre-operative data. Once the pre-operative and intra-operative coordinate systems are aligned, a special locator tool (referred to as the pointer) or the surgeon's instruments will be tracked using vision and their position indicated on the pre-operative data. It is this real-time tracking system for hand-held tools that will be described in this paper.

Using a system based on vision provides flexibility and power, offering advantages over alternative systems of patient and tool location, for example mechanical arms (eg Galloway *et al*, 1992) which can be cumbersome, or sonic (Reinhardt *et al*, 1993) or LED (Krybus, 1991) systems which require active tools connected by wires. VISLAN uses passively lit targets that will not unduly disrupt the surgeon's normal operating methods.

Tracking System

It is the eventual aim of the VISLAN tool tracking system to be able to provide the coordinates of the surgeon's instruments in real-time during an operation. However, the initial requirements allow a specially designed pointer tool to be used. The basic requirements for the tracking system are to:

- provide 3D location and attitude of a hand-held instrument.
- be accurate to the order of 1mm maximum error at the instrument's tip.
- act on passive instruments that are of similar size and weight to those currently used by the surgeon.
- to be convenient to use in an operating theatre.

BMVC 1994 doi:10.5244/C.8.60

To implement these requirements a video-based tracker has been developed using a pair of stereo cameras and a tracker tool with high contrast markings printed on it. Tracking takes place in two stages: an estimate is obtained for the initial position and orientation of the pointer (acquisition) and then it is tracked in real-time using a version of the RAPiD model-based tracker.

The functioning of RAPiD has previously been reported for a single camera (Harris and Stennett, 1993). However, with this monocular system, errors in the measurement of range were found to be too large for use in VISLAN. Using multiple cameras for this new application has reduced the range errors significantly making it a practical solution. Further refinements have been made to improve tracking in cluttered scenes, and to improve the accuracy when processing power is limited.

2 ACQUISITION

Acquisition Target Design

As described above, the function of the acquisition procedure is to find an initial 3D location and attitude of the tracked tool to start off the real-time tracker which can only search a limited image area for speed. The problem may therefore be defined as finding a pre-determined object in an arbitrary and possibly cluttered video scene using monochrome cameras.

Fortunately the object to be found, the target, can be carefully designed to ease the acquisition process. The main design decisions are listed below:

- It must be possible to resolve all six 3D degrees of freedom.
- The target should be binary to provide maximum contrast in the monochrome images.
- It should be planar - this makes it easier to construct and avoids problems with varying illumination over the object.
- A careful trade-off must be made between the likely uniqueness of the target in an arbitrary scene against the number of calculations required to locate it.
- The acquisition algorithm should run quickly without special hardware on standard computer equipment such as a PC or workstation.

Many simple shapes may be discounted either as being non-unique or as containing insufficient information to fully resolve the pose. The chosen design, the binary acquisition target or BAT, complies with all of the requirements and has been shown to operate quickly and reliably in practice.

The BAT (figure 1) is constructed from fourteen equilateral triangles forming a hexagonal shape with three concavities. It is found in a given image by a series of filtering stages and then the pose is calculated using a model fitting function. These are described in the following sections.

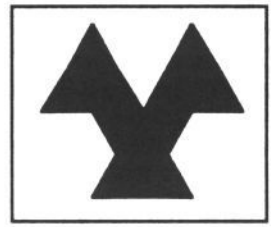

Figure 1. The Binary Acquisition Target (BAT)

Target Location

The major stages in the filtering an image are listed in order below.

1 Local Mean Removal

The first filtering stage must operate at the pixel level on the image, and therefore should be cheap on processing to limit the time required. Thus, for example, canny edge extraction, with several multiply-accumulate operations per pixel would be too slow. The method used is local mean removal (LMR) - a simple square convolution function - followed by level thresholding. This produces a binary image which should contain the high contrast BAT shape if present in the grey level image.

2 Connected Component Find
The next operation identifies connected components in the binary image. Memory requirements are minimised by using a rolling buffer and passing components to the next filter immediately after completion.

3 Boundary Walk
If the size of a connected component is considered great enough to allow the BAT to be resolved, a boundary walk is performed to produce a list of pixels on the perimeter of the shape. The average contrast is also calculated between pixels inside the perimeter and those outside, and checked against the highest contrast found so far in a connected component that has been identified as a valid BAT. Another test is then performed to ensure that the component is not too thin to be resolved.

4 Corner Count Checks
A recursive binary chopping procedure is then used to identify those points on the perimeter that lie on one of the concavities. The six angular transition points (corners) are then identified and a filter removes any components that have the wrong number.

5 Geometry Check
Now a geometric check is made of the shape based on the model. For an orthographic projection of the hexagonal shaped BAT in an arbitrary orientation (see figure 2), it may be shown that the ratio of lengths A:B:C must be 1:2:1. This is extended to the perspective projection for this application.

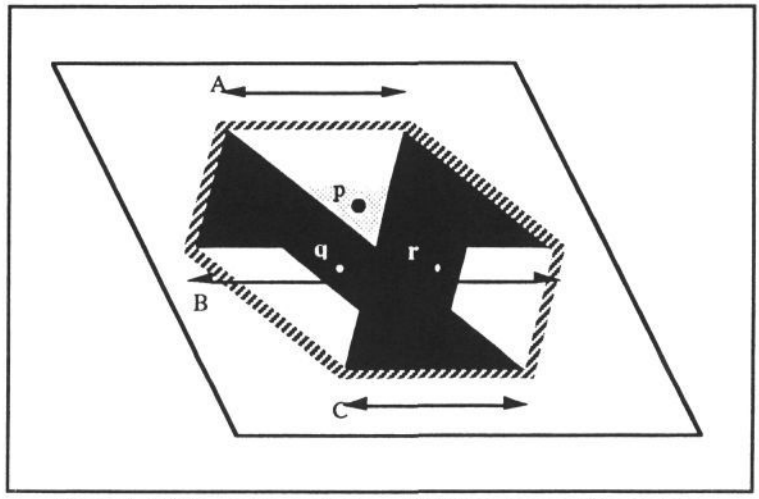

Figure 2. Orthographic projection of BAT on to an arbitrary plane.

Pose Resolution

The six corners of the BAT obtained from the initial filtering are converted into homogeneous coordinates, ie directions, by use of a set of camera calibration parameters (Harris, 1993). These are then processed as described below to calculate the BAT pose.

Problem: given set of n 3D model points $\{\mathbf{R}_k\}$ and matching image points in homogeneous coordinates $\{\mathbf{r}_k\}$, what is the best estimate of the pose (ie. attitude and position) of the model in camera coordinates? To define the model pose, a point at $\mathbf{R}$ in model coordinates will be seen at location $\mathbf{R}'$ in camera coordinates, where

$$\mathbf{R}' = A\mathbf{R} + \mathbf{T}$$

Here A is the orthonormal rotation matrix, and $\mathbf{T}$ is the translation vector. This point will be imaged at

$$\mathbf{r}_k' = (X_k', Y_k') / Z_k'$$

$$= (A_{11}X_k + A_{12}Y_k + T_x,\ A_{21}X_k + A_{22}Y_k + T_y) / (A_{31}X_k + A_{32}Y_k + T_z)$$
$$\equiv (a\,X_k + b\,Y_k + c,\ d\,X_k + e\,Y_k + f\ ) / (g\,X_k + h\,Y_k + 1)$$

We then need to minimise the image-plane error between observations and projected model, that is to minimise

$$E'(a..h) = \sum_{k=1}^{n} | \mathbf{r}_k' - \mathbf{r}_k |^2$$

but this is difficult, so instead assume the dependence on range can be ignored and minimise the now quadratic objective function

$$E(a..h) = \sum_{k=1}^{n} | (\mathbf{r}_k' - \mathbf{r}_k).(g\,X_k + h\,Y_k + 1)|^2$$

$$= \sum_{k=1}^{n} ((a\,X_k + b\,Y_k + c) - x_k\,(g\,X_k + h\,Y_k + 1))^2 +$$
$$((d\,X_k + e\,Y_k + f) - y_k\,(g\,X_k + h\,Y_k + 1))^2$$

$$= \sum_{k=1}^{n} ((\mathbf{u}_k.\mathbf{p} - x_k)^2 + (\mathbf{v}_k.\mathbf{p} - y_k)^2)$$

where

$$\mathbf{p} = (a, b, c, d, e, f, g, h)^T$$
$$\mathbf{u}_k = (X_k, Y_k, 1, 0, 0, 0, -x_kX_k, -x_kY_k)^T$$
$$\mathbf{v}_k = (0, 0, 0, X_k, Y_k, 1, -y_kX_k, -y_kY_k)^T$$

Optimal values of the parameters are found by setting to zero the differentials $\partial E/\partial \mathbf{p}^T = \mathbf{0}$

$$\sum_{k=1}^{n} \left(\mathbf{u}_k\,\mathbf{u}_k^T + \mathbf{v}_k\,\mathbf{v}_k^T \right) \mathbf{p} = \sum_{k=1}^{n} \left(x_k\,\mathbf{u}_k + y_k\,\mathbf{v}_k \right)$$

which can be written as the equation

$$\mathbf{M}\,\mathbf{p} = \mathbf{w}$$

Solving this order-8 matrix equation for the optimum parameters, $\mathbf{p}^* = \mathbf{M}^{-1}\,\mathbf{w}$.

Now, not all eight components of $\mathbf{p}$ will be reliable, especially if the perspective is weak. The most reliable components will be the 6 constant and linear expansion coefficients. Expanding for small X_k, Y_k

$$\mathbf{r}'_k \approx (a\,X_k + b\,Y_k + c,\ d\,X_k + e\,Y_k + f) - (g\,X_k + h\,Y_k)\,(c, f)$$
$$= (a'\,X_k + b'\,Y_k + c,\ d'\,X_k + e'\,Y_k + f')$$

where

$$a' = a - c.g$$
$$b' = b - c.h$$
$$d' = d - f.g$$
$$e' = e - f.h$$

There are 6 reliable coefficients (a',b',c,d',e',f), which are sufficient to solve for the 6 parameters of the pose.

$$A = T_Z \begin{pmatrix} a'+c.g & b'+c.h & \ldots \\ d'+f.g & e'+f.h & \ldots \\ g & h & \ldots \end{pmatrix}$$

The orthonormality of the first two columns of the rotation matrix, A, results in three equations to be solved for T_Z, g and h. This can be done in closed form, but results in two solutions, arising from the two roots of a quadratic equation. For each solution, the objective function

$$E(\mathbf{p}^*) = |\mathbf{w}|^2 - 2\,\mathbf{w}.\mathbf{p}^* + \mathbf{p}^*.(M\,\mathbf{p}^*)$$

is evaluated with $\mathbf{p}^* = (a' + c.g, b' + c.h, c, d' + f.g, e' + f.h, f, g, h)^T$, and for the correct solution it should be approximately n times the squared pixel measurement accuracy. If perspective effects are strong, the incorrect solution will be much larger, but in the orthographic limit both solutions will give small costs, and so a two-fold ambiguity results.

Next, the 3-fold rotational degrees of freedom of the BAT can be resolved by examining the three image pixels located at points p, q and r (see figure 2). The one which is brighter than the others, must correspond to the 'missing triangle', and therefore fixes the pose of the BAT.

Stereo Views

When using a stereo calibrated camera setup, the pose obtained above may be refined, particularly in the range measurement and the two-fold pose ambiguity for an orthographic view.

The BAT is located in the second view using the filtering process described above. To speed the process up, the search is carried out over a small image rectangle defined by the projection of the epipolar line through the BAT from the first camera into the second over a limited set of ranges. The BAT pose is solved as before, but when faced with the two-fold pose ambiguity, both poses are saved for each camera view. The two results are then combined using epipolar geometry to provide a greatly improved estimate for the target range.

Now the correct attitude must be found given two possibilities for each view by finding a cost for all four combinations of the two results from the views. An evaluation matrix, B, is calculated by:

$$B = \tilde{A}^* A_{\pm}^{(0)} \tilde{A}_{\pm}^{(1)}$$

where $A_{\pm}^{(n)}$ is the 3x3 rotation matrix describing either the positive or negative pose of the target for view n, and A^* is the camera pose for view 1 in view 0 co-ordinates. For the correct pose B should approach the unit matrix, so a scalar cost function is given by the trace of B. The correct pose is taken to be the one which produces the largest trace.

3 TRACKING

As stated previously, the tracking system used in VISLAN is based on the monocular system described by Harris and Stennett (1990) known as RAPiD. This system uses a model of the object to be tracked consisting of a number of high contrast edge features. Tracking is achieved by projecting each point on to the image plane from an estimate of the current pose and then searching a single pixel width bar perpendicular to the edge. The difference between expected and observed edge positions is recorded and then passed to a minimiser which calculates a new estimate of pose. A Kalman filter is used to provide an accurate estimate of pose for the next frame (Evans, 1990). The advantage of the RAPiD tracker over many similar systems is that it requires very little processing power to run in real-time, as the amount of pixel-level processing is limited to a number of short bars.

The three main changes that have been made to the original RAPiD tracker for the VISLAN pointer application are outlined in the following sections.

Upgrading to Stereo

Consider first a binocular stereo system. Designate one camera as prime, and the other camera as secondary. Associate with each a local camera coordinate system, called respectively *primary* and *secondary* . Local camera coordinates will have its origin at the camera pin-hole, Z-axis along the optical axis, and X and Y axes parallel respectively to the horizontal and vertical pixel axes. Let the

secondary camera be located at $\mathbf{T}^*$ in primary coordinates, and possess an attitude defined with respect to the primary coordinates by the rotation matrix $\mathbf{A}^*$. Thus a point at $\mathbf{R}_0$ in primary coordinates would be situated in secondary coordinates at

$$\mathbf{R}_1 = (\mathbf{A}^*)^T (\mathbf{R}_0 - \mathbf{T}^*) \tag{1}$$

Consider a point P on the model, situated at $\mathbf{P}_0$ in model coordinates, which are translated, but not rotated, from the primary camera coordinates. Let the model origin be at $\mathbf{T}_0$ in primary coordinates, so that in secondary coordinates it is at

$$\mathbf{T}_1 = (\mathbf{A}^*)^T (\mathbf{T}_0 - \mathbf{T}^*) \tag{2}$$

Consider a small change in the model pose of $\mathbf{q}_0 = (\Delta_0, \theta_0)$, where Δ_0 is a change in the model position, and θ_0 is the rotation-vector specifying change in the model attitude about the model origin, whose direction is the axis of rotation, and whose magnitude is the angle of rotation in radians about this axis. This small change will thus move the point P to

$$\mathbf{R'}_0 \approx \mathbf{T}_0 + \Delta_0 + \mathbf{P}_0 + \theta_0 \times \mathbf{P}_0 \tag{3}$$

In the secondary camera coordinates, this change in model pose will be experienced as a change in pose of $\mathbf{q}_1 = (\Delta_1, \theta_1)$,

which will move the point P to

$$\mathbf{R'}_1 \approx \mathbf{T}_1 + \Delta_1 + \mathbf{P}_1 + \theta_1 \times \mathbf{P}_1 \tag{4}$$

where

$$\mathbf{P}_1 = (\mathbf{A}^*)^T \mathbf{P}_0$$

Noting that equations (1) to (4) will be true for all model points, P, we find that

$$\Delta_1 = (\mathbf{A}^*)^T \Delta_0$$

$$\theta_1 = (\mathbf{A}^*)^T \theta_0$$

which can be written as

$$\mathbf{q}_1 = \begin{pmatrix} (\mathbf{A}^*)^T & 0 \\ 0 & (\mathbf{A}^*)^T \end{pmatrix} \mathbf{q}_0 \tag{5}$$

Denote the model control points tracked by the primary camera by the index i, and those tracked by the secondary camera by the index j. As with the monocular RAPiD system (Harris and Stennett, 1990), denote the image displacements between the predicted control point and its observed matching edge by {d}, and the differential pose vector for the point by {c}. Then the best estimate of the small change in model pose, $\mathbf{q}_0$, is obtained by minimising the combined residual errors for the two cameras:

$$E(\mathbf{q}_0, \mathbf{q}_1) = \sum_i [\, d_i + \mathbf{q}_0 . \mathbf{c}_i \,]^2 + \sum_j [\, d_j + \mathbf{q}_1 . \mathbf{c}_j \,]^2$$

Now since $\mathbf{q}_1$ is related linearly to $\mathbf{q}_0$ by equation (5), we can define

$$\mathbf{c}^*_j = \begin{pmatrix} \mathbf{A}^* & 0 \\ 0 & \mathbf{A}^* \end{pmatrix} \mathbf{c}_j$$

and so express E as a function of $\mathbf{q}_0$ alone

$$E(\mathbf{q}_0) = \sum_i [\, d_i + \mathbf{q}_0 . \mathbf{c}_i \,]^2 + \sum_j [\, d_j + \mathbf{q}_0 . \mathbf{c}^*_j \,]^2$$

Since E is quadratic in q_0, it can be minimised simply by solving an order-6 matrix equation, just as in monocular RAPiD.

Improving Robustness

The original version of RAPiD converted to run with multiple cameras was found to operate reasonably well in scenes with an uncluttered background. However, in more realistic situations it was found that tracking was often 'pulled' away from the model by high contrast edges in the background scenery. This was generally because the edge searching algorithm examined only a single line of pixels running approximately perpendicular to the expected edge position. If this search bar crossed more than one edge, the one with the highest contrast was assumed to be that from the model. Decreasing the length of the search bar reduced the number of spurious edges, but inaccuracy in model position estimates often lead to edges not being found at all.

The implemented solution to this problem is to find a set of candidate edge positions in the search bar and then to compare the angle of each of them with the expected angle of the model's edge in the image. This is achieved by obtaining a 3x3 section of image centred on the detected edge and applying horizontal and vertical Sobel edge detectors to find the edge strength in orthogonal directions. The difference in angle between the detected and expected edge directions is then found from their dot product, avoiding the use of slow trigonometric functions. The edge which has the minimum directional error and meets predefined angle and contrast thresholds is selected.

Improving Accuracy

For maximum robustness, the amount the model moves between image frames needs to be minimised, thus a high frame rate is desired; for maximum accuracy the number of edge points contained in the model should be maximised. In practice, the amount of processing power available is limited (currently to a Sparc IPX) and a compromise must be reached between these two requirements.

However, for use in VISLAN it is expected that the pointer will be used in two distinct ways. Firstly, it might be used for identifying particular static points on the patient so that, for example, the tissue type can be identified from pre-operative images; for this high accuracy will be required. Secondly, the pointer may be used in more dynamic situations, such as selecting an area of interest in an image or tracking the surgeon's tools inside the patient; here robust tracking will be needed. Thus, the accuracy and robustness must be available, but not necessarily at the same time.

This dynamic trade-off has been implemented in the software as follows. When the velocity calculations from the Kalman filter indicate that there is little movement for several successive frames, the static mode is entered which processes a large number of edge features in the image for accuracy, but does not operate at full field rate (50Hz). In order to prevent tracking being lost when the model begins to move, a cumulative total is kept of the difference between expected and detected edge positions while dealing with each frame. If this total becomes large the pointer has obviously started moving and image processing of that frame is immediately cut short. The system then drops back to the standard tracking mode in which a limited number of points are tracked at frame rate.

4 EXPERIMENTAL RESULTS

The current prototype pointer tool (figure 3) is made from a shaped block of aluminium. A black and white pattern suitable for tracking is screen printed on to a silicone rubber sheet which is then glued to the aluminium. The reason for using this unusual construction method is that it can withstand sterilisation in a surgical autoclave, as required by the VISLAN project. Future prototypes may consist of gas sterilised stickers or plastic attachments which can be fitted to a range of surgical instruments.

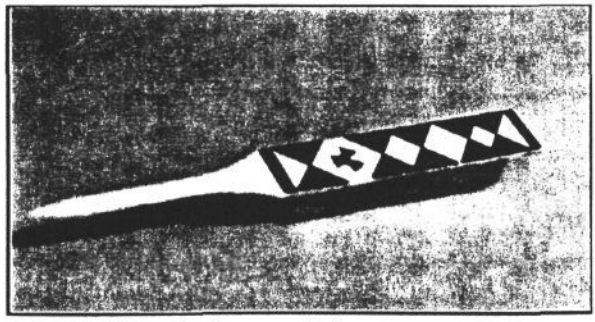

Figure 3. Prototype Pointer Instrument

Acquisition

The basic BAT acquisition system has been tested in a wide range of situations, not only in the VISLAN project, and has been found to work very effectively under varying conditions. The use of stereo in the VISLAN version improves the performance significantly, particularly on the small BATs (14 mm diameter) used on the pointer.

Table 1 shows the time taken for the system to locate BATs of different sizes both in mono and in stereo on a standard Sun Sparc IPX fitted with an SBUS 768x576 video capture card. For larger BATs, the system sub-samples the image, thus speeding it up significantly.

Table 1. Time Taken to Acquire BAT

Range	Bat Size		Sub-sampling	Time to Acquire (secs)	
mm	mm	pixels	factor	mono	stereo
1194	80	160x176	16	0.241	0.288
1963	80	96x96	8	0.127	0.160
1006	7	16x16	2	0.957	1.470
1341	7	12x11	1	2.803	3.360

Tracking

The current implementation of the software runs on a Sun Sparcstation IPX fitted with a Datacell SBUS video capture card. Using this setup two basic types of testing have been undertaken: tracking robustness and accuracy of measurement. It has been very difficult to measure the former category quantitatively, although comparative testing of the pointer through successive generations has shown subjective qualitative improvements. However, quantitative accuracy measurements have been made using our camera calibration tile which forms a set of known points on an arbitrary planar surface.

First, the location and attitude of the tile is found using a version of our camera calibration software (Harris and Teeder, 1993) and the tracker outputs the tip position in the tile's local co-ordinate system. Next, the pointer is placed on each of the tile's 'dots' in turn (see figure 4) and the co-ordinates logged - the software automatically insures that points are only recorded when running in the high accuracy mode to ensure minimum errors. The results are then analysed statistically and graphical results are produced. A large number of tests have been performed examining different aspects of the pointer's behaviour, such as:

- Dependence of measured tip location on:
  - pointer attitude
  - lighting
- The effect on overall accuracy of:
  - camera separation
  - camera vergence angle
  - camera range
  - number of tracked model points

Table 2 shows the results of some basic accuracy measurements for two camera configurations. It may be seen that the accuracy improves for a larger camera separation, particularly along the cameras z-axis.

The graphs in figure 5 show the pointer 3D location over time during a short section of the accuracy test procedure. The measurements are shown in tile coordinates relative to the nearest calibration point for the wide camera separation configuration. Each division on the horizontal axis represents 2 seconds and each vertical division 2mm with a solid line along the zero axis. The large fluctuations in the graphs occur when the pointer is moved between measurement points (approximately every 2.5 seconds) with all but one move being along the y axis. The fast tracking mode is generally entered as the pointer is moved from one point to the next, as indicated by sections of dotted line; solid lines show high accuracy mode.

Table 2. Results of Accuracy Tests

Camera Separation	Target Range	Number of test points	Average Position Error			SD of Error		
			x	y	z	x	y	z
759	950	64	0.01	-0.55	0.21	0.28	0.36	0.17
227	950	64	0.67	-0.99	0.49	0.49	0.33	0.22

Note: All measurements are in mm

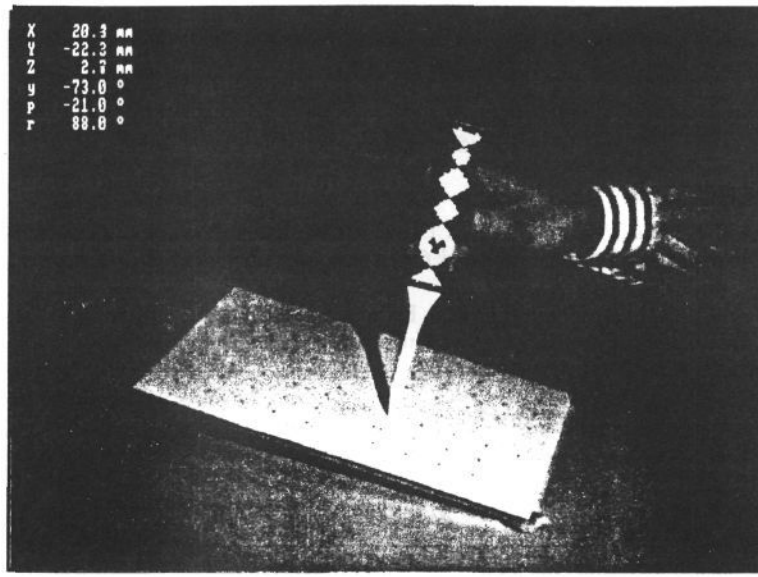

Figure 4. Measuring Accuracy with Calibration Tile

5 CONCLUSIONS

The version of RAPiD that has been developed for VISLAN has had many enhancements to the basic algorithms. This paper has discussed several of the major improvements, including the model acquisition system using the BAT, accuracy improvements using multiple cameras and dual mode tracking, and improved robustness with edge angle measurements.

In experiments the current software has been shown to work effectively, with results that compare favourably with alternative tracking systems, both mechanical and optical, where such systems generally have the disadvantage of trailing wires or sensing hardware attached to the pointer. In addition, the processing requirements are very modest, operating comfortably on a Sparc IPX workstation with no specialised hardware except the video capture card.

The pointer system is now ready to be integrated with the rest of the VISLAN components so that trials may begin in the operating theatre.

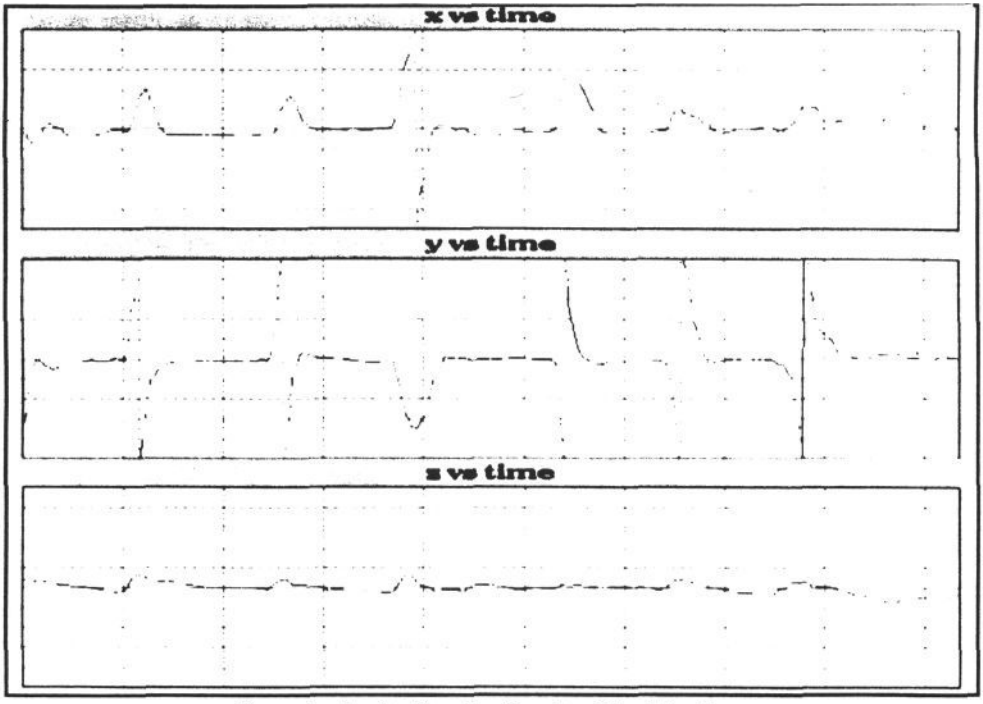

Figure 5. Graphs Showing Results of Tracking Tests

6 REFERENCES

Colchester, A C F *et al* (1994). VISLAN: Using Visible Landmarks for Intra-operative Computer Assisted Neurosurgery. *Joint Framework for Information Technology UK IT Forum* , Heriot-Watt University. pp. 275-284.

Evans, R J (1990). Kalman Filtering of Pose Estimates in Applications of the RAPiD Video Rate Tracker. *Proceedings of the BMVC 1990*, University of Oxford. pp. 79-83.

Evans, R J *et al* (1994). VISLAN: Combining Intra-Operative Video and Pre-operative Images for Surgical Guidance. *Proceedings of AAAI Symposium on Applications of Computer Vision in Medical Image Processing*, Stanford University. pp. 235-236.

Galloway R L *et al* (1992). Interactive Image Guided Neurosurgery. *IEEE Trans Biomed Enging* 39:1226-1231.

Harris, C G and Teeder, A (1993). Geometric Camera Calibration for Vision-based Navigation. *Proceedings of IFAC International Conference on Intelligent Autonomous Vehicles*, University of Southampton. Pergamon Press. pp. 77-82.

Harris, C G and Stennett, C (1990). RAPID - A Video Rate Object Tracker. *Proceedings of BMVC 1990*, University of Oxford. pp. 73-77.

Krybus W*et al* (1991). Navigation Support for Surgery by means of optical position detection. *Comput Assist. Radiology*. Lemke HU Rhodes ML Jaffee CC Felix R (eds), Springer-Verlag, Berlin p362.

Rheinhardt H F *et al* (1993). Sonic Stereometry in Microsurgical Procedures for Deep-Seated Brain Tumours and Vascular Malformations. *Neurosurg*. 32:51-57.

7 ACKNOWLEDGEMENTS

VISLAN is a collaborative project partly funded by the Department of Trade and Industry. It is being undertaken by image processing groups at Roke Manor Research Limited and the department of Neurology and Radiological Sciences at UMDS (Guy's Hospital), with the assistance of surgical teams at the National and Maudsley Hospitals.