Registration of Video Images to Tomographic Images by Optimising Mutual Information Using Texture Mapping

M. J. Clarkson, D. Rueckert, A. P. King, P. J. Edwards,
D. L. G. Hill, and D. J. Hawkes

Division of Radiological Sciences and Medical Engineering
The Guy's, King's and St. Thomas' Schools of Medicine and Dentistry
Guy's Hospital, London SE1 9RT, UK

Abstract. In this paper we propose a novel tracking method to update the pose of stereo video cameras with respect to a surface model derived from a 3D tomographic image. This has a number of applications in image guided interventions and therapy. Registration of 2D video images to the pre-operative 3D image provides a mapping between image and physical space and enables a perspective projection of the pre-operative data to be overlaid onto the video image. Assuming an initial registration can be achieved, we propose a method for updating the registration, which is based on image intensity and texture mapping. We performed five experiments on simulated, phantom and volunteer data and validated the algorithm against an accurate gold standard in all three cases. We measured the mean 3D error of our tracking algorithm to be 1.05 mm for the simulation and 1.89 mm for the volunteer data. Visually this corresponds to a good registration.

1 Introduction

We consider the general problem of relating pre-operative MR/CT data to an intra-operative scene for image guided interventions. Registration of 2D video images to the 3D pre-operative data provides a link between what is currently visible in the intra-operative scene and the information present in the 3D pre-operative data. The potential applications of such a registration method will be in image guided surgery in which video, endoscopy or microscopy images are used. Clinical applications include neurosurgery, ENT surgery, spinal surgery, image guided laproscopy and high precision radiotherapy. We have previously considered registering video images taken from an operating microscope to CT data, using video image intensity directly [3]. We propose a novel method of updating the registration over a period of time using a tracking algorithm that incorporates texture mapping and describe work in progress in applying the algorithm to clinical data.

A video image is a projection of the 3D scene onto a 2D imaging plane. This can be characterised by the pinhole camera model [6]. Given a point in the

C. Taylor, A. Colchester (Eds.): MICCAI'99, LNCS 1679, pp. 579–589, 1999.

3D scene, denoted in homogeneous coordinates by $\mathbf{x} = (x, y, z, 1)$ the aim is to find the 3×4 transformation matrix $\mathbf{G}$ that relates $\mathbf{x}$ to a point on the 2D image denoted by $\mathbf{u} = (u, v, 1)$

$$k\mathbf{u}^T = \mathbf{G}\,\mathbf{x}^T . \tag{1}$$

The matrix $\mathbf{G}$ represents a rigid body transformation from 3D scene coordinates to 3D camera coordinates, followed by a projective transformation onto the 2D image plane and k represents a scale factor for homogeneous coordinates. The rigid body transformation describes the pose (position and orientation) of the camera relative to the scene, and has six degrees of freedom. The projective transformation is determined by the internal characteristics of the camera. Assuming fixed zoom and focus, the camera can be calibrated so that the projective transformation is known. This reduces the registration problem to finding the six rigid body parameters which determine the camera pose.

1.1 Previous Work

The problem of estimating the pose of a camera with respect to an object is an essential step in many machine vision applications. Pose estimation can be achieved through corresponding pairs of 2D and 3D features, such as points [8] or points and lines [11]. The accuracy of the pose estimation algorithm is determined by the number of features, the accuracy with which the 2D and 3D coordinates of the features are known and the accuracy of their correspondence. Often many features are required for an accurate estimate of the pose. Registration of 3D medical images to 2D video images of a patient using anatomical point based methods is likely to be inaccurate. The landmark points are often difficult to localise accurately in the 3D image due to limited resolution and contrast. In addition, localisation of landmarks can be difficult in the 2D image as some landmarks may be hidden or prone to movement. This leads to poor registration accuracy.

Colchester *et al.* [4] have described a system in which a light pattern is projected onto the skin surface. Video images from a pair of calibrated cameras are used to reconstruct the illuminated surface, which is then registered to a surface model derived from the 3D image. Viola [15] demonstrated the use of an information theoretic framework to register a surface model to a video image of a skull phantom. A video image of an object will be related to a 3D model of the object by a geometric transformation mapping model points to image points, and an imaging function describing lighting conditions, surface properties, imaging device characteristics and so on. In general, reflectance is a function of lighting direction, surface normal and viewing direction. If the light source can be assumed to be far from the object, its rays will be parallel. In addition if the camera is assumed to be far from the object, then the viewing direction at each point will be constant. Thus the observed intensity will vary with the surface normal direction. The problem of pose estimation can then be formulated as maximising the mutual information between the video intensities and the surface normal vectors of the model. This does not assume a specific lighting model, it only assumes that a relationship between the video intensities and the surface normals exists.

We have previously demonstrated a registration algorithm based on maximising the mutual information between video and rendered image intensities alone. We have studied several information theoretic methods for utilising the information from multiple video and rendered images which showed that a mean 3D error of 1.05 mm using five video views can be achieved [3].

1.2 Objectives

We present a novel tracking algorithm, which is used to maintain registration between multiple video cameras and a 3D surface model, through the use of the video image intensities and texture (colour) mapping. In addition we use an accurate gold standard to validate the accuracy of our algorithm. We performed five tracking experiments on simulated, phantom and volunteer data and tested the tracking over a range of motion that might be encountered during, for example, a neurosurgical or ENT procedure without head immobilization.

2 Methods

2.1 Coordinate Transformation

The matrix $\mathbf{G}$ in equation (1) represents a transformation from 3D scene coordinates to 2D video image pixels. In this paper we use stereo pairs of cameras. Let $i = 1, 2$ denote the camera number. Using these two cameras we acquire or simulate a sequence of video images. Let $j = 1 \ldots N$ denote the video image number. The matrix $\mathbf{G}$ will be different for each camera and for each video image. Therefore let $\mathbf{G}_{ij}$ be the transformation from 3D scene coordinates to 2D video image pixels for camera i and for video image j. The matrix $\mathbf{G}_{ij}$ can be represented as

$$\mathbf{G}_{ij} = \mathbf{P}_i \, \mathbf{T}_{ij} \ . \tag{2}$$

First we perform tracking experiments using a plastic skull phantom. In this case the 3D coordinate system is defined by the CT coordinate system. $\mathbf{T}_{ij}$ is a transformation from 3D CT coordinates to 3D camera coordinates, and $\mathbf{P}_i$ is a transformation from 3D camera coordinates into 2D image pixels. In addition we perform experiments using a volunteer who has been fitted with a locking dental registration device as described in Edwards *et al.* [5]. In volunteer based experiments $\mathbf{T}_{ij}$ is a transformation from MR coordinates to the camera coordinate system. The matrix $\mathbf{P}_i$ is calculated using a calibration process and is fixed throughout the tracking process. Consider a sequence of N images where $j = 1 \ldots N$ taken from camera i. Assume that for both cameras in the system $i = 1, 2$, the initial registration of 3D image coordinates to the video image pixels is known. This means that for $j = 1$, $\mathbf{T}_{i1}$ is known. The goal of the tracking is to find the rigid body transformation which when combined with the initial known registration matrix $\mathbf{T}_{i1}$ and camera calibration matrix $\mathbf{P}_i$, transforms

3D image points onto the corresponding 2D video image pixels throughout a sequence of video images. The desired rigid body transformation is represented by $\mathbf{R}_j$ where

$$\hat{\mathbf{T}}_{ij} = \mathbf{T}_{i1}\, \mathbf{R}_j \ . \tag{3}$$

is the updated rigid body transform produced by our algorithm. The matrix $\mathbf{R}_j$ is determined by six parameters. These are t_x, t_y and t_z which represent translations with respect to the x, y and z axes respectively, and r_x, r_y and r_z which represent rotations about the x, y and z axes respectively. The matrix $\mathbf{R}_j$ is the output of the algorithm after each video frame, j, in the sequence. If the gold standard transformation $\mathbf{T}_{ij}$ is known then $\hat{\mathbf{T}}_{ij}$ should be approximately equal to $\mathbf{T}_{ij}$. Thus the tracking problem is to determine the six degrees of freedom t_x, t_y, t_z, r_x, r_y and r_z which updates the transformation from 3D model coordinates to 2D pixel coordinates for each video frame in a sequence.

2.2 Tracking without Texture Mapping

The problem of finding the correct alignment for each video frame can be seen as a problem of re-registration. To find the correct registration our algorithm produces a rendering of the surface model and compares it to the video image using mutual information. Given a video image with intensities $a \in A$ and a rendered image with intensities $b \in B$, we can calculate the mutual information of A and B denoted by $I(A; B)$ using

$$I(A; B) = H(A) + H(B) - H(A, B) \ . \tag{4}$$

where H denotes the Shannon entropy [12]. Registration of one video image to a 3D image is performed by maximising $I(A; B)$ by varying the six transformation parameters $t_x \ldots r_z$ and using a simple gradient ascent search strategy.

In some cases the registration of mono view video images can fail due to the symmetry of the surface or the lack of surface structure. Furthermore the registration of mono view video images is often poorly constrained along the optical axis of the video camera. We have previously shown that the use of stereo or multiple view video images can improve the accuracy, precision and robustness of the registration compared to the mono view case [3]. In our experiments images are always acquired in stereo pairs, and the transformation from one camera to the other is known. This means that we have two video images, denoted by A_1 and A_2, and we can produce the two corresponding rendered images, denoted by B_1 and B_2. Registration is performed by maximising an objective function which corresponds to the sum of the mutual information between each video image and the rendered image, i.e. $I(A_1; B_1) + I(A_2; B_2)$. As before the objective function is maximised using a simple gradient ascent search strategy.

2.3 Tracking with Texture Mapping

Texture mapping, or more specifically colour mapping [2] is a common computer graphics technique for adding realism to rendered images [6]. Each vertex in a surface model is given a 2D texture coordinate, which maps it to a position in a 2D texture image. The graphics pipeline then maps the texture image onto a surface patch of the 3D model. This can be seen in figure 1. The question arises as to how to map the video texture accurately to the surface model. As mentioned above, we start the tracking process with a known registration $\mathbf{G}_{i1} = \mathbf{P}_i\, \mathbf{T}_{i1}$. Using this equation, we take points in the surface model and project them onto 2D image pixels, and use these as the corresponding texture coordinates. The accuracy of the texture mapping is therefore determined by the accuracy of the initial registration.

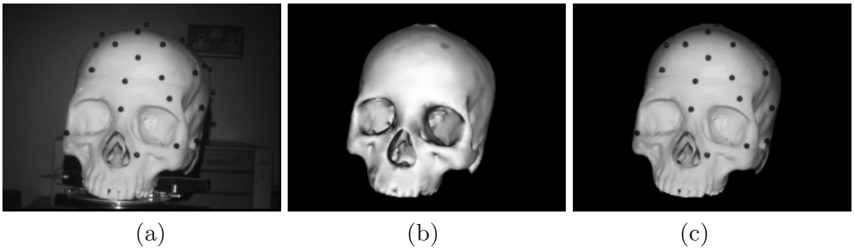


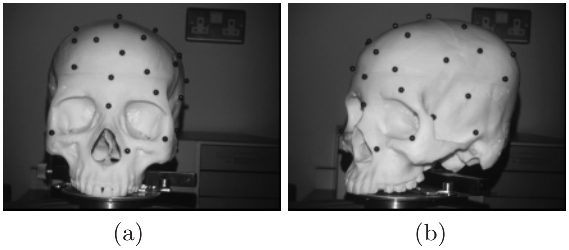
Fig. 1. Texture Mapping Example: (a) sample video image, (b) rendering of a registered model, (c) rendering of the same model with the video texture pasted onto the surface.

Without texture mapping we were matching a rendering of a surface with a video image. This assumes that the rendering looks fairly similar to the video image. It also assumes that only one surface is being imaged and that this surface has constant reflectance. The algorithm is trying to match the component of diffuse reflection in the video image intensities. However, with texture mapping we can associate information from the video image directly with the 3D model. This means we now have additional knowledge about the surface texture of the 3D object. The algorithm can proceed as before, by producing a rendering, where the rendered image intensities are a projection of the texture map intensities, and matching this to subsequent frames in a video sequence. This can be seen as a region matching algorithm, with mutual information measuring the similarity of a surface patch in the current image, with its known appearance in the previous image.

3 Experiment Design

After registration to each video frame j, the parameters $t_x \ldots r_z$ produce an estimate $\hat{\mathbf{T}}_{ij}$ of the gold standard matrix $\mathbf{T}_{ij}$. To assess the error in the registration we measured the projection error in mm. The projection error is the mean of the Euclidean distance between a 3D point and the closest point on a line projected through the corresponding 2D point. In addition we measured the 3D error in mm, which is the mean of the Euclidean distance between a 3D point multiplied by the gold standard matrix $\mathbf{T}_{ij}$, and the same 3D point multiplied by the estimated rigid body matrix $\hat{\mathbf{T}}_{ij}$ [3]. Furthermore we can measure the 3D distance between video frames by measuring the mean of the Euclidean distance between a 3D point, multiplied by the gold standard matrix $\mathbf{T}_{ij}$, and the same point multiplied by the gold standard matrix of the frame before $\mathbf{T}_{ij-1}$. We can measure the accumulative 3D distance as the sum of the 3D distance over each frame of the video sequence.

3.1 Tracking Simulation



(a) (b)

Fig. 2. Example images: (a) and (b) are the stereo pair used for the skull phantom experiments as described in section 3.1.

A CT scan (Philips TOMOSCAN SR 7000 0.488 × 0.488 × 1.0 mm, 512 × 512 × 142 voxels) of a plastic skull phantom was acquired. Our skull phantom has 23, 5 mm aluminium ball bearings which have been painted black attached to it. Two video images were taken of the skull phantom, shown in figure 2(a) and (b), and a 3D surface model was extracted from the CT scan using VTK [13]. The initial registration can be calculated using six or more pairs of corresponding 2D and 3D points. The aluminium ball bearings can be accurately localised in the 3D image using an intensity weighted centre of gravity operator [1], and interactively localised in the 2D images. We used Tsai's [14] camera calibration method to calculate the matrices $\mathbf{P}_i$ and $\mathbf{T}_{i1}$ for the two cameras $i = 1, 2$. The two camera views were separated by 45 degrees. The video image texture was mapped onto the surface model by projecting the surface model points onto the texture image. We generated 100 pairs of simulated images. This was accomplished by changing the pose of the surface model with respect to each camera view and producing a corresponding pair of texture mapped renderings, ie. one rendering for each view. The change of the model pose between each frame was a rotation of one degree. The sequence was 10 rotations to the left, 10 up, 20 right, 20 down, 20 left, 10 up and 10 right. We added zero mean, Gaussian noise ($\sigma = 7$) to these simulated images. The value of σ was chosen to simulate video image noise. We then performed a 'mono view' tracking experiment by taking the sequence of simulated images for camera $i = 1$ and using the known initial registration $\mathbf{T}_{i1}$ to initialise the tracking algorithm. The algorithm then attempted to recover the transformations $\mathbf{R}_j$ for $j = 2 \ldots 100$. We repeated this experiment, performing a 'stereo view' tracking experiment by taking the sequence of images for both cameras $i = 1, 2$, and using our algorithm to recover the transformations $\mathbf{R}_j$ for $j = 2 \ldots 100$.

3.2 Tracking a Skull Phantom

We subsequently took a series of real video images of the same skull phantom. The skull phantom was placed on a goniometer and a sequence of 21 images was taken, where the skull was rotated by 2 degrees clockwise between each image. The 3D surface model was registered to the initial view, using the above method and the algorithm used to recover the rotating motion.

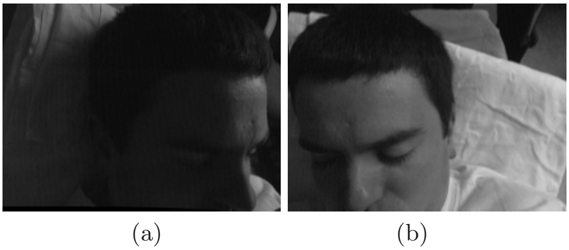

(a) (b)

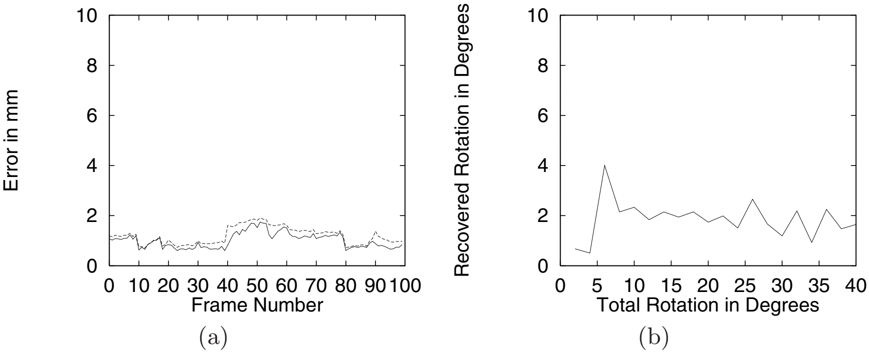
Fig. 3. Example images: (a) and (b) are the stereo pair used for volunteer experiments, as described in section 3.3.

3.3 Tracking a Volunteer

We then tested the algorithm on images of a volunteer. An MRI scan (1.016 × 1.016 × 1.250 mm, 256 × 256 × 150 voxels) was taken of the volunteer. This was corrected for scaling errors [9], and a skin surface extracted using VTK [13]. The volunteer was scanned whilst wearing a locking acrylic dental stent (LADS) [5]. A stereo pair of video cameras were fixed with respect to each other and calibrated using SVD [7], which produces the matrix $\mathbf{P}_i$ for each camera $i = 1, 2$ as mentioned in section 2.1. A bivariate polynomial deformation field for each camera was calculated to correct for distortion effects. The translational separation of the two cameras was approximately 30 centimetres and the disparity between their optical axes was approximately 45 degrees. Using the LADS [5] we calculate the gold standard transformation $\mathbf{T}_{ij}$ for each camera $i = 1, 2$ and for each image $j = 1 \ldots 25$. We then performed a 'mono view' tracking experiment by taking the sequence of simulated images for camera $i = 2$ and using the known initial registration $\mathbf{T}_{i1}$ to initialise the tracking algorithm. The algorithm then attempted to recover the transformations $\mathbf{R}_j$ for $j = 2 \ldots 25$. Subsequently we repeated this experiment, performing a 'stereo view' tracking experiment by taking the sequence of images for both cameras $i = 1, 2$, and using our algorithm to recover the transformations $\mathbf{R}_j$ for $j = 2 \ldots 25$.

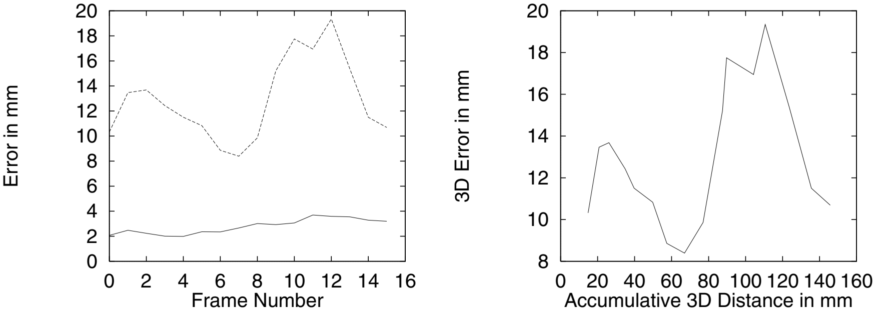
4 Results

Figure 4 (a) shows a graph of the results for the mono view simulation. The simulation did not include translations parallel to the optical axis of the camera so we would expect the mono tracking algorithm to work well. The mean projection error and mean 3D errors are 1.01 and 1.19 mm respectively for the 100 frames. Figure 4 (b) shows the results for the phantom tracking experiment. The sequence was a set of images, where the phantom had been rotated by 2 degrees between each video image. It can be seen that the tracking algorithm misses the first few frames, but then manages to recover approximately 2 degrees for the subsequent frames. The mean and standard deviation of the rotation estimates is 1.85 ± 0.76 mm. Figure 5 (a) shows the results for the mono view experiment on the volunteer. This graph shows that projection error and 3D error can be significantly different. Specifically the projection error can be reasonably low while the 3D error is high. A mono view experiment can fail to recover translations along the optical axis of the camera [3,10]. Figure 5 (b) shows the 3D error plotted against the accumulative 3D distance (the sum of 3D distance over each frame of the video sequence), which shows that the camera has moved 140mm. Figure 6(a) shows that with stereo views, the tracking performance is much better. An example pair of images is shown in figure 3. It can be seen that of the two images, one is significantly lower in contrast than the other. Figure 6(b) shows the 3D error as a function of accumulative 3D distance moved. Table 1 summarises the performance of the mono and stereo view algorithms. The simulation experiment performed well for both mono and stereo views with a mean 3D error of 1.19 and 1.05 mm respectively over the whole of the 100 frames sequence. For the volunteer tracking experiment, it can be seen that the stereo algorithm performs significantly better. However after 14 frames, corresponding to 140 mm of accumulative 3D movement, the stereo algorithm fails to track successfully.



(a) (b)

Fig. 4. 3D (dotted line) and projection (solid line) errors for (a) the mono view simulation and (b) the mono view phantom experiment. As described in sections 3.1 and 3.1 respectively.



(a) (b)

Fig. 5. (a) 3D error (dotted line) and projection error (solid line) for mono view, volunteer tracking experiment. (b) 3D error plotted against the accumulative 3D distance for the mono view volunteer tracking experiment. See section 3.3.

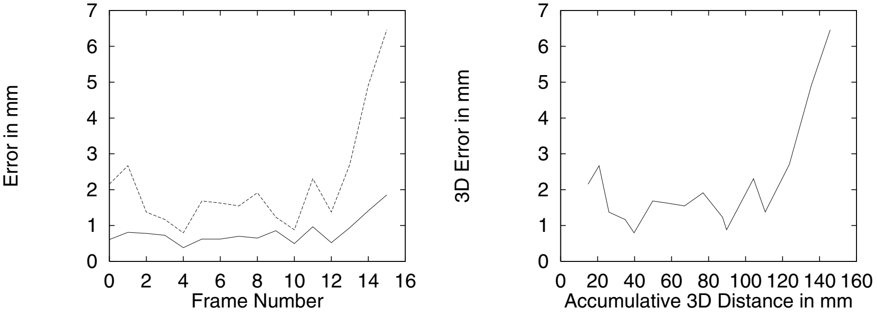
Table 1. A comparison of mono view and stereo view performance for (a) the simulation and (b) the volunteer experiments.

Case	Projection Error	3D Error
Mono	1.01	1.19
Stereo	0.83	1.05

(a)

Case	Projection Error	3D Error
Mono	2.75	13.03
Stereo	0.74	1.89

(b)

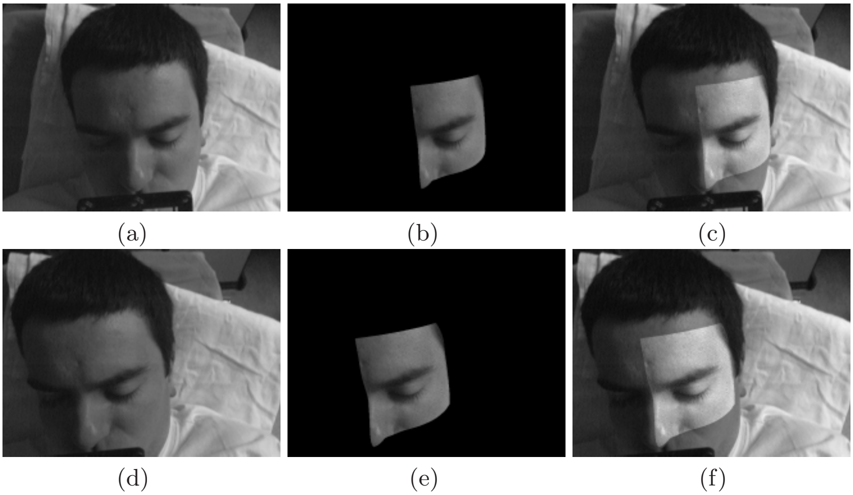


(a) (b)

Fig. 6. (a) 3D Error (dotted line) and Projection Error (solid line) for stereo view, volunteer tracking experiment. (b) 3D Error plotted against the Accumulated 3D Distance for the stereo view volunteer tracking experiment. See section 3.3

5 Conclusions

This paper has described a new tracking algorithm that uses texture mapping to register a pair of video images to a 3D surface model derived from MR/CT. We tested the algorithm with simulated data. This achieved registration with a mean 3D error of 1.05 mm for stereo views. This level of accuracy is largely determined by the step size of the gradient ascent algorithm, and so could be improved, but with increased computational cost. The mono tracking experiments with the phantom (section 3.1) and the volunteer (section 3.3) showed that tracking performance is poor if only one camera is used. However tracking was possible by using two camera views. We tested the tracking over a range of motion that might be encountered during for example a neurosurgical or ENT procedure without head immobilization. Future work will concentrate on improving robustness and assessing the effect of contrast, visible texture, surface area and surface curvature. In addition, this work uses a simple gradient ascent search method to maximise the mutual information. This could be improved by using predictive methods such as the Kalman filter.



(a) (b) (c)

(d) (e) (f)

Fig. 7. Results of volunteer tracking experiment: (a) Video image 1 (b) Texture mapped model (c) Model registered and overlaid on video image, at the initial pose, before tracking. (d) Video image 12 (e) Texture mapped model at the tracked pose (f) Model registered and overlaid on video image at the tracked pose.

Acknowledgements

MJC would like to thank the Engineering and Physical Sciences Research Council for funding his Ph.D. studentship, and also Jane Blackall, Graeme Penney and Daryl DeCuhna at the Computational Imaging Science Group, for their assistance.

References

1. C. B. Bose and I. Amir. Design of fiducials for accurate registration using machine vision. *IEEE Transactions on Pattern Analysis And Machine Intelligence*, 12(12):1196–1200, 1990. 194
2. E. Catmull. Computer display of curved surfaces. In *Proceedings of the Conference on Computer Graphics, Pattern Recognition and Data Structures*, pages 11–17, New York, May 1975. IEEE Computer Society. 192
3. M. J. Clarkson, D. Rueckert, D. L. G. Hill, and D. J. Hawkes. Registration of multiple video images to pre-operative CT for image guided surgery. In *Proceedings of SPIE Medical Imaging 1999 - in press*, 1999. 189, 191, 192, 194, 196
4. A. C. F. Colchester, J. Zhao, S. K. Holton-Tainter, C. J. Henri, N. Maitland, P. T. E. Roberts, C. G. Harris, and R. J. Evans. Development and preliminary evaluation of VISLAN, a surgical planning and guidance system using intra-operative video imaging. *Medical Image Analysis*, 1(1):73–90, 1996. 190
5. P. J. Edwards, A. P. King, D. J. Hawkes, O. Fleig, C. R. Maurer Jr., D. L. G. Hill, M. R. Fenlon, D. A. de Cunha, R. P. Gaston, S. Chandra, J. Mannss, A . J. Strong, M. J. Gleeson, and T. C. S. Cox. Stereo augmented reality in the surgical microscope. In *Proceedings MMVR.*, 1999. 191, 195
6. J. Foley, A. van Dam, S. Feiner, and J. Hughs. *Computer Graphics 2nd Edition*. Addison Wesley, 1990. 189, 192
7. R. C. Gonzalez and R. E. Woods. *Digital Image Processing*. Addison-Wesley Publishing Company, 1992. 195
8. R. B. Haralick, H. Joo, C-N. Lee, X. Zhuang, V. G. Vaidya, and M. B. Kim. Pose estimation from corresponding point data. *IEEE Transactions On Systems, Man, and Cybernetics*, 19(6):1426–1445, 1989. 190
9. D. L. G. Hill, C. R. Maurer Jr., C. Studholme, J. M. Fitzpatrick, and D. J. Hawkes. Correcting scaling errors in tomographic images using a nine degree of freedom registration algorithm. *Journal of Computer Assisted Tomography*, 22(2):317–323, 1998. 195
10. M. E. Leventon, W. M. Wells III, and W. E. L. Grimson. Multiple view 2D-3D Mutual Information registration. In *Image Understanding Workshop*, 1997. 196
11. T.Q. Phong, R. Horaud, and P. D. Tao. Object pose from 2D to 3D point and line correspondences. *International Journal Of Computer Vision*, 15:225–243, 1995. 190
12. F. M. Reza. *An Introduction To Information Theory*. McGraw Hill, 1961. 192
13. W. Schroeder, Martin K., B. Lorensen, L. Avila, R. Avila, and C. Law. *The Visualization Toolkit An Object-Oriented Approach to 3D Graphics*. Prentice-Hall, ISBN: 0-13-954694-4, 1997. 194, 195
14. R.Y. Tsai. A versatile camera calibration technique for high-accuracy 3D machine vision metrology using off-the-shelf TV cameras and lenses. *IEEE Journal Of Robotics and Automation*, RA-3(4):323–344, 1987. 194
15. P. Viola and W. M. Wells III. Alignment by Maximization of Mutual Information. *International Journal of Computer Vision*, 24(2):137–154, 1997. 190