LOCALISATION OF A MOBILE ROBOT USING A LASER SCANNER ON RECONSTRUCTED 3D MODELS[1]

João Gomes-Mota, Maria Isabel Ribeiro

Instituto Superior Técnico/Instituto de Sistemas e Robótica
Av.Rovisco Pais, P 1096 Lisboa CODEX, PORTUGAL
e-mail: (jsgm,mir)@isr.ist.utl.pt

Abstract: This paper presents a localisation algorithm for a mobile robot based on laser range and reflectance data measured with a Laser Range Scanner. It is suited to structured indoor environments, with horizontal floor. The environment models are available from a previous 3D reconstruction using the same laser device. The algorithm estimates the robot's posture without any initial estimate. Its core element is the frame object, a geometric entity that provides a precise localisation estimate when matched to another similar frame.

Keywords: Mobile Robot, Localisation, Laser Range Scanner, Reconstructed 3D Models.

1. INTRODUCTION

The RESOLV project's goal is to develop a general purpose tool to reconstruct large and complex indoor environments, rendering the representation available to users anywhere in the world. The system, termed EST (Environment Sensing for Telepresence) consists of a sensing head with a Laser Range Scanner and a video camera connected to a computer where the sensor control and data processing are done. To extend the project capabilities allowing automatic reconstruction and sensor displacement, the EST was mounted on top of a mobile platform which is required to exhibit an autonomous behaviour, thus reducing human local intervention and allowing remote control facilities. The mobile system, termed AEST (Autonomous EST) is shown on Fig. 1, with a sample view of a 3D reconstructed scene presented in (Sequeira *et al*, 1998). In the presence of a mobile platform, accurate localisation becomes necessary. Furthermore, accurate posture estimation improves the reconstruction quality, while reducing the time of execution of the environment reconstruction.

The project follows the Internet paradigm (RESOLV homepage: www.hhdc.bicc.com/resolv/), thus the reconstructed environment model is described in VRML format, all the reconstruction procedure is made on board, whilst it is controlled through the Internet with a browser-like tool.

The operation cycle begins with the localisation of the mobile robot respective to the map reference and proceeds with the 3D range acquisition with the laser scanner followed by the surface texture's acquisition with the video camera. Then, a new acquisition point is computed in order to enhance the current reconstructed model and to resolve occlusions, the robot travels towards the new goal, driven by an autonomous navigation tool developed by (Castro *et al, 1998).* When it reaches the goal point, the cycle repeats.

Fig. 1. AEST system and output

[1] This work was supported by Project RESOLV - Reconstruction using Scanned Laser and Video of the ACTS programme, EU.

The proposed localisation algorithm is built upon a geometric object, termed a *frame*. When the *frames* from the laser data and from the 3D reconstructed maps are created, , each *frame* from one set is compared to all similar *frames* on the other set. If a match occurs, a candidate solution to the posture estimation problem results. The combination of all possible solutions generates a cloud of postures, which, after clustering, reduce to a maximum likelihood solution.

The paper is organised as follows. Section 2 summarises the input data procedures and the main parts of the algorithm. Section 3 begins with the *frame* motivation and proceeds to the *frame* concept and its operations. Section 4 shows several stages of an experimental case and Section 5 contains the conclusions and some comments.

2. ALGORITHM DESCRIPTION

The localisation algorithm was developed within the work frame of (Sequeira 1996a, 1996b and 1996c), which provides the 3D reconstructed models of the environment, and the acquisition tools: the Laser Range Scanner. The proposed localisation solution is optimised to this sensor and to the 3D models.

The Laser Range Scanner provides very accurate range measurements, from which 3D environment models are created, at the expense of a long acquisition time. A 2D laser profile, scanned likewise, keeps the accuracy while minimising the acquisition time, since only one horizontal line is taken. Since the localisation is bounded to the 2D planar space (x, y and orientation) a 2D horizontal profile ought to be adequate.

In order to associate the 3D reconstructed model to the 2D range data, a similar 2D profile must be extracted from the 3D model which is represented by a list of surface descriptions, currently planes and biquadratic surfaces (Fig. 2a). To create a 2D profile from the 3D reconstructed model, the 3D map is intersected by an horizontal plane at the same height of the sensor head, resulting in a list of parametrised lines and quadratic curves (Fig. 2b), denoted onwards the map line list.

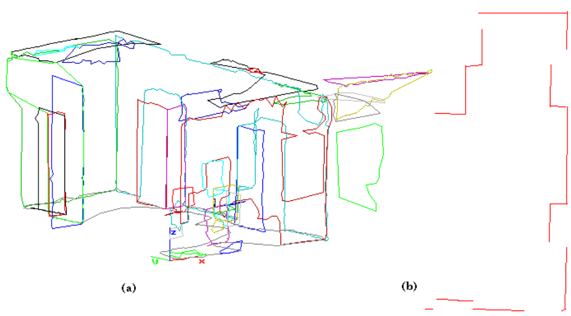

Fig. 2 (a) Surface model; (b) extracted map profile

On the range data side, the lines and quadratic curves are extracted from the range data (Figure 3a) by means of one of several statistical methods, namely the most popular Least Square Estimation, e.*g*. Press *et al*. This method was chosen because it may be implemented iteratively, enhancing the efficiency of the line extraction procedure. The result is the scan line list (Fig. 3b).

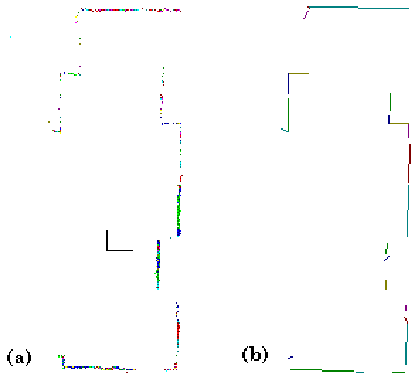

Fig. 3 (a) Raw range data; (b) extracted lines

Currently, the localisation algorithm uses only lines, since they proved sufficient and including the biquadratic curves would add an unnecessary computational burden along with some noise. The two line lists share the same parameter notation and, after sorting by decreasing size, constitute the start-up data for the frame algorithm.

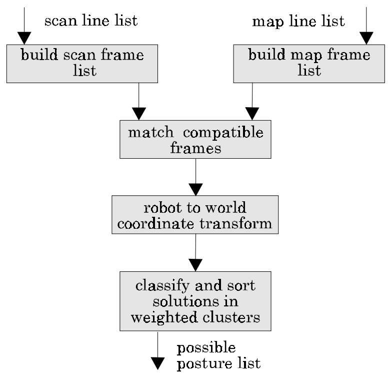

Fig. 4. Frame algorithm steps

In Fig. 4 the algorithm flow is described. It begins with the two line lists, used to create the *frame* list associated to each data source. Then, each element of the scan list is matched to all possible elements of the map list. When a match occurs, the coordinate transform between scan *frame* and map *frame* is computed and the resulting posture is stored.

The final posture estimate is obtained through a weighted clustering procedure of all postures, aggregating all points into a few clusters. The existence of several solutions is due to symmetries. However, if a sufficient number of *frames* is used, the wrong solutions have residual weight.

3. THE FRAME CONCEPT AND ITS OPERATIONS

The lines computed from the range data are similar to the lines extracted from the 3D models, except that most of the scan lines are shorter, as a consequence of partial surface occlusion in non-convex spaces and scan angle hardware limits.

In the absence of any *a priori* estimate on the mobile posture, one must concede that any line in the scan list could correspond to any line in the map list, as long as the map line is of equal or greater length. If the two lines correspond, the possible posture loci is a straight line in the xy-space with orientation determined by the relative position of the scanned line to the mobile (Fig. 5). In fact, due to line symmetry, there is a "mirror" solution if the orientation is rotated π radians.

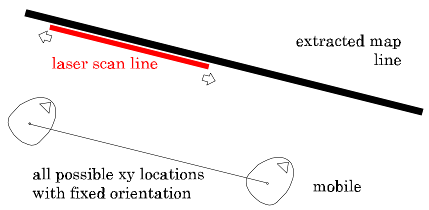

Fig. 5. Matching a scan line to a larger map line

To disambiguate the xy position, a new pair of lines is considered: if the two pair of lines are concurrent, the intersection of the possible loci due to each pair is a single point:

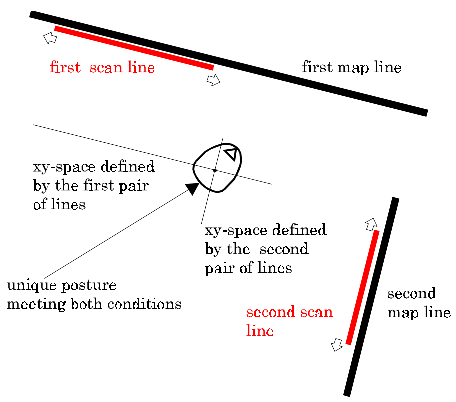

Fig. 6. Localisation using two concurrent lines

The *frame* concept is a formalised version of this reasoning. A *frame* is a geometric object formed by two concurrent lines. Its elements are the origin - the point where the lines or their extension cross - and two axis - the two lines.

The *frame* is defined by five local parameters and three global parameters. The local data is used to compare *frames* and is independent of the coordinate system, while the global data is used to compute the coordinate transform between map frames and scan frames. The local parameters are: the internal angle between axis, **α**, and the distance from the origin to the start and end points of both axis: $\mathbf{start_1}$, $\mathbf{start_2}$, $\mathbf{end_1}$ and $\mathbf{end_2}$ (Fig. 7). The global parameters are the posture values, **x**, **y**, **γ** (Fig. 8). The map frames global data is expressed in world coordinates while the scan frames global data is defined relative to the robot.

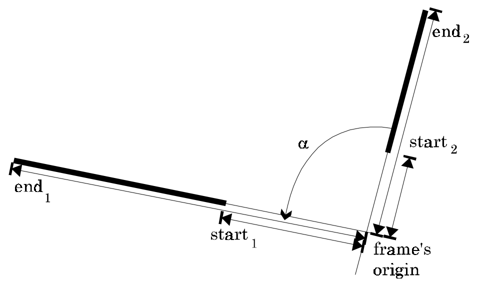

Fig. 7. Local frame parameters

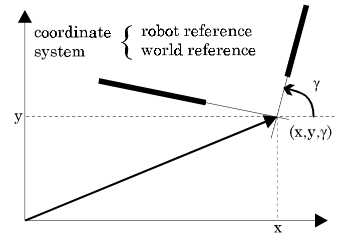

Fig. 8. Global frame parameters

Once the two lists of frames are fully defined, the frame match procedure begins. To test whether two frames match, their local parameters are compared; first the inner angle must be nearly equal (1), otherwise the test ends; then both limits of the scan frames axis' must lie within the limits of the map frames axis', (2) and (3), with a small degree of tolerance. It should be noticed that the tests must account for symmetries in case the axis are reversed, i.e., if axis 1 on the scan frame corresponds to axis 2 on the map frame. Therefore, the tests must be done for both cases. For the sake of clarity only the direct case will be described. The test equations are (see also Fig. 9),

$$\alpha^{Map} - \delta_{ang} \le \alpha^{Scan} \le \alpha^{Map} + \delta_{ang} \quad (1)$$

$$start_1^{Map} - \delta_m \le start_1^{Scan} < end_1^{Scan} \le end_1^{Map} + \delta_m \quad (2)$$

$$start_2^{Map} - \delta_m \le start_2^{Scan} < end_2^{Scan} \le end_2^{Map} + \delta_m \quad (3)$$

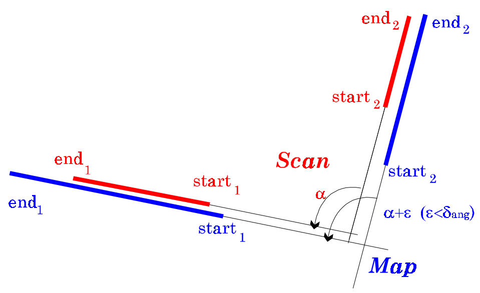

Fig. 9. Frame match

The **δ**-parameters are thresholds which may be tuned to meet the noise level in the scan data and the degree of symmetry of the reconstructed map. In case a frame match is detected, the three global values are used to compute the transform vector relating the two references, the scan data in robot coordinates and the map data in world (inertial) coordinates.

The rotation angle, **θ**, expresses as

$$\theta = \gamma_1^{Scan} - \gamma_1^{Map} \tag{4}$$

and the reference translation, **Δx** and **Δy,** results from the canonical transform equation:

$$\begin{bmatrix} x \\ y \\ 1 \end{bmatrix}_{Map} = \begin{bmatrix} \cos\theta & -\sin\theta & \Delta x \\ \sin\theta & \cos\theta & \Delta y \\ 0 & 0 & 1 \end{bmatrix} . \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}_{Scan} \tag{5}$$

The resulting vector (**Δx**, **Δy**, **θ**) is the mobile posture in the world reference; it is added to the candidate posture list. The final step is the clustering procedure to group the cloud of postures.

The cluster algorithm is based on fixed cluster radius, variable number of clusters and weighted samples. These options follow the statistic characteristics of the frame matching, resulting from the statistics of the Laser Range Scanner: the range measurements have a small spread, leading to accurate defined lines and precise matching with the map frames. The result is a 3D space (**Δx**, **Δy**, **θ**), where dense distinct clouds are separated by large empty areas. In spite of the precise matching, weights are accorded to the various results to discriminate the different levels of confidence granted to each frame match.

The weighting criterion was deduced from the actual range data analysis. The scan frames with comparable axis produce more accurate results, because a small axis is more sensitive to errors, the origin embeds a larger error, specially if the axis are far away from the origin (if **start$_{1,2}$** is large); small axis are also more prone to angle errors because of the reduced number of points used in the line generation. This reasoning led to a weighting criterion based on the product of the two axis' sizes. The map frames are not used in the weighting criterion since they are regarded as the exact map representation.

Each element on the 3D solution space, $\mathbf{p_i}$ (i=1,...,N), and the cluster points, $\mathbf{c_j}$ (j=1,...,C), are defined likewise,

$$p_i = \begin{bmatrix} \Delta x_i & \Delta y_i & \theta_i & w_i \end{bmatrix} \quad \text{where}$$
$$w_i = (end_1 - start_1) \cdot (end_2 - start_2) \tag{6}$$

$$c_j = \begin{bmatrix} x_j^{cl} & y_j^{cl} & \theta_j^{cl} & w_j^{cl} \end{bmatrix} \tag{7}$$

The cluster generation is very simple, at the expense of some generality. It runs as follows:

i. The first point, $\mathbf{p_1}$, is the seed of the first cluster, $\mathbf{c_1}$.
ii. If (8) holds, $\mathbf{p_i}$ is added to the cluster $\mathbf{c_j}$ according to (9) and the cluster weight is updated (10). Otherwise, $\mathbf{p_i}$ initiates a new cluster, $\mathbf{c_{j+1}}$.
iii. The loop ends when all points are tested.

$$\begin{gathered} \left|\theta_i - \theta_{cj}\right| < \delta_{angle} \\ \sqrt{(x_i - x_{cj})^2 + (y_i - y_{cj})^2} < \delta_{cent} \end{gathered} \tag{8}$$

$$\begin{cases} x_j^{cl} = \dfrac{w_j^{cl}.x_j^{cl} + w_i.\Delta x_i}{w_j^{cl} + w_i} \\ y_j^{cl} = \dfrac{w_j^{cl}.y_j^{cl} + w_i.\Delta y_i}{w_j^{cl} + w_i} \\ \theta_j^{cl} = \dfrac{w_j^{cl}.\theta_j^{cl} + w_i.\theta_i}{w_j^{cl} + w_i} \end{cases} \tag{9}$$

$$w_{cj} = w_{cj} + w_i \tag{10}$$

The result of the clustering process is a set of C clusters in the form of (7). The cluster list is then sorted by decreasing weight and the clusters with less than four points are discarded, eliminating the majority of wrong solutions generated by environment and algorithm symmetries. The result is the **possible posture list**, concluding the frame localisation algorithm.

Usually, the first element of the list concentrates more than 80% of the total weight. However, in some rare cases the weights may be distributed more evenly, creating a decision problem, which has been solved with a likelihood evaluation (see Section 4).

The major drawback of the frame localisation algorithm is its exponential nature. However, it is possible to cope with it, by effectively sorting and trimming the data objects. Also, since the objects used are stored in lists, the algorithm is inherently iterative; if necessary, it is possible to extend the database by simply adding new objects (lines or frames) and process the new data while preserving the current results.

To begin with an efficiency optimisation, the small and/or ill defined lines are discarded (small lines are extracted from a few points, incorporating larger errors) and the remaining ones are sorted. This measure prevents the propagation to the frame generation of more than half of the lines, in the average. In case more frames are needed, new, smaller lines can be used to generate them. Due to square corners, the lines are usually grouped into two orthogonal directions, thus N lines produce only $N^2/4$ frames instead of N^2 frames, as it would be expectable.

To reduce the number of frames and frame tests further, the frame list also comprises some trimming techniques: all frames must have at least one "long" axis. This eliminates small frames which match with many other frames and are usually associated with larger errors. It also helps reducing the number of tests. If the frames are sorted by decreasing order of their longer axis, the match loop with one constant frame ends as soon as the other candidate's longer axis is shorter.

4. EXPERIMENTAL RESULTS

To illustrate the frame algorithm, an example is discussed. The office room area shown in all images is approximately 11 meter long by 5 meter wide. The 3D model is shown in Fig. 2a; it is fairly incomplete, in particular in the area represented at the lower left in Figures 10 to 12.

In Fig. 10a, the scan line list is shown. The possible posture list has five elements, used to sketch the mobile robot over the map lines (Fig. 10b). Fig. 11a represents the correct posture, chosen by its weight (82% of the total weight), along with map lines and the scan lines subset used to determine it. In a similar representation, Fig. 11b shows a wrong solution, highlighting the coincident lines which generate this odd result.

To solve this example only 15 frames are required. Incidentally, the scan profiles for localisation were taken several months after the generation of the re-constructed 3D model, underlining the robustness of the algorithm to minor model changes.

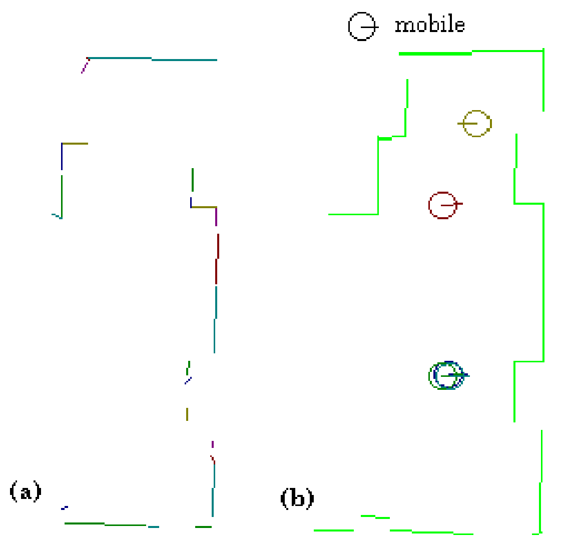

Fig. 10 (a) Scan line list; (b) possible posture list on the extracted map list

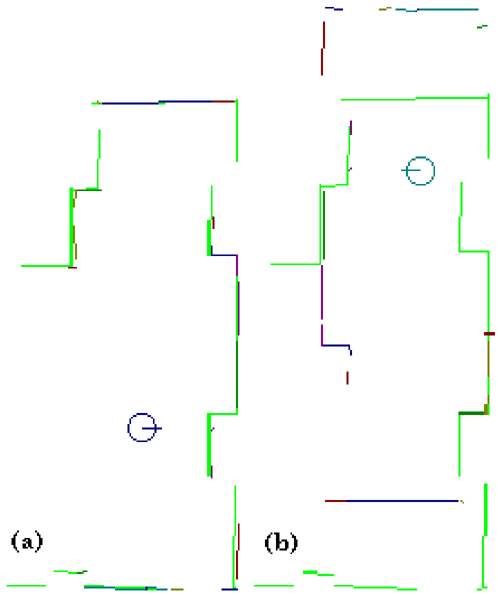

Fig. 11 (a) True solution; (b) false solution due to symmetry

A likelihood evaluation using range data and map lines was developed to measure the algorithm accuracy. It is not a part of the frame algorithm, but a useful, almost necessary, extension.

The likelihood evaluation for two postures on the possible posture list is presented in Fig. 12 together with the mobile robot and the range data drawn over the map lines. Although the postures are less than 0.1m apart (see also Fig. 10b), solution (a) is far more accurate than solution (b).

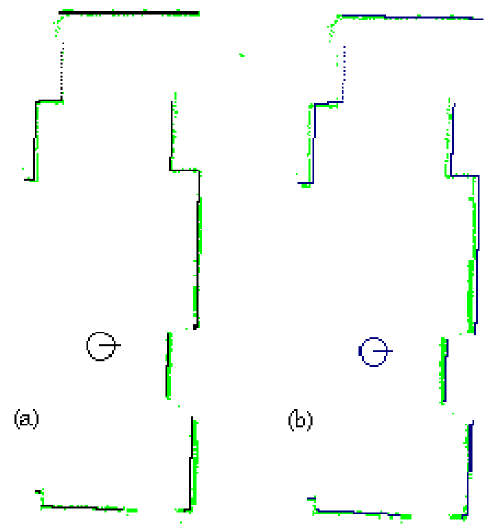

Fig. 12. Likelihood test of two solutions using range data and map lines

The likelihood evaluation uses the distance from each point in the range scan data to the map lines. A distance histogram is computed and the likelihood measure is defined as the fraction of scan points closer to the map lines than 0.05 m.

The table shows the two posture coordinates and the associated likelihood measure:

	Posture (a)	Posture (b)
x^{cl} [m]	2.296	2.217
y^{cl} [m]	2.381	2.366
θ^{cl} [rad]	0.02164	0.01920
Likelihood [<0.05m]	84.5%	47.5%

The likelihood evaluation is also used to discard the wrong solutions present in the possible posture list, solving any ambiguous cases.

5. Conclusions and Comments

The major achievement of this algorithm is to accurately localise the mobile without any *a priori* knowledge of its posture. It proved most adequate to the RESOLV applications, where the goal is to map an environment, because it can cope with large unknown areas, keeping its "anchor" on the available reconstructed 3D model; its constraints are met by most indoor environments: flat ground and vertical surfaces.

The line extraction - not described - is the only computational intense procedure. Afterwards, all data are comprehensively embedded in geometric objects, characterised by few parameters. The match operations reduce to simple arithmetic and tests; the cluster procedure is also very simple and is performed in one loop.

Unfortunately, problems may occur if the environment has a high degree of symmetry, for instance, in a long corridor with evenly spaced doors or in a cubic or cylindrical room. In such cases, the algorithm needs an asymmetric portion of the environment to disambiguate between the periodic solutions - just like humans.

A second, more subtle, difficulty arises from the akin nature of the mapping and localisation algorithm: they use the same sensor. Thus, when the reconstruction fails, the localisation may also fail. This happens most often with mirrors and windows and with dark, soft or highly textured surfaces. The problems with glassed surfaces have been minimised with the reflectance data. The latter cases require a new sensor, because such surfaces return very weak signals, if not, no signal at all. To overcome these local problems the 3D model must contain some reliable surfaces where the algorithm anchors.

The Frame Algorithm has been embedded in a multi-layer algorithm to increase the accuracy. It is now the first stage of an algorithm with 0.02 m and 0.003 rad maximum error.

REFERENCES

Castro, J., Vítor Santos, M.Isabel Ribeiro, *A Multi-Loop Robust Navigation Architecture for Mobile Robots*. In Proceedings ICRA'98 - IEEE Int. Conference on Robotics and Automation, Leuven, Belgium.

Press, Teukolsky, Vetterling, Flannery, *Numerical Recipes in C*, Chapter 15, Cambridge University Press, USA.

Sequeira, V. (1996a), J.G.M.Gonçalves, M.I.Ribeiro, *Active View Selection for Efficient 3D Scene Reconstruction*, Proceedings ICPR'96 - 13th Int. Conference on Pattern Recognition, pp. 815-819. Vienna, Austria,

Sequeira, V. (1996b), J.G.M.Gonçalves, M.I.Ribeiro, *3D Reconstruction of Indoor Environments*, Proceedings ICIP'96 - Int. Conference on Imaging Processing, pp.405-408. Lausanne, Switzerland.

Sequeira, V. (1996c), *Active Range Sensing for Three Dimensional Environment Reconstruction*, *Ph.D. Thesis*, Instituto Superior Técnico, Technical University of Lisbon, Lisbon, Portugal.

Sequeira, V., K. C.Ng, S. Butterfield, J.G.M. Gonçalves, D.C. Hogg (1998), *Three-dimensional textured models of indoor scenes from composite range and video images*, In R.N. Ellson and J.H. Nurre, editors, Proceedings SPIE, Three-Dimensional Image Capture and Applications, vol. 3313, To appear 1998.