[29] TINA: A 3D Vision System for Pick and Place

John Porrill, Stephen B Pollard, Tony P Pridmore, Jonathan B Bowen,
John E W Mayhew and John P Frisby

AI Vision Research Unit
University of Sheffield, Sheffield S10 2TN, UK

Reprinted, with permission of Butterworth Scientific Ltd, from *Image and Vision Computing*, 1988, **6**, 91-99.

Abstract

This paper provides an overview of the Sheffield AIVRU 3D vision system for robotics. The system currently supports model based object recognition and location; its potential for robotics applications is demonstrated by its guidance of a UMI robot arm in a pick and place task. The system comprises:

1) The recovery of a sparse depth map using edge based passive stereo triangulation.
2) The grouping, description and segmentation of edge segments to recover a 3D description of the scene geometry in terms of straight lines and circular arcs.
3) The statistical combination of 3D descriptions for the purpose of object model creation from multiple stereo views, and the propagation of constraints for within view refinement.
4) The matching of 3D wireframe models to 3D scene descriptions, to recover an initial estimate of their position and orientation.

The system is currently being developed to allow robot navigation by utilising visual feedback. The idea is to exploit the temporal coherence that exist in a sequence of images in order to provide quickening strategies.

1. Introduction.

The following is a brief description of the system. Edge based binocular stereo is used to recover a depth map of the scene from which a geometrical description comprising straight lines and circular arcs is computed. Scene to scene matching and statistical combination allows multiple stereo views to be combined into more complete scene descriptions with obvious application to autonomous navigation and path planning. Here we show how a number of views of an object can be integrated to form a useful visual model, which may subsequently be used to identify the object in a cluttered scene. The resulting position and attitude information is used to guide the robot arm.

Figure 1 illustrate our system at work. A pair of Panasonic WV-CD50 CCD cameras are mounted on an adjustable stereo rig. Here they are positioned with optical centers approximately 15cm apart with asymmetric convergent gaze of approximately 16 degrees verged upon a robot workspace some 50cm distant. The 28mm Olympus lens (with effective focal length of approximately 18.5mm) subtends a visual angle of about 27 degrees. The system is able to identify and accurately locate a modelled object in the cluttered scene. This information is used to compute a grasp plan for the known object (which is precompiled with respect to one corner of the object which acts as its coordinate frame). The UMI robot which is at a predetermined position with respect to the viewer centered coordinates of the visual system is able to pick up the object.

The system is a continuing research project: the scene description is currently being augmented with surface geometry and topological information. We are also exploring the use of predictive feed forward to quicken the stereo algorithm. The remainder of the paper will describe the modules comprising the system in more detail.

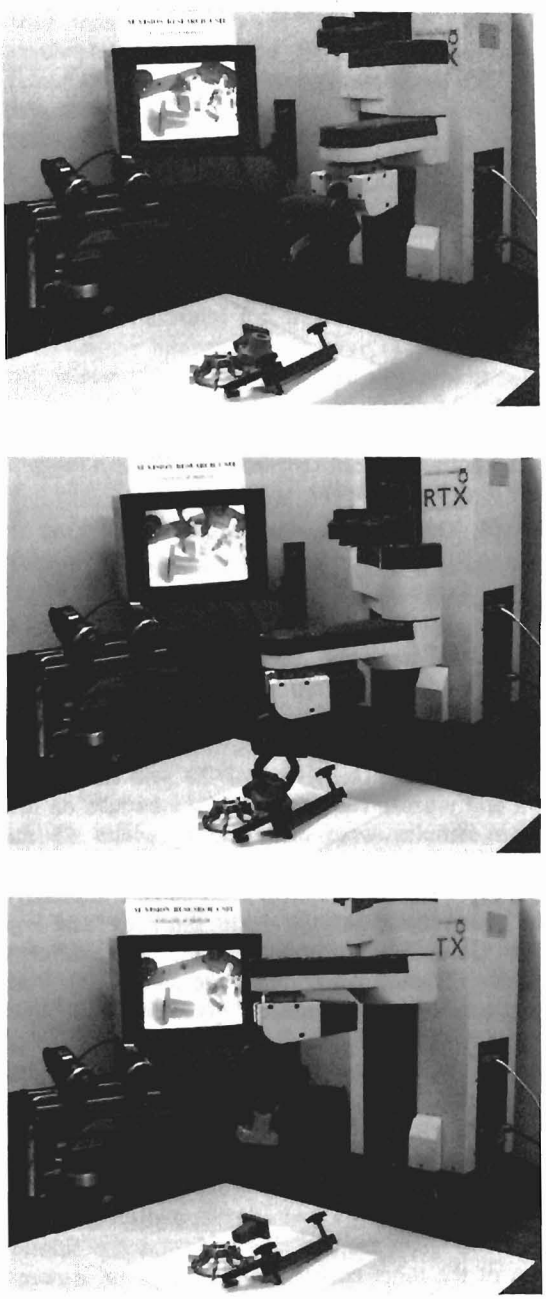

Figure 1. A visually guided robot arm.

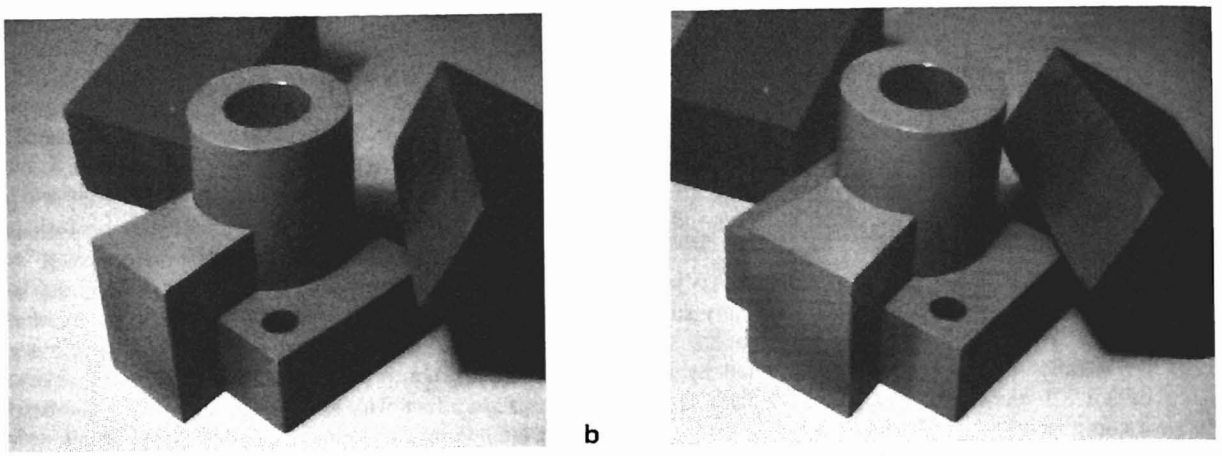

Figure 2. **a, b,** *Stereo images*

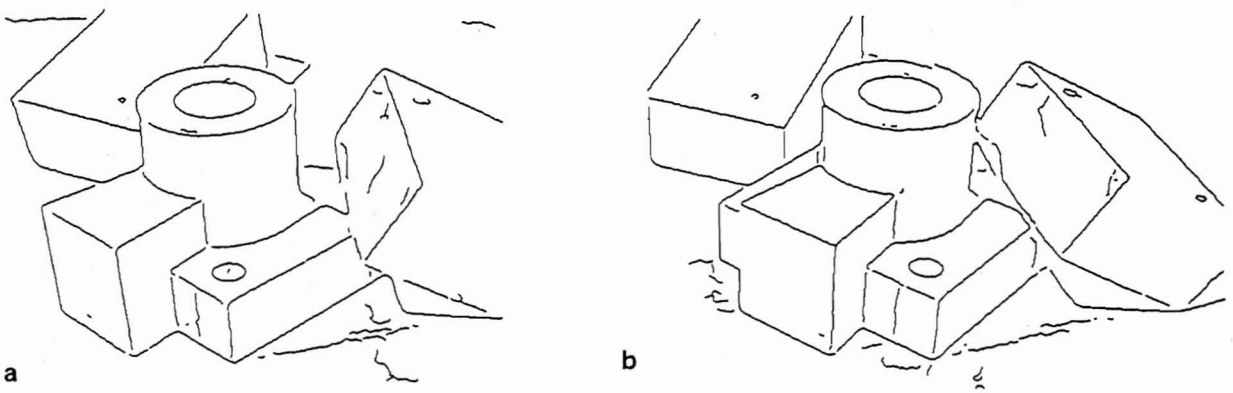

Figure 3. **a, b,** *Edge maps*

2. PMF: The recovery of a depth map.

The basis is a fairly complete implementation the Canny edge operator[1] applied to two images obtained from the CCD cameras (Figure 2). The images are 256x256 with 8 bit grey level resolution. In the camera calibration stage, a planar tile containing 16 squares equally spaced in a square grid was accurately placed in the workspace at a position specified with respect to the robot coordinate system such that the orientation of the grid corresponded to the x and y axes. The position of the corners on the calibration stimulus were measured to within 15 microns using a Steko 1818 stereo comparator. Tsai's calibration method was used to calibrate each camera separately. We have found errors of the same order as Tsai reported which are sufficiently small for the purposes of stereo matching. The camera attitudes are used to transform the edge data into parallel camera geometry to facilitate the stereo matching process. To recover the world to camera transform the calibration images are themselves used as input to the system, ie are stereoscopically fused and the geometrical description of the edges and vertices of the squares statistically combined. The best fitting plane, the directions of the orientations of the lines of the grid corresponding to the x and y axes, and the point of their intersection gives the direction cosines and position of the origin of the robot coordinate system in the camera coordinate system. The use of the geometrical descriptions recovered from stereo as feedback to iterate over the estimates of the camera parameters is a project for the future.

Currently a single scale Canny operator with $\sigma = 1$ pixel is used (Figure 3). The non maxima suppression which employs quadratic interpolation gives a resolution of 0.1 of a pixel (though dependent to some extent upon the structure of the image). After thresholding with hysteresis (currently non adaptive), the edge segments are rectified (Figure 4) so as to present parallel camera geometry to the stereo matching process. This also changes the location of the centre of the image appropriately, allows for the aspect ratio of the CCD array (fixing the vertical and stretching the horizontal) and adjusts the focal lengths to be consistent between views.

Camera calibration returns good estimates of the optical centre, focal length and orientation of each camera. This *original* geometry is illustrated in Figure 4 behind the interocular axis O_lO_r. The point at which the principal axis of the camera intersects the image plane is denoted P_l and P_r for the left and right hand cameras respectively. Not that (i) the principal axes need not meet in space (though it is advantageous if they almost do), and (ii) the focal lengths are not necessarily equal. It is desirable to construct an equivalent parallel camera geometry. For convenience this is based upon the left camera; the principal axis of the imaginary left camera $O_l\vec{Q}_l$ is chosen to be of focal length F, perpendicular to $O_l\vec{O}_r$, and to be coplanar with $O_rO_lP_l$ (as is the x axis of the image plane). An identical imaginary camera geometry is constructed for the right camera (ie $O_l\vec{Q}_l$ and $O_r\vec{Q}_r$ are parallel). Note that $O_r\vec{Q}_r$ need not be coplanar with $O_lO_rP_r$. For pictorial simplicity the new coordinate frames are shown in front of

the interocular axis. Points on the original image planes can now be projected through the optical centres of each camera onto the new and imaginary image planes. With the result that corresponding image points will appear on corresponding *virtual* rasters. For the sake of economy and to avoid aliasing problems this transformation is applied to edge points rather than image pixels themselves.

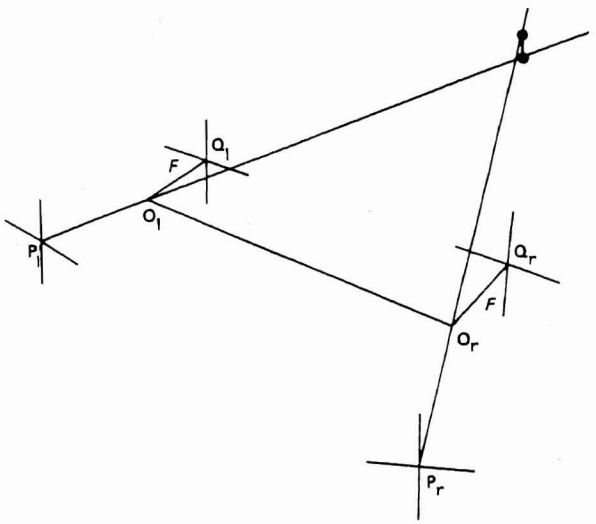

Figure 4. Parallel Camera Geometry.

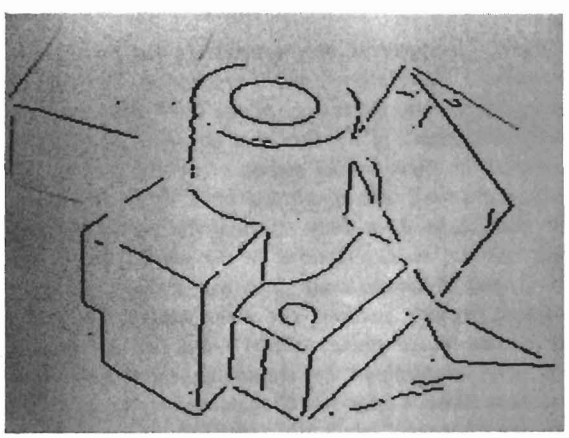

Figure 5. The depth map, displayed with respect to the left image, with disparities coded by intensity (near-dark far-light). The total range of disparities in the scene was approximately 35 pixels from a search window of 200 pixels. PMF is a neighbourhood support algorithm and in this case the neighbourhood was 10 pixels radius. The limiting disparity gradient employed in PMF was 0.5. The iteration strategy used a conservative heuristic for the identification of correct matches, and their scores were frozen. This effectively removes them from succeeding iterations and reduces the computational cost of the algorithm as it converges to the solution. 5 iterations were sufficient.

The two edge maps are and stereoscopically combined to form a depth map (Figure 5). The PMF[2, 3] stereo algorithm uses the disparity gradient constraint to solve the stereo correspondence problem. The parallel camera geometry allows potential matches to be restricted to corresponding rasters. Initial matches are further restricted to edge segments of the same contrast polarity and of roughly similar orientations (determined by the choice of a disparity gradient limit). Matches for a neighbouring point may support a candidate match provided the disparity gradient between the two does not exceed a particular threshold. Essentially, the strategy is for each point to choose from among its candidate matches the one best supported by its neighbours.

The disparity gradient limit provides a parameter for controlling the disambiguating power of the algorithm. The theoretical maximum disparity gradient is 2.0 (along the epipolars), but at such a value the disambiguating power of the constraint is negligible. False matches frequently receive as much support as their correct counterparts. However, as the limit is reduced the effectiveness of the algorithm increases and below 1.0 (a value proposed as the psychophysical maximum disparity gradient by Burt and Julesz[4]), we typically find that more than 90% of the matches are assigned correctly on a single pass of the algorithm. The reduction of the threshold to a value below the theoretical limit has little overhead in reduction of the complexity of the surfaces that can be fused until it is reduced close to the other end of the scale (a disparity gradient of 0.0 corresponds to fronto-parallel surfaces). In fact we find that a threshold disparity gradient of 0.5 is very powerful constraint for which less than 7% of surfaces (assuming uniform distribution over the Gaussian sphere: following Arnold and Binford[5]) project with a maximum disparity gradient greater than 0.5 when the viewing distance is four times the interocular distance. With greater viewing distances, the proportion is even lower.

It has been shown[6,7] that enforcing a disparity gradient ensures Lipschitz continuity on the disparity map. Such continuity is more general than and subsumes the more usual use of continuity assumptions in stereo.

The method used to calibrate the stereo cameras was based on that described by Tsai[8] (using a single plane calibration target) which recovers the six extrinsic parameters (3 translation and 3 rotation) and the focal length of each camera. This method has the advantage that all except the latter are measured in a fashion that is independent of any radial lens distortion that may be present. The image origin, and aspect ratios of each camera had been recovered previously. The calibration target which was a tile of accurately measured black squares on a white background was positioned at a known location in the XY plane of the robot work space. After both cameras have been calibrated their relative geometry is calculated.

Whilst camera calibration provides the transformation from the viewer/camera to the world/robot coordinate spaces we have found it more accurate to recover the position of the world coordinate frame directly. Stereo matching of the calibration stimulus allows its position in space to be determined. A geometrical description of the position and orientation of the calibration target is obtained by statistically combining the stereo geometry of the edge descriptions and vertices[9].

3. GDB: The recovery of the geometric descriptive base.

In this section we briefly report the methods for segmenting and describing the edge based depth map to recover the 3D geometry of the scene in terms of straight lines and circular arcs. A complete description of the process can be found in Pridmore *et al*[10] and Porrill *et al*[11].

The core process is an algorithm (GDF) which recursively attempts to describe, then smooth and segment, linked edge segments recovered from the stereo depth map. GDF is handed a list of edge elements by CONNECT[12]. Orthogonal regression is used to classify the input string as a straight line, plane or space curve. If the edge list is not a statistically satisfactory straight line but does form an acceptable plane curve, the algorithm attempts to fit a circle. If this fails, the curve is smoothed and segmented at the extrema of curvature and curvature difference. The algorithm is then applied recursively to the segmented parts of the curve.

Some subtlety is required when computing geometrical descriptions of stereo acquired data. This arises in part from the transformation between the geometry in disparity coordinates and the camera/world coordinates. The former is in a basis defined by the X coordinates in the left and right images and the common vertical Y coordinate, the latter, for practical considerations (eg there is no corresponding average or cyclopean image), is with respect to the left imaging device, the optical centre of the camera being at (0,0,0) and the centre of the image is at (0,0,f) where f is the focal length of the camera. While the transformation between disparity space and the world is projective, and hence preserves lines and planes, circles in the world have a less simple description in disparity space. The strategy employed to deal with circles is basically as follows: given a string of edge segments in disparity space, our program will only attempt to fit a circle if it has already passed the test for planarity, and the string is then replaced by its projection into this plane. Three well chosen points are projected into the world/camera coordinate frame and a circle hypothesised, which then predicts an ellipse lying in the plane in disparity space. The mean square errors of the points from this ellipse combined with those from the plane provide a measure of the goodness of fit. In practice, rather than change coordinates to work in the plane of the ellipse, we work entirely in the left eye's image, but change the metric so that it measures distances as they would be in the plane of the ellipse.

Typically, stereo depth data are not complete; some sections of continuous edge segments in the left image may not be matched in the right due to image noise or partial occlusion. Furthermore disparity values tend to be erroneous for extended horizontal or near horizontal segments of curves. It is well known that the stereo data associated with horizontal edge segments is very unreliable, though of course the image plane information is no less usable than for the other orientations. Our solution to these problems is to use 3D descriptions to predict 2D data. Residual components derived from reliable 3D data and the image projection of unreliable or unmatched (2D) edges are then statistically combined and tested for acceptance. Where an edge segment from the left image is entirely unmatched in the right then a 2D description is obtained (and flagged as such). By these methods we obtain more complete 2D and 3D geometrical description of the scene from the left eyes view than if we used only the stereo data. Figure 6 illustrates the GDB description.

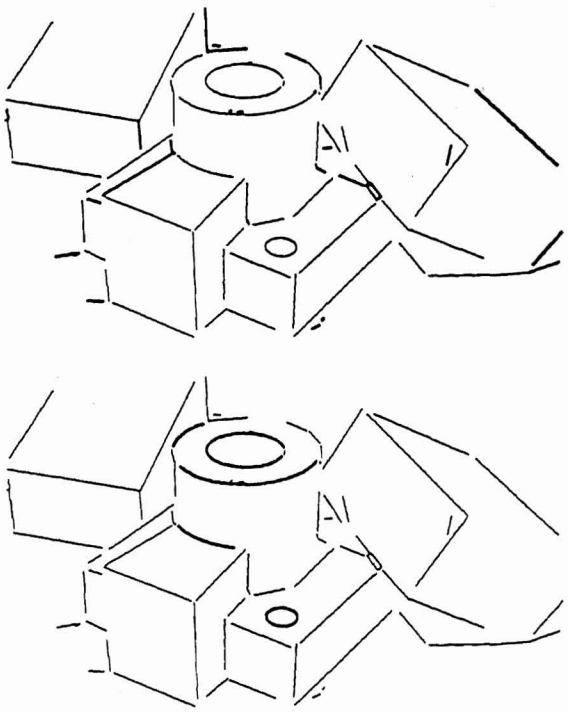

Figure 6. Geometrical descriptions. In (a) both 2 and 3 dimensional descriptions, with respect to the left hand image, are shown. Primitives of the GDB that are flagged as 2D, as a result of the fact that no depth data has been recovered for them by the stereo algorithm (perhaps as a result occlusion), are displayed bold. It is important to note that these exist only as descriptions in the image plane and not as descriptions in the world. In (b) again both 2 and 3 dimensional data are shown, but on this occasion circular sections (in three dimensions and not only in the image plane) of the GDB are the ones that have been highlighted by displaying them bold. Before segmentation each edge list is smoothed either by diffusion or by the approximately equivalent Gaussian (σ=2.5).

Evaluation of the geometrical accuracy of the descriptions returned by the GDF has employed both natural and CAD graphics generated images. The latter were subject to quantisation error and noise due to the illumination model but had near perfect camera geometry; they were thus used to provide the control condition, enabling us to decouple the errors due to the camera calibration stage of the process. A full description of the experiments are to be found in[13], suffice it to say that we find that typical errors for the orientation of lines is less than a degree, and for the normals of circular arcs subtending more than a radian, the errors are less than 3 degrees in the CAD generated images and only about twice that for images acquired from natural scene. The positional accuracy of features and curvature segmentation points has also been evaluated, errors are typically of the order of a few millimetres which maybe argues well for the adequacy of Tsai's camera calibration method more than anything else.

4. SMM: The Scene and Model Matcher.

The matching algorithm[14], which can be used for scene to scene and model to scene matching, exploits ideas from several sources: the use of a pairwise geometrical relationships table as the object model from Grimson and Lozano-Perez[15–17], the least squares computation of transformations by exploiting the quaternion representation for rotations from Faugeraus *et al*[18,19], and the use of focus features from Bolles *et al*[21]. We like to think that the whole is greater than the sum of its parts!

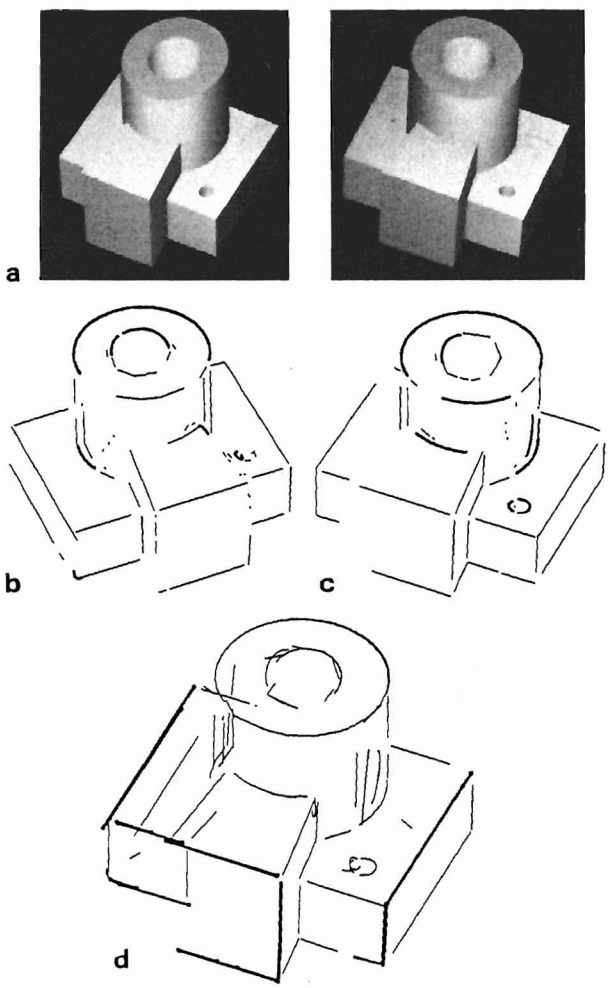

Figure 7. Matching scene descriptions: (a) shows a stereo view of the object/scene (obtained from the IBM WINSOM CSG body modeler); (b) and (c) GDB data extracted for two views of this object. Each description consists of approximately 50 above-threshold GDB line primitives. The 10 focus features chosen in view (b) obtained a total of 98 potential matches in view (c). Setting S to 7 and C to 4 only 78 independent implicit transformations result. After extension the best consistent transformation included 9 matched. The least square transformation (rotation followed by translation) that takes view (b) to view (c) is computed by the method discussed by Faugeras et al [18] in which rotations are represented as quaternions. In figure (d) view (b) is transformed into view (c) (the error in the computed rotation is 0.7 degrees) and matching lines are shown bold, the vast majority of the unmatched lines are not visible in both views (often as a result of noise).

The matching strategy proceeds as follows:

1) a focus feature is chosen from the model;
2) the S closest salient features are identified (currently salient means lines with length greater than L);
3) potential matches for the focus feature are selected;
4) consistent matches, in terms of a number of pairwise geometrical relationships, for each of the neighbouring features are located;
5) the set of matches (including the set of focus features) is searched for maximally consistent cliques of cardinality at least C, each of these can be thought of as an implicit transformation.
6) synonymous cliques (that represent the same implicit transformation) are merged and then each clique is extended by adding new matches for all other lines in the scene if they are consistent with each of the matches in the clique. Rare inconsistency amongst an extended clique is dealt with by a final economical tree search.
7) extended cliques are ranked on the basis of the number and length of their members.
8) the transformation implicitly defined by the clique is recovered using the method described by Faugeras *et al*[18].

The use of the parameters S (the neighbours of the focus feature), and C (the minimum subset of S) are powerful search pruning heuristics that are obviously model dependent. Work is currently in hand to extend the matcher with a richer semantics of features and their pairwise geometrical relationships, and also to exploit negative or incompatible information in order to reduce the likelihood of false positive matches.

The pairwise geometrical relationships made explicit in the matching algorithm can be used to provide a useful indexing scheme. Each primitive has associated with it a 1 dimensional hash table quantised by θ (their angular difference), each element of which includes a list, sorted by their absolute minimum separation, of pointers to look up table entries that lie within the associated θ, bucket (this can be searched rapidly using a binary search). This scheme allows relationships found in one scene description to be compared rapidly with relationships present in the other.

An example of the performance of the matching algorithm is given in Figure 7.

5. TIED: the integration of edge descriptions.

The geometrical information recovered from the stereo system described above is uncertain and error prone, however the errors are highly anisotropic, being much greater in depth than in the image plane. This anisotropy can be exploited if information from different but approximately known positions is available, as the statistical combination of the data from the two viewpoints provides improved location in depth. From a single stereo view the uncertainity can only be improved by exploiting geometrical constraints. A method for the optimal combination of geometry from multiple sensors based on the work of Faugeras *et al*[21] and Durrant-Whyte[22] has been developed[23],

and extended to deal both with the specific geometrical primitives recovered by the GDF and the enforcing of constraints between them.

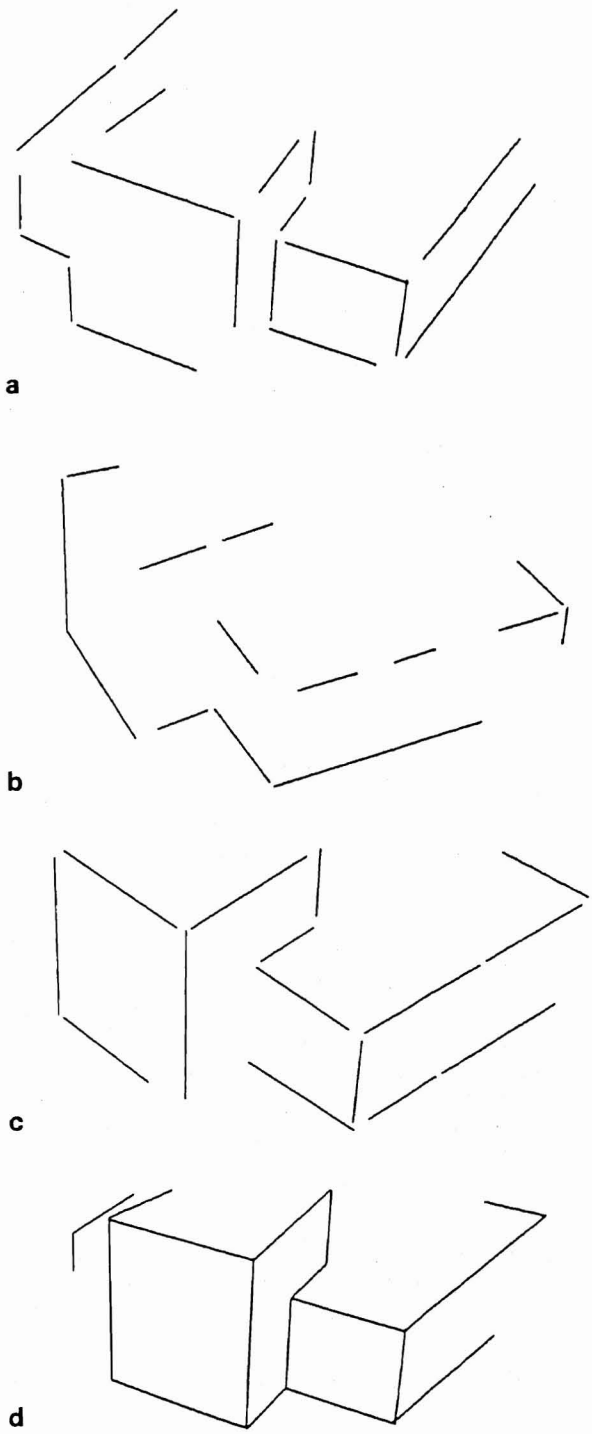

Figure 8. Geomstat: statistical combination: (a), (b) and (c) show details from real stereo views of our test object. These are matched by SMM, optimally combined, and have obvious geometrical constraints imposed (eg perpendicularity, intersection, parallelity etc) with the result given in (d).

One application of GEOMSTAT (Figure 8) is the acquisition of accurate and complete wireframe models of objects or environments from multiple stereo views. To allow the problem to be linearised SMM is used to give a first estimate of the transformation that takes one scene description into another. This suboptimal transformation is applied to each description in the first view to bring them into near correspondence with those in the second. Subsequently the constraints that lines from one view match with lines in the other are imposed in turn and the [statistical] estimate of their positions updated. Each merge results in minor modifications to the current estimate of the transformation, with the final result being optimal (due to the fact that the anistropies of the stereo process have been taken into account). This process can be repeated to incorporate subsequent views resulting in ever improved statistical estimates of the structure of the object.

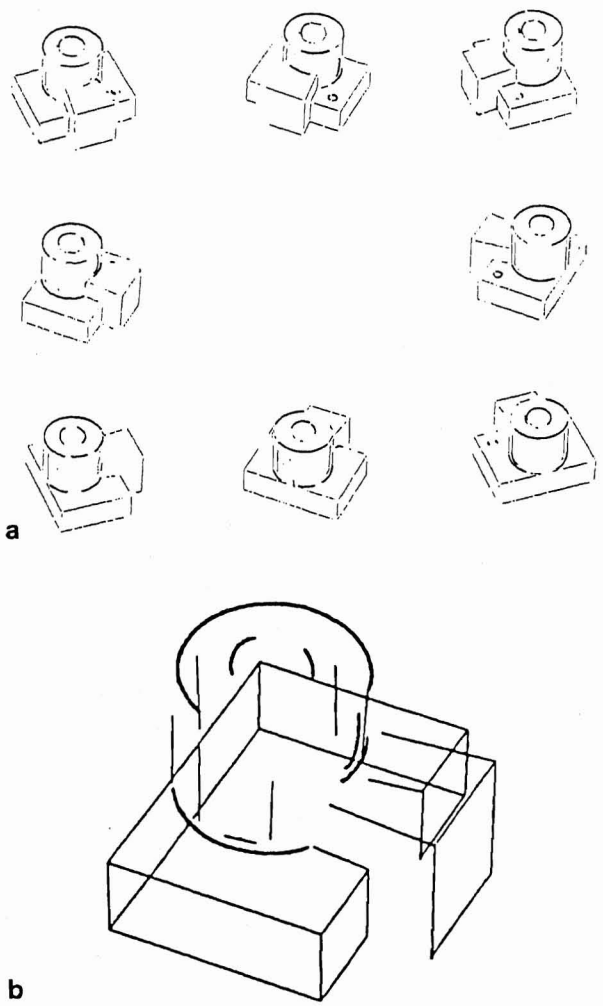

Figure 9. The integration of linear edge geometry from multiple views: (a) 3D data extracted from eight views of the object to be modelled (produced by the IBM WINSOM CSG body modeler); (b) the final visual model.

Figure 9 shows the automatic generation of a visual model through the integration of linear edge geometry from multiple views. To ensure a description of the model suitable for visual recognition and to allow greater generality we combine geometrical data from the multiple views of the object to produce a primitive visual model of it. Their combination is achieved by incrementally matching each view to the next. Between each view the model is

updated, novel features added and statistical estimation theory used to enforce consistency amongst them (here only through the enforcement of parallelism and perpendicularity). Finally only line features that have been identified in a more than a single view appear in the final visual model. The positions of extremal boundaries are viewpoint dependent and their treatment requires a degree of subtlety not yet present in our vision system, firstly to identify them, and secondly to treat them appropriately in the matching and geometrical integration processes. Clearly, though not position invariant, in the case of cylinders at least the relative orientation is stable over view and this information could be exploited. In figure 9, both the circular arcs and the extremal boundaries are displayed for largely cosmetic purposes.

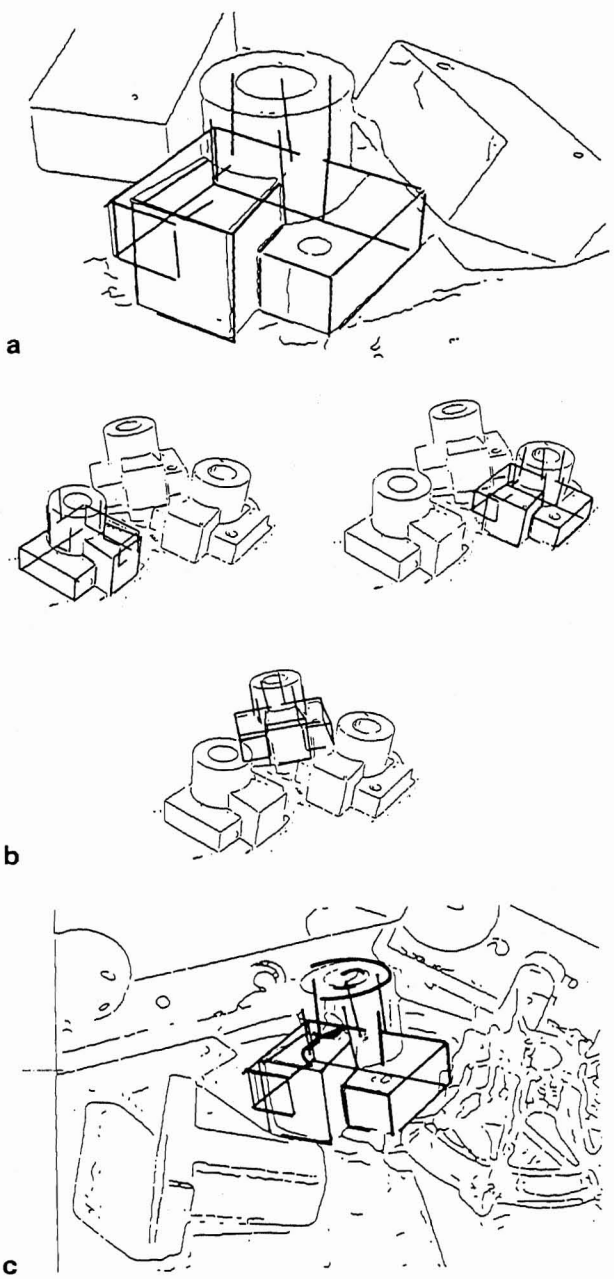

Figure 10. Object location.

GEOMSTAT is also used to obtain the statistically optimum estimate of the position and direction cosines of the target object coordinate frame after the SMM matching stage has been completed. This is done by enforcing the constraints that the axes of the coordinate frame are parallel to all the lines they should be, that they are mutually perpendicular, and intersect at a single point The result of the application of this stage of the process is the position and attitude of the object in the world coordinates.

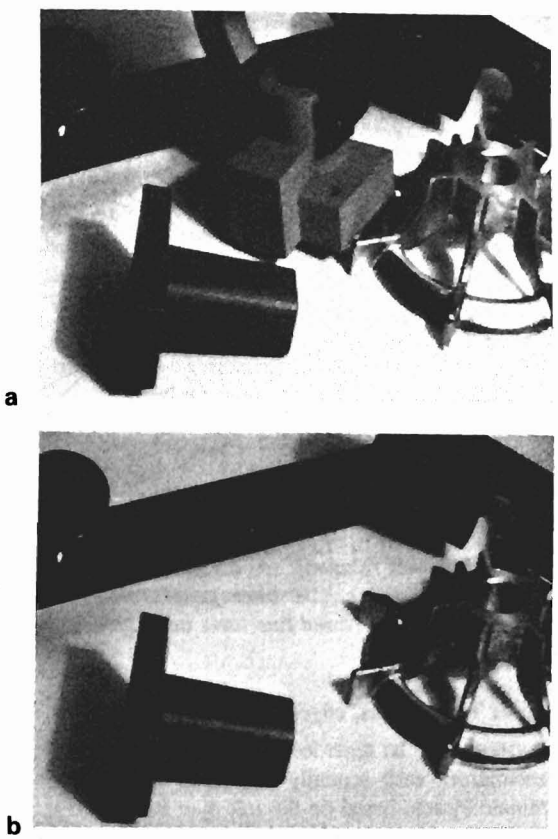

Figure 11. Closing the loop: (a) and (b) show the arm grasping the object and the scene with the object removed. Model matching for this scene (from which the grasp position is calculated) is illustrated in figure 10(c).

Figure 10 illustrates the SMM matching the compiled visual model in a number of scenes. Given the simple visual *model* that has been constructed in the previous sections it is possible to match it, using SMM, to an instance of the object in a cluttered scene. Three examples are illustrated here. In each example the dark lines depict the projection of the object model into the scene geometry after being transformed by the rotation and translation produced by the matching process (SMM) and the geometry integration process (TIED). To give some idea of the scale of the matching search problem, the object model contains 41 features and the scene in Figure 10 (a) contains 117. Some 10 model focus features, chosen on the basis of length, resulted in the expansion of only 292 local cliques. The latter were required to be of magnitude at least C=4 from S=7 neighbouring features. The largest extended

clique found by the matcher contained 13 matched lines. Figure 10 (c) depicts the scene viewed by the camera rig in figure 1.

The information provided by matching gives the RHS of the inverse kinematics equation which must be solved if our manipulator is to grasp the object (Figure 11).

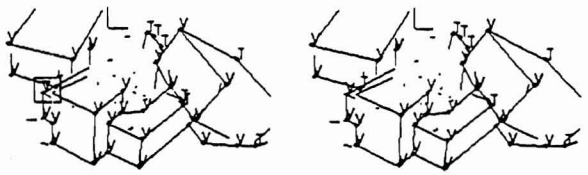

Figure 12. Wireframe completion.

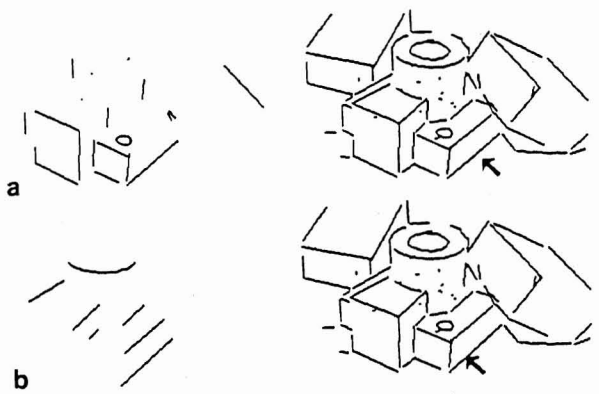

Figure 13. Pairwise relations: (a) all the lines perpendicular to the arrowed line have been generated; (b) all the lines parallel to the arrowed line have been generated.

6. REV: The regions, edges, vertices graph.

The system may be regarded as generating a sequence of representations each spatially registered with respect to a coordinate system based on the left eye: image, edge map, depth map and geometrical description. In the initial stages of processing a pass oriented approach may be appropriate but we consider that it is desirable to provide easy and convenient access between the representations at a higher level of processing. The REVgraph is an environment, built in Franz Lisp, in which the lower level representations are all indexed in the same co-ordinate system. On top of this a number of tools have been and are being written for use in the development of higher level processes which we envisage overlaying the geometrical frame with surface and topological information. Such processes will employ both qualitative and quantitative geometrical reasoning heuristics. In order to aid debugging by keeping a history of reasoning, and increase search efficiency by avoiding backtracking, the REVgraph contains a consistency maintenance system (CMS), to which any processes may be easily interfaced. The CMS is our implementation of most of the good ideas in Doyle[24] and DeKleer[25] augmented with some our own. The importance of truth maintenance in building geometrical models of objects was originally highlighted by Hermann[26]. Details of the REVgraph and CMS implementation may be found in Bowen[27].

Figure 12 illustrates a prototype wireframe completion algorithm. It links straight edges together to form T-junctions or vertices as appropriate. Inconsistencies between such labelings are identified and handled by the CMS. In this case the ambiguity is slight only 6 possible solutions (contexts) result, two of which are shown above (with vertices labeled V and T-junctions labeled T). The context on the right was adjudged by the program to be the most complete, while the one on the left contains a rather dubious T-junction where there should be a vertex (marked on the far left of the modeled object). The search space was bounded by some simple heuristics, using evaluations over the various CMS contexts, which is why a few lines are left incomplete where insufficient depth information is available. Note that incorrect decisions are possible (eg. the edge along the right hand side of the base of the cylinder which forms a vertex with the block on its right).

Figure 13 showe some useful pairwise relationships that are made explicit within the REVgraph environment. The formation of a pairwise relations table is a utility in the REVgraph. It generates pairs of lines and the geometric relations between them according to certain user requests.

7. Conclusions

We demonstrate the ability of our system to support visual guided pick and place in a visually cluttered but, in terms of trajectory planning, benign manipulator workspace. It is not appropriate at this time to ask how long the visual processing stages of the demonstration take, suffice it to say that they deliver geometrical information of sufficient quality, not only for the task in hand but to serve as a starting point for the development of other visual and geometrical reasoning competences.

Acknowledgements

We gratefully acknowledge Dr Chris Brown for his valuable technical assistance. This research was supported by SERC project grant no. GR/D/1679.6-IKBS/025 awarded under the Alvey programme.

References

1 Canny J.F. (1983), Finding edges and lines in images, MIT AI memo, 720, 1983.

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

3 Pollard S.B., J. Porrill, J.E.W. Mayhew and J.P. Frisby (1985), Disparity gradient, Lipschitz continuity and computing binocular correspondences, *Proc. Third Int. Symp. on Robotics Res.* 19-26.

4 Burt P. and B. Julesz (1980), Modifications of the classical notion of panum's fusional area, *Perception* 9, 671-682.

5 Arnold R. D. and T. O. Binford (1980) Geometric constraints in stereo vision, *Soc. Photo-Optical Instr. Engineers*, 238, 281-292.

6 Trivedi H.P. and S.A. Lloyd (1985), The role of disparity gradient in stereo vision, Comp. Sys. Memo 165, GEC Hirst Research Centre, Wembley, England.

7 Porrill J. (1985) Notes on: the role of the disparity gradient in stereo vision, AIVRU Lab Memo 009, University of Sheffield.

8 Tsai R.Y. (1986), An efficient and accurate camera calibration technique for 3D machine vision, *Proc IEEE CVPR 86*, 364-374.

9 Pollard S.B. and J. Porill (1986), Using camera calibration techniques to obtain a viewer centred coordinate frame, AIVRU Lab Memo 026, University of Sheffield.

10 Pridmore T.P., J. Porrill and J.E.W. Mayhew (1986), Segmentation and description of binocularly viewed contours, *Alvey Computer Vision and Image Interpretation Meeting*, University of Bristol, and *Image and Vision Computing* 5 No 2 132-138.

11 Porrill J., T. P. Pridmore, J. E. W. Mayhew and Frisby, J. P. (1986a) Fitting planes, lines and circles to stereo disparity data, AIVRU memo 017

12 Pridmore T.P., J.E.W. Mayhew and J.P. Frisby (1985), Production rules for grouping edge-based disparity Data, *Alvey Vision Conference*, University of Sussex, and AIVRU memo 015, University of Sheffield.

13 Pridmore T.P. (1987), The Interpretation of edge based binocular disparity information, Phd Thesis, University of Sheffield.

14 Pollard S.B., J.Porrill, J.E.W. Mayhew and J.P. Frisby (1986), matching geometrical descriptions in 3-space, *Alvey Computer Vision and Image Interpretation Meeting*, Bristol, AIVRU Memo 022 and *Image and Vision Computing* 5 No 2 73-78.

15 Grimson W.E.L. and T. Lozano-Perez (1984), Model based recognition from sparse range or tactile data, *Int. J. Robotics Res.* 3(3): 3-35.

16 Grimson W.E.L. and T. Lozano-Perez (1985), Recognition and localisation of overlapping parts from sparse data in two and three dimensions, *Proc IEEE Int. Conf. on Robotics and Automation*, Silver Spring: IEEE Computer Society Press, 61-66.

17 Grimson W.E.L. and T. Lozano-Perez (1985), Search and sensing strategies for recognition and localization of two and three dimensional objects, *Proc. Third Int. Symp. on Robotics Res.*

18 Faugeras O.D., M. Hebert, J. Ponce and E. Pauchon (1984), Object representation, identification, and positioning from range data, *Proc. 1st Int. Symp. on Robotics Res*, J.M. Brady and R. Paul (eds), MIT Press, 425-446.

19 Faugeras O.D. and M. Hebert (1985), The representation, recognition and positioning of 3D shapes from range data, *Int. J. Robotics Res*

20 Bolles R.C., P. Horaud and M.J. Hannah (1983), 3DPO: A three dimensional part orientation system, *Proc. IJCAI 8*, Karlshrue, West Germany, 116-120.

21 Faugeras O.D., N. Ayache and B. Faverjon (1986), Building visual maps by combining noisy stereo measurements, *IEEE Robotics conference*, San Francisco.

22 Durrant-Whyte H.F. (1985), Consistent integration and propagation of disparate sensor observations, *Thesis, University of Pennsylvania.*

23 Porrill J., S.B. Pollard and J.E.W Mayhew (1986b), The optimal combination of multiple sensors including stereo vision, *Alvey Computer Vision and Image Interpretation Meeting*, Bristol, AIVRU Memo 25 and *Image and Vision Computing* 5 No 2 174-180.

24 Doyle J. (1979), A truth maintenance system, *Artificial Intelligence* 12, 231-272.

25 DeKleer J. (1984), Choices without backtracking, *Proc, National Conference on Artificial Intelligence*,

26 Herman M. (1985), Representation and incremental construction of a three-dimensional scene model, CMU-CS-85-103, Dept. of Computer Science, Carnegie-Mellon University.

27 Bowen J.B. and J.E.W. Mayhew (1986), Consistency maintenance in the REV graph environment, *Alvey Computer Vision and Image Interpretation Meeting*, University of Bristol, AIVRU Memo 20, and *Image and Vision Computing* (in press).