

## USE OF GRAPHICS IN DECISION AIDS FOR TELEROBOTIC CONTROL (Parts 5-8 of an 8-Part MIT Progress Report)

Thomas B. Sheridan, James B. Roseborough, Hari Das, Kan-Ping Chin, and Seiichi Inoue

Man-Machine Systems Laboratory  
Massachusetts Institute of Technology  
Cambridge, MA 02139

This paper summarizes four separate projects recently completed or in progress at the MIT Man-Machine Systems Laboratory. Four others are described in a companion paper in Volume I.

### 5. A DECISION AID FOR RETRIEVING A TUMBLING SATELLITE IN SPACE - James B. Roseborough

**Abstract.** A decision aid for retrieving a tumbling satellite in space was constructed and tested in the laboratory. It was found that, though a perfect aid improves decision making, an aid whose modeling errors are nearly equal to those of the human operator will degrade performance. In addition, an aid presenting only point estimates is superior to an aid that also presents uncertainties.

**Introduction.** In the future, servicing of satellites in space may become more cost effective than simply replacing malfunctioning satellites with new ones. Retrieving a satellite for servicing can be a problem, however, especially if the satellite is nutating or tumbling. A decision aiding system was constructed for this situation that will help astronauts in this and other retrieval tasks.

The present approach to this problem is described in Rice et.al. [1] and Hartley et. al. [2]. In it, a vehicle such as a Manned Maneuverable Unit (MMU) is first positioned near the target satellite. Then, the MMU circles the target at the same rotation rate and about the same axis as the target. Finally the MMU closes in on the target until the grappling fixture of the target is secured to the MMU. Once this is done the MMU can be used to slow the rotation of the target with its thrusters. Experiments have shown that the task performed in this way is difficult, especially for complex motion like nutation and tumbling, and the Solar Max mission confirmed the problems in this approach.

**Decision aid design.** A decision aid has been designed [3] in which normatively derived state estimates are presented to the human operator. Figure 1 shows the elements of this system.

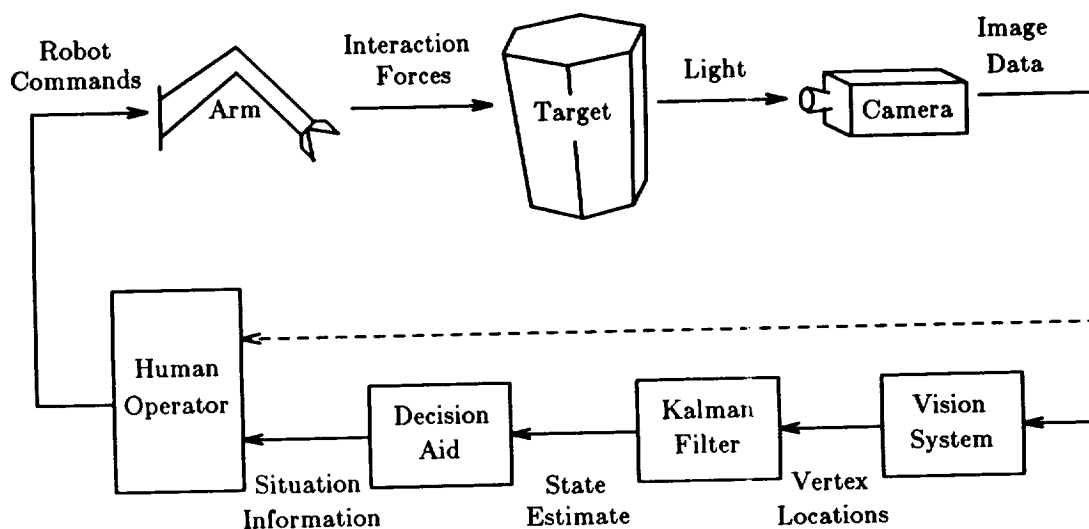


Figure 1. The components of a satellite retrieval decision aid based on state estimates.

In our conception of the retrieval mission, the satellite first achieves a parking orbit adjacent to the target satellite. Then a decision support system is used to build up a model of the rotational motion of the satellite in a computer. The operator can work with this model, using it to predict future states of the target or look at past states. Finally, the operator will use a robotic arm to reach out and grapple the satellite as it rotates, using his decision aid to help him select a good opportunity for mission success. Alternatively, the operator gives only the final signal to an automatic arm control and retrieval subsystem.

The information provided by the decision aid was considered along three dimensions:

1) what variables were displayed, 2) over what timescales they were shown, and 3) what statistical information was included. Displays included both pictographic images, in which a measured position was displayed directly, and derived decision variables. For example, we defined an objective function that can be loosely called "grappleability," or simply "goodness." This function conveys in a single variable the degree to which a certain process state represents an ideal grappling opportunity. This function reduces the information demands on the operator by transforming the decision problem from that of observing and processing six independent variables to that of processing a single process state indication, grappleability. Also, an acceptability display, was available which presented the probability that the orientation would be within a certain range of orientations that are acceptable for grappling.

For this problem, past as well as future values were provided. Past values give the operator a sense of the overall statistics of the process, while future predictions provide the operator with uncertain estimates of where the process will be in over a prediction interval. Predictor displays having three different resolutions were used in the experiment. These were called historic, long term, short term, and '8-12' second displays.

To behave normatively, the operator must hold a *belief state*, which is a distribution over the state space of the process. While this is impossible for most reasonable systems, a first order approximation such as that provided by a Kalman filter can be used by the computer and communicated to the operator. Our decision aid provided both information with varying levels of statistical detail.

**Experiments and Results.** Experiments were performed [3] using the decision aid in various configurations to determine which types of information were most useful. Regardless of the specific performance measure used, the perfect aid was superior to no aid, and degraded aids were worse than no aid. It is noteworthy that in both local and global terms, point estimates produced better decision making than displays that used uncertainties. A possible explanation is that when presented with statistical information, the human will first derive point estimates, so by having the computer provide only point estimates, workload is reduced and accuracy is increased.

Another interesting result is the relation of global performance (error) to motion complexity, or degrees of freedom, as in Figure 2. Here it is seen that for simple tasks, that is, one DOF tasks, the decision aid is not necessary, and even reduces decision performance slightly. For difficult cases of two and three DOF motion, the decision aid was useful, since the human has no ready algorithms for computing the long term future states of the process.

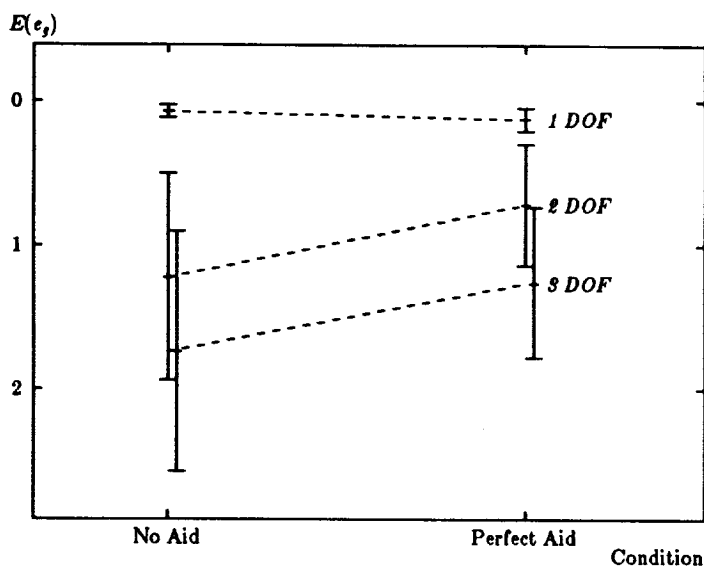


Figure 2. Global squared error as a function of the decision aid and the motion degrees of freedom.

An interesting comment was made by one subject regarding the use of low quality information provided by the decision aid. When asked if he thought it helped him, he said, "My guess is it did, but I am not sure. Whenever the information was there I just wanted to use it."

This suggests a picture of the displayed information as being an active element, rather than the having the passive, inert nature that it is most often given. Further support is found when we recognize that the presence of predictive displays resulted in hasty decisions.

The following qualitative observations were also made:

- (1) The task is actively reshaped in terms of the available measurements. If the probability of an acceptable range of endpoint angles is displayed, the subject will perform the task in a way that tends to maximize this probability.
- (2) Much of the human's activity in the task was directed towards finding patterns in the environment. Frequently historic displays were used to examine presence or absence of periodicity - a concept that is not explicitly modeled by the state-estimation-based decision aid.
- (3) Information about past values is less active than information about future values. Subjects were less likely to make hasty decisions when using historic information than when using predictive information.
- (4) The decision aid can have significant emotional effects on the user. For example, one subject reported that the aid gave him a feeling of confidence in making his decisions, and allowed him to manage his own attention resources much better than when it was not available. Another example is the undue sense of urgency that caused hasty actions in several cases. As these ultimately affect the quality of the decisions made, they should not be passed over in real systems.

#### References

- (1) Hartley, C.S., Cwynar, D.J. Garcia, K.D. and Schein, R.E., Capture of satellites having rotational motion, *Proc. Human Factors Soc. 30th Ann. Meeting*, 1986.
- (2) Rice, J.R., Torchak, J.P., and Hartley, C.S., Planning for unanticipated satellite servicing operations, *Proc. Human Factors Soc. 30th Ann. Meeting*, 1986.
- (3) Roseborough, J.B., *Aiding Human Operators with State Estimates*, PhD Thesis, MIT, Cambridge, MA, May 1988.

## 6. KINEMATIC CONTROL AND GRAPHIC DISPLAY OF REDUNDANT TELEOPERATORS - Hari Das

**Abstract.** The goal of this work is to help the operator of a redundant teleoperator perform end effector positioning and orienting tasks in a three dimensional world while avoiding collision between the teleoperator and obstacles in the environment.

**Introduction.** We limit ourselves to:

- 1) situations in which the operator does not have direct view of the teleoperator and has to use a displayed view to perform tasks,
- 2) solving the kinematic problem (we do not address the dynamics and control problem) of attaining the desired end effector position and orientation while keeping all parts of the teleoperator away from obstacles,
- 3) improving the operator's perception of the environment and the location of the teleoperator in it with an improved visual display, and
- 4) showing, with simulation experiments on human subjects, that our proposals achieve the goal .

We develop an interface between the operator and the teleoperator that is designed to give the operator good visual sense of the environment and to enable simple instruction by the operator on desired trajectories for the teleoperator.

The unique features of this work are the numerical inverse kinematic algorithm for handling kinematically redundant teleoperators to reduce the workload on the operator, and a technique for modelling the environment to enable its display from various viewpoints. Our experiments on human operator performance show that the display greatly affects the ability to perform tasks well. The idea of automatically determining the best view to display is also explored.

**Kinematic Control.** The method chosen in our work is a numerical inverse kinematic approach. The operator specifies end effector paths while the computer determines the best configuration for the teleoperator that maintains the end effector at the specified position and orientation while keeping other parts of the teleoperator as distant from obstacles as possible. Path planning is a cooperative effort between the operator and the computer aid.

The numerical method we use to solve the inverse kinematics is to first formulate the problem as a constrained optimization, then use a hybrid of the generalized inverse and the method of steepest descent to solve for joint positions (Das, 1989).

**Visual Display.** We propose the creation, from sensor information, of a computer data structure to represent objects in the environment. In this scenario, as more sensor data is received, the data structure is updated to improve the model. A view of the world chosen by the operator and represented in the computer can then be displayed on a graphic screen ( see Fig. 3). Advantages of representing the environment in such a data structure are:

- 1) views of the model can be drawn as seen from any point in space.
- 2) any geometric information on objects in the environment may be processed from the data.
- 3) the reconstructed environment may be displayed visually to the operator while a camera image in poor lighting or visibility conditions may not provide much information.

We assume that a model of the environment is available. The problem of processing sensor information to form a model of objects in the environment has been pursued by others (Winey, 1981, Marce and Even, 1988). We have identified some elements of a good view and have developed an algorithm to select it. The determination of a good view is based on the idea that it is desirable to have closest distances between the teleoperator and obstacles orthogonal to the line of sight (Das, 1989).

**Experiments.** A simulation of a twelve d.o.f. vehicle-manipulator teleoperator (6 d.o.f. wrist-partitioned manipulator on a 6 d.o.f. vehicle) has been developed to test the proposals suggested in this work. In experiments, the performance of human subjects on positioning and orienting the end effector at specified locations in a 3 dimensional space among a field of obstacles was measured. Time taken to complete the tasks and number of collisions with obstacles while performing tasks were used to determine performance.

Experiments have been conducted, comparing the inverse kinematic control with the operator having a 6 d.o.f. end effector position and orientation input to a fully manual control method with the operator having a 12 d.o.f. input. Two six d.o.f. input devices were used, one to position and orient the vehicle and the other to position and orient the end effector with respect to the vehicle. A closed-form solution was used to solve the inverse kinematics of the wrist-partitioned manipulator.

In addition, human subject performance with an automatically selected view was compared to an operator selected view, a simulated view from a fixed point in the 3 dimensional space and a simulated view from the vehicle.

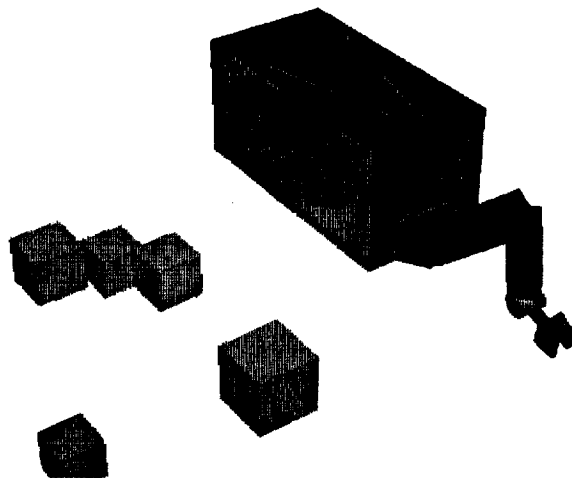


Figure 3. Vehicle-manipulator system as seen by human operator.

**Conclusions.** Briefly, the results from our experiments indicate:

- 1) Performance is better when the human subjects used the 6 d.o.f. end effector positioning and orienting method.
- 2) The different options on viewing greatly affected human subjects' performance.
- 3) Performance was best when the subjects were able to select a view.
- 4) Performance with the automatically selected view was better than with the simulated views either from the vehicle or from a fixed point.

## References

- (1) Das, H., *Kinematic Control and Visual Display of Redundant Teleoperators*, Ph. D. thesis in progress, MIT., Jan., 1989.
- (2) Winey, C. M., *Computer Simulated Visual and Tactile Feedback as an Aid to Manipulator and Vehicle Control*, M.S. Thesis, MIT, June, 1981.
- (3) Even, P. and Marce, L., *Manned Geometric Modelling for Computer Aided Teleoperation*, *Proc. Int. Symp. on Teleoperation and Control*, Bristol, England, July, 1988.

## 7. REAL-TIME TERRAIN / OBJECT GENERATION: A QUAD-TREE APPROACH - Kan-Ping Chin

**Abstract.** A hidden surface removal algorithm for a projected grid surface is developed to accelerate the display of 3-dimensional terrain or object. The size of the database is reduced to about one fifth of the database with constant sampling resolution; and the frame speed of the system is about four times faster.

**Introduction.** A realistic 3-dimensional computer-graphic presentation is important for training the operator of a teleoperated vehicle [1] or for control based on a computer model. The feeling of realism depends on two factors : (1) a realistic image, and (2) a real-time display of the image. A realistic terrain image usually requires a very large database to record appropriate geometric terrain height information at latitude and longitude in a regular reference grid (coordinates of each sample point); however, the manipulation of a large database is very time consuming. On the other hand, a small database which samples at sparser points of a terrain can be displayed very fast, but may sacrifice the realism of the image. Thus, there is a conflict in terms of computer graphics between these two aspects: the realistic instantaneous image, and the real-time display of the image.

A method has been developed to reduce the database size without sacrificing realism in the image. In this method, instead of sampling height at regularly spread points of the horizontal grid of the terrain, only the points where the gradient changes most rapidly are sampled. In this way, the database size is reduced. A quad-tree data structure is used to store the sampled data. A recursive algorithm is developed to retrieve the data from the quad-tree and to display it in a specific order so that the hidden surfaces are also removed.

**Sampling Data at Different Resolutions.** As Attneave [2] suggested, the points of concentrated information in figures are where their gradients change most rapidly. Therefore, common objects may be represented with great economy and fair fidelity by marking the points around the sharpest gradient change (i.e. small radius of curvature) and then connecting these points with straight lines. As shown in Figure 4, the recognizability of the object indicates that most of the important information has been retained.

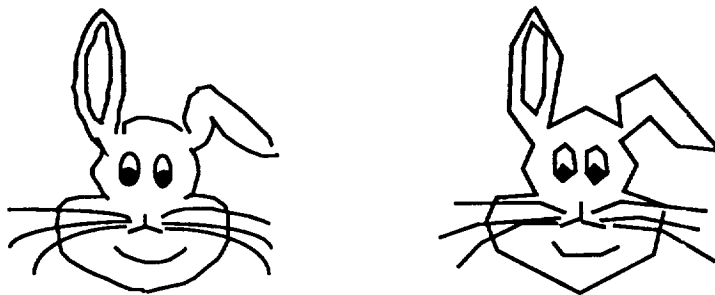


Figure 4. A curvilinear object represented by straight lines.

Applying the above concept, a mountain range that has varying gradient is intensively sampled; and a plain, being regarded as featureless, is sparsely sampled. In this way, a sampled database is smaller than the one with constant sampling distance. For a natural scenery such as terrain, this averaging over "texture" of the figure is analogous to what happens when a halftone photograph is recopied several times on high contrast paper. However, since the data is irregularly scattered if we sample it at various resolutions, we need a special data structure to handle it.

**Quad-tree Data Structure.** A quad-tree data structure[3], often used in image processing, can hierarchically store graphic data of a picture at various resolutions. A square picture can be divided into four sub-squares, or not divided at all, according to a predefined criterion such as, in this case, curvature of the picture; the sub-squares can be iteratively further subdivided if necessary. In this way, variable resolution of a picture are efficiently stored in a hierarchical, tree-like structure.

The place to store the graphic information of the picture is called a "node". Each node either has four branches which correspond to the four subdivided pictures, or has no branch at all; accordingly, either the addresses of the child branches, or the information of the picture if there is no child branch, is stored at each node (Fig. 5).

To retrieve the pictorial information, we have to check the availability of the information at each node first. If the information of a picture is stored at a node, then we can retrieve it and generate the picture from the data retrieved. If there is no pictorial information in the node, we then use the addresses of the child branches to check the availability of data at each child branch repetitively. The procedure described is applicable to nodes everywhere in the quad tree; therefore, a subroutine calling itself recursively is used to walk through the tree.

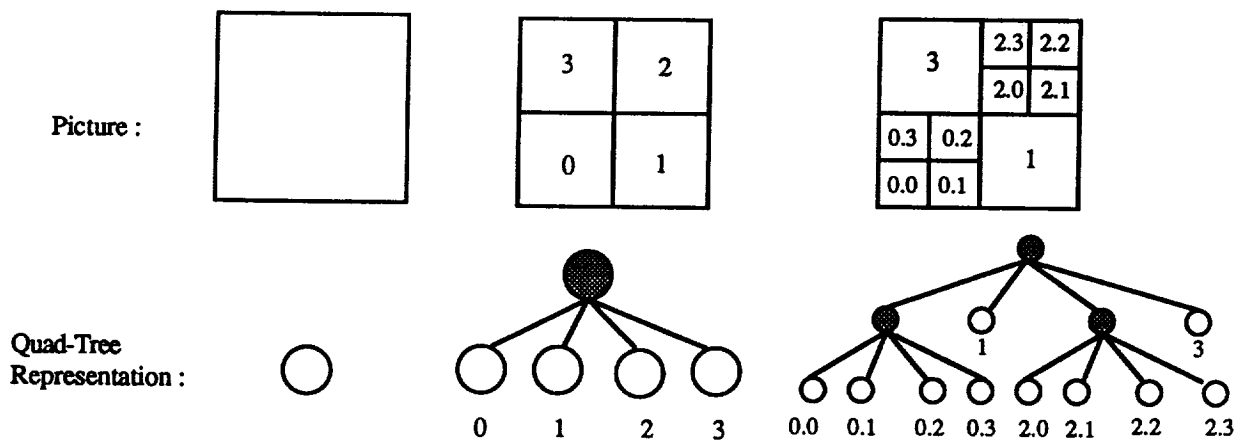


Figure 5. A quad-tree representation of a 4 X 4 "picture".

**Hidden Surface Removal Algorithm for Projected Grid Surface.** For a bivariate (in the form of  $z=f(x,y)$ ) grid surface, which we used to model the terrain, the projection of any facet is bounded by the projection of the boundary of the facet. This geometric property provides a simple way to enumerate the facets from far to near for a given view point [4]. Consequently, two of the most time consuming jobs in hidden surface removal algorithm, to compute the distance of objects to the viewing position and to sort those distances to determine the display sequence, are not essential anymore. That is why the present method, without doing these two jobs, is very fast compared with other methods.

We distinguish three different types of viewer location: the viewer sitting on top of the terrain, the viewer looking at the terrain from an edge, and the viewer looking at the terrain from a corner. As shown in Figure 6, for case 1, the viewer on top of the terrain, the terrain can be separated into nine regions and the viewer will be looking from region 0. The algorithm can be divided into 3 steps. Each step processes a specific group of regions. In each region, the facets are always processed row by row for display from the farthest one from the viewer to the nearest one, as depicted by the arrows in the picture. The other two cases, where the viewer is looking from an edge or from a corner of the terrain, are only special instances of case 1, and therefore can be solved by part of the above procedure. For example, if the viewer is looking from the lower edge of the terrain, the terrain can be separated into three regions and their display sequences are the same as in case 1.

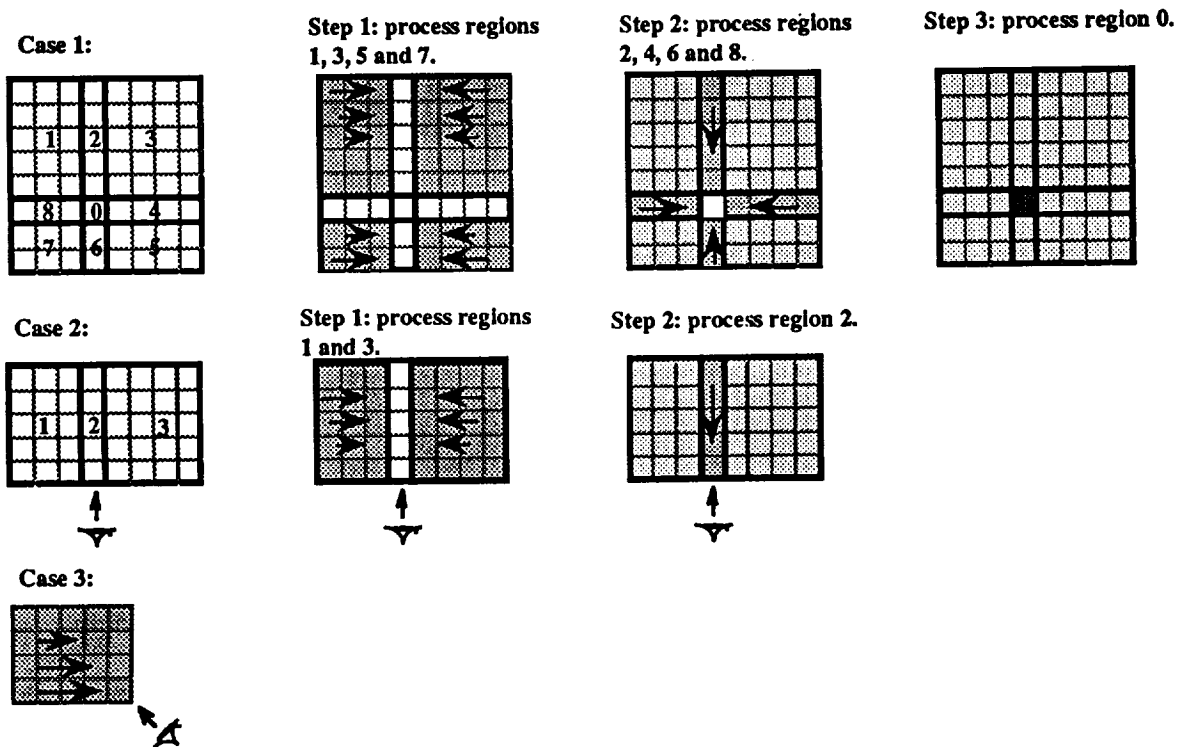


Figure 6. Hidden surface removal for a gridded surface. Case 1: viewer sitting on top of region 9. Case 2: viewer looking from bottom edge. Case 3: viewer looking from a corner.

Applying this method to the quad-tree data structure, each facet is represented by a tree. Each sub-facet (branch of the tree) must also be processed from far to near for each region. We can produce a table for the display sequence of the sub-facets for each region.

**Conclusion.** This method has been implemented in a C language computer program on an IRIS 2400 graphic workstation. The database is sampled manually from a real-world contour map. By sampling at different resolutions for different areas, the database is reduced to about one-fifth (from 2,048 patches to 416 patches) of the one with constant sampling resolution (the ratio may vary depending on the shape of the terrain.); and the frame speed of the system is about four times faster (from 3 frames per second to 13 frames per second). Gouroud (smooth) shading can be implemented to enhance the realism of the image; however, it decreases the frame speed to about 3 frames per second.

The terrain is integrated with a simulator which features the dynamics of a telerobot and associated structure. The simulator is about 9 frames per second without smooth shading and about 2.4 frames per second with Gouroud smooth shading.[5].

#### References

- [1] Sheridan, T.B., *Improving the Effectiveness of Remotely Operated Vehicles for Inspection, Maintenance and Repair of Offshore Structures through Simulation Aids*. MIT Sea Grant Program, 1985.
- [2] Attneave, F. Some informational aspects of visual perception, *Psychological Review*, Vol. 61, No. 3, 1954.
- [3] Hunter, G.M. and Steiglitz, K. Operations on images using quadtrees, *IEEE Trans. on Pattern Analysis and Machine Intell.*, V1, No. 2 pp. 145-153, April 1979.
- [4] Anderson, D.P. Hidden Line Elimination in Projected Grid Surfaces, *ACM Transactions on Graphics*, Vol. 1, No. 4, pp. 274-288, October 1982.
- [5] Chin, K.-P., *Human-Interactive Simulator for Underwater Remotely Operated Vehicles*, SM Thesis, Dept. of Mechanical Engineering, MIT, 1988.

## 8. TWO DIMENSIONAL CONTROL FOR THREE DIMENSIONAL OBSTACLE AVOIDANCE - Seiichi Inoue

**Abstract.** A technique has been developed by which a two-dimensional display can be used for guiding a hand or vehicle and ensuring that there is no collision in three-space. The human operator, using only a mouse (and cursor on his 2-D computer display), locates key points of 3-D obstacles, as seen by two cameras. The computer then indicates the intersection boundaries for any selected plane from the start point to the goal or subgoal. The operator can then select a robot hand trajectory on the plane by using the mouse and cursor and thereby guarantee obstacle avoidance.

**Key Point Measurement of 3-D Obstacle.** First we have to know the position of arbitrary points in 3-D space. Two TV cameras (mounted parallel to each other on the vehicle) are used to measure the projections of the object on the 2-D image plane.

Let 3-D points in task environment and 2-D points on either TV image plane (screen) be represented respectively by  $[x, y, z]$  and  $[x^*, y^*]$ . The relation between these points can be written as follows, where H is a scale factor.

$$[x \ y \ z \ 1] T = H [x^* \ y^* \ 0 \ 1] \dots \dots \dots (1)$$

The perspective transformation matrix T is

$$T = \begin{vmatrix} T_{11} & T_{12} & 0 & T_{14} \\ T_{21} & T_{22} & 0 & T_{24} \\ T_{31} & T_{32} & 0 & T_{34} \\ T_{41} & T_{42} & 0 & T_{44} \end{vmatrix}$$

Eliminating H yields

$$(T_{11} - T_{14} x^*) x + (T_{21} - T_{24} x^*) y + (T_{31} - T_{34} x^*) z + (T_{41} - T_{44} x^*) = 0 \dots \dots (2)$$

$$(T_{12} - T_{14} y^*) x + (T_{22} - T_{24} y^*) y + (T_{32} - T_{34} y^*) z + (T_{42} - T_{44} y^*) = 0 \dots \dots (3)$$

When two cameras are used, and the perspective transformation matrices are represented respectively by T1 and T2, the following simultaneous equations are obtained from equations (2) and (3).

$$(T_{111} - T_{141} x^*1) x + (T_{211} - T_{241} x^*1) y + (T_{311} - T_{341} x^*1) z + (T_{411} - T_{441} x^*1) = 0 \dots \dots \dots (4)$$

$$(T_{121} - T_{141} y^*1) x + (T_{221} - T_{241} y^*1) y + (T_{321} - T_{341} y^*1) z + (T_{421} - T_{441} y^*1) = 0 \dots \dots \dots (5)$$

$$(T_{112} - T_{142} x^*2) x + (T_{212} - T_{242} x^*2) y + (T_{312} - T_{342} x^*2) z + (T_{412} - T_{442} x^*2) = 0 \dots \dots \dots (6)$$

If matrices T1 and T2 and  $x^*1, x^*2, y^*1$  are known, the 3-D position x, y, z is obtained by use of equations (4), (5), and (6).

**Avoidance of Obstacles.** By use of the preceding method we can know the 3D positions of arbitrary points on a video display. Accordingly the operator, using a mouse and a cursor, locates key points  $[x^*1, x^*2, y^*1]$  of 3D obstacles on the 2D display. The computer then indicates the intersection point of an obstacle and any selected plane which includes the start point and the goal or subgoal.

Figure 7 shows, using an example of a task environment and the plane which an end-effector moves on, that obstacle A and plane C intersect, and obstacle B is above plane C. Anywhere the end effector moves on plane C, the operator can easily avoid the obstacle A.



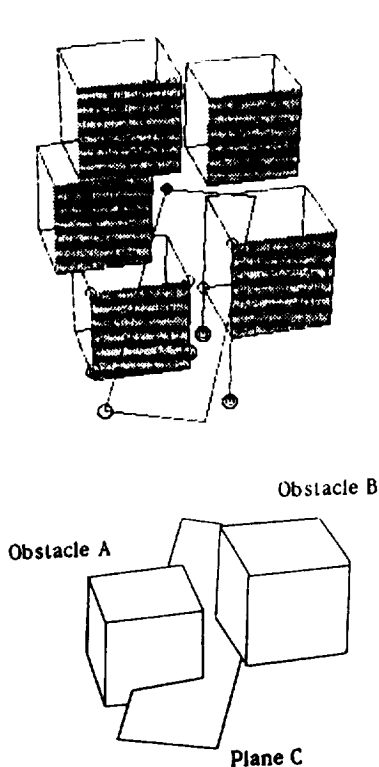


Figure 7. Task environment and intersection points of plane and obstacle.

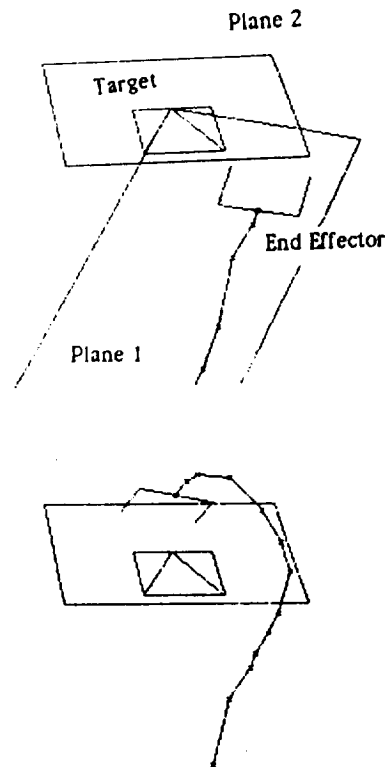


Figure 8. Moving end effector from plane 1 to plane 2.

**Controlling the End Effector on the Plane.** When the end effector moves on the plane from start point to the goal, we can have position control and one-axis rotation control in 3-D space by use of a 2-D input device like the mouse. If the desired position  $(x^*, y^*)$  for the end effector on the display is decided upon, then the 3-D position  $(x, y, z)$  of it can be calculated.

Not only robot hand or other end effector translation in 3-D space, but also rotation of it about the perpendicular axis to the plane, can be controlled by use of a 2-D input device. For example, one must control the rotation of the end effector when one is ready to grasp the target. Then another plane included the target can be set up, and the end effector can be transferred to the new plane, as shown Figure 8. After transfer to another plane, it is very easy for the end effector to approach and grasp the target.

**Reference.**

Inoue, S. and Hashimoto, T., 3-D Data Measurement with a Single Camera in an Arbitrary Position, *Research Reports of Fukui College of Technology*, Japan, 1986

**ACKNOWLEDGMENT.** Projects 1 and 2 were sponsored by Jet Propulsion Laboratory, Project 3 by Woods Hole Oceanographic Institution, and Project 4 by the Japanese Government. Their support is gratefully acknowledged.

