Dynamic Sensor Planning

Steven Abrams Peter K. Allen*
Center for Research in Intelligent Systems
Computer Science Department
Columbia University
New York, NY 10027

Konstantinos A. Tarabanis
IBM T. J. Watson Research Center
Yorktown Heights, NY 10598

Abstract

In this paper, we describe a method of extending the sensor planning abilities of the "MVP" Machine Vision Planning system to plan viewpoints for monitoring a preplanned robot task. The dynamic sensor planning system presented here analyzes geometric models of the environment and of the planned motions of the robot, as well as optical models of the vision sensor. Using a combination of swept volumes and a temporal interval search technique, it computes a series of viewpoints, each of which provides a valid viewpoint for a different interval of the task. By mounting a camera on another manipulator, the viewpoints can be executed at appropriate times during the task so that there is always a robust view suitable for monitoring the task. Experimental results monitoring a simulated robot operation are presented, and directions for future research are discussed.

1 Introduction

Recently, there has been much research in the field of sensor planning [3, 4, 5, 8]. The basic problem is that in setting up an automated system for monitoring some process, the effectiveness of the system can largely be determined by the locations, types and configurations of the sensors used. To manually determine these parameters on a case by case basis may not be cost effective or accurate. It may be better to have an automated system for determining the sensor locations and parameters for monitoring a given task.

To that end, many systems have been and are being developed which, based on geometric models of an environment and models of the sensors, can generate sensor locations and settings which provide a robust view of specific features so that the features are detectable, recognizable, measurable, or meet some other task constraints. In general, the sensors are cameras and a robust view implies that the camera must have an unobstructed view of the entire feature set, which must lie within the depth-of-field of the camera and must be magnified to a given specification. Sensor planning systems can then generate camera locations, orientations, lens settings (focus-ring adjustment, focal length, aperture), and in some cases lighting plans to insure a robust view of the features.

It is interesting to note that while research in robot motion planning abounds, research in sensor planning has focused on sensor planning for static scenes. It is our belief that an intelligent robot system capable of planning its own actions should be capable of planning its own sensing strategies. With a dynamic sensor planning system, this goal is closer to a reality. Robots involved in manufacturing or assembly can determine appropriate sensor locations. Teleoperators can have the robot system guarantee robust viewpoints during the operation. The intelligent motion plans which researchers spend so much effort computing can be monitored in an intelligent fashion.

We have been exploring methods of extending the sensor planning abilities of the "MVP" Machine Vision Planning [7, 8] system to function in environments where objects are moving, such as an active robot work-cell. In previous work [1], we described a technique for sensor planning in a dynamic environment, which was implemented using a simulated model of a simple moving object. Here, we present a detailed analysis of the dynamic sensor planning problem and improved versions of the original algorithms. In addition, experimental results using a model of a dual-robot work-cell are presented in which we automatically monitor a task in the work-cell.

2 Overview of Static Planning

A complete description of the MVP system is beyond the scope of this paper. For details, see [7, 8, 9]. In brief, MVP takes a constraint based description of the vision task

*This work was supported in part by DARPA contract DACA-76-92-C-007, NSF grants IRI-86-57151 and CDA-90-24735, North American Philips Laboratories, Siemens Corporation and Rockwell International.

1050-4729/93 $3.00 © 1993 IEEE

requirements and synthesizes what has been termed a *generalized viewpoint*, which is an eight-dimensional vector incorporating sensor location, orientation, and lens parameters including aperture and effective focal length.

MVP contains analytical relationships for each of the optical task constraints (resolution, focus, field-of-view), and uses 3-D solid geometric models of the environment to formulate visibility constraints. These constraints are combined in an optimization setting to produce a generalized viewpoint which meets all task constraints with as much margin for error in sensor placement and setting as possible (i.e., as far away from all hypersurfaces as possible). Using CAD descriptions of the object to be viewed and its environment, MVP generates the visibility region for viewing the desired features. This region is calculated to be the total volume in space from which the features are viewable without obstruction. This volume is used in the optimization stage of MVP for finding the best viewpoint.[1]

3 Motion in the Work-Cell

There are two basic cases which must be dealt with separately in the dynamic sensor planning problem. First is the case where the target objects, i.e. those features which must be viewed, remain stationary and other objects, such as the robot which is performing some operation on the stationary part, moves. This case can arise in tasks such as spray-painting, and spot-welding. Second is the case where the targets to be viewed are moving. This case arises in tasks such as pick-and-place, part insertion, materials handling, etc.

The main difference between these two cases is that in the first case, if a viewpoint is found to be valid at some point during the task, it is guaranteed to be valid with respect to all optical constraints at all times during the task. This is because the functions defining the optical constraints only depend on the target feature locations and the sensor parameters, and not on the positions or orientations of obstacles in the environment. This fairly obvious, but important property allows us to ignore changes in the optical constraints over time and focus only on changes in the geometric parameters, i.e. the visibility constraint.

The second case is more difficult because it requires an examination of how changes in the position and orientation of the target features effect the optical parameters, particularly focus and resolution. However, if the viewpoint is considered in terms of a coordinate frame attached to the feature set, the target can always be considered stationary with the entire environment considered as moving. The only limitation is that the entire feature set must be moving as a single rigid body, i.e. features can not move independently. While extremely important, independently moving features are not yet handled in this work, although it is being examined as part of ongoing research.

4 Overview of Our Approach

The approach being taken is a *Temporal Interval Search* method, which is is based on the use of swept volumes. The geometric models of the moving objects are swept through their paths to compute the regions in space which, during some interval, are occupied by some moving object in the environment. The MVP algorithms are then run using the swept volumes for the occluding bodies as opposed to the actual models, thus reducing the dynamic sensor planning problem to a static problem. If no viewpoint is found considering these swept objects over a time interval, a temporal interval search is performed to find the largest time intervals which can be monitored by a single viewpoint. This allows us to plan a series of viewpoints and the times at which they become feasible.

Given that we have an object O whose motion is known over a time interval T, we define $\mathcal{S}(T, O)$ to be the volume swept out by O during T. The key to using swept objects for sensor planning (or, in fact, for any collision avoidance problem) is that in planning around an obstacle given by $\mathcal{S}(T, O)$, you guarantee that you have avoided the actual obstacle O at any instant in interval T. This observation was made by Cameron in [2] for the "clash detection" (robot collision avoidance) problem.

Let V represent visibility volume for $\mathcal{S}(T, O)$. V is the set of all points (in 3-space) which give views of the target which have no obstructions (due to O) for the entire time interval T. If V is a null volume, there is no single viewpoint which would be valid for all of T. Even if V is not null, there is no guarantee that there are viewpoints within V which satisfy the optical constraints of MVP.

In using swept volumes for collision avoidance type problems, one looses all information regarding where the object is at any particular moment. We present a technique for recovering sufficient temporal information to plan sensor locations. If using V as a visibility volume, MVP is unable to find a viewpoint which meets all constraints, we conclude that T is too large an interval to plan a single viewpoint for, given the motion of O. We have no information concerning when any particular viewpoint becomes invalid; we only know that we can not find a single viewpoint which is valid for the entire interval. Recomputing $\mathcal{S}(T, O)$ for a shorter time interval T will yield a smaller obstacle, a larger V, and MVP may now be able to find a viewpoint.

[1] Here, and elsewhere in this paper, when we refer to a *viewpoint* we are actually referring to the *generalized* viewpoint mentioned earlier.

We can now present the algorithm formally. Assume we have a polygonal target τ which we wish to monitor during the time interval $T = [t_0, t_n]$. During T, there is a set of known obstacles O_0 through O_m, which move in known paths. The goal is to plan a single viewpoint valid for the entire interval, if such a point exists, or to determine a sequence of viewpoints which, when executed at the appropriate times, allow the features to be monitored for the entire interval.

Temporal Interval Search

1. Compute $\mathcal{S}(T, O_i)$ for each of the m obstacles.
2. Use MVP to compute a viewpoint using $\mathcal{S}(T, O_0)$ through $\mathcal{S}(T, O_m)$ as well as all stationary objects in the environment as the set of potential occluding bodies.
3. If MVP can successfully find a viewpoint, use this viewpoint for the entire time interval T.
4. If no such viewpoint is obtainable, divide the time interval in half yielding $T_1 = [t_0, t_{n/2}]$. Go back to step 1 using interval T_1.
5. If the entire time interval T has been planned, we are finished. If not, go to step 1 using the remaining portion of the the original interval T.

This process continues until a viewpoint has been found which is valid until t_n. The *critical times* are the endpoints of the intervals, i.e. the times at which the sensor must be moved.

The computation of swept volumes is central to this algorithm. If piecewise linear translational motion is all that is allowed, then the computation of swept volumes is certainly tractable [10]. Unfortunately, sweeping is not closed over the set of polyhedra when rotational motion is permitted. An articulated robot arm moves strictly in rotations about its joint axes, so the resulting swept volumes are not polyhedral (they would contain circular arcs, spherical patches, and other curved surfaces). These objects would not be usable in MVP. Korein gives an algorithm for computing polyhedral approximations [6] of the swept volumes formed by the motion of articulated robot links. These techniques may be used to simplify the computation of the swept volumes.

In the current dynamic sensor planning implementation, instead of computing a swept volume and then computing the resulting occlusion volume, we compute a set of volumes of occlusion at discrete points along the trajectory. These volumes of occlusion are then unioned to form the volume of occlusion for the entire interval. This is possible because the volume of occlusion generated by the union of a set of obstacles (for viewing a particular target) is equal to the union of the volumes of occlusion generated by each obstacle. The volumes of occlusion for a particular interval are subtracted from the reachability volume for the manipulator to give volume containing all points where the robot can place the camera such that the camera can see the target. The primary benefit of this approach is that subdivisions of the time interval do not require recomputing new swept volumes; the appropriate subset of the instantaneous occlusion volumes are simply unioned to form the volume of occlusion for any given interval.

5 Realization of the Viewpoints

The result of the temporal interval search will be a set of viewpoints and critical times at which to execute them. However, an explicit representation of time is not required for the temporal interval search, in which case the critical times are not times at all but, rather, critical *events*. If, for example, the motions of a robot have been planned as a series of joint-space moves, the critical events would be joint angle values. If the motion was planned in cartesian space, the critical events would be cartesian positions. Finally, if the robot motion was planned on some global time scale (perhaps avoiding other moving obstacles), the critical events would be actual times on this scale. As long as at task execution time there is a way to determine when the critical events arise, (i.e. by waiting for the robot to be within some distance of the prescribed position), the viewpoints can be realized.

6 Experimental Results

We have modeled our laboratory environment using a CAD system (see figure 1). The model includes two PUMA 560 robots and the object to be monitored during the task. The first robot (I) executes tasks, while the second robot (II) has a camera mounted on it. In the simulated experiment, robot I passes over the object as if it were performing an operation on it, such as spray-painting. During the task, robot II needs to monitor a feature inside the object. A CAD model of the object and the feature is shown in figure 2. The target (i.e. the feature to be viewed) is the top face of the inner cube.

In the experiment, the robot model is stepped through a series of positions along its planned trajectory. At each step, the volume of occlusion is computed as in the static sensor planning problem. The individual volumes of occlusion are unioned together to form the volume of occlusion for the entire trajectory. In this way, we approximate the volume of

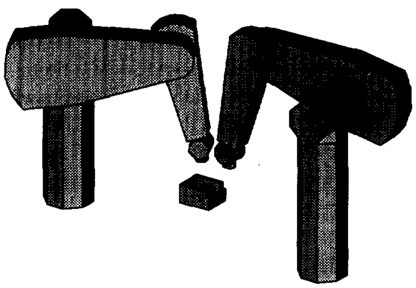

Figure 1: CAD Model of the environment.

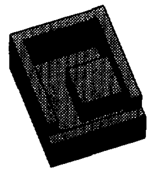

Figure 2: CAD Model of the part to be viewed. The target itself is the top face of the inner cube.

occlusion for $\mathcal{S}(\mathit{TaskInterval}, \mathit{RobotI})$ without explicitly computing $\mathcal{S}(\mathit{TaskInterval}, \mathit{RobotI})$. In figure 4 we show a discrete approximation to the volume swept out by Robot I during its task (i.e. $\mathcal{S}(\mathit{TaskInterval}, \mathit{RobotI})$). The volume of occlusion resulting from this motion is shown in figure 5. The volume of occlusion resulting from the walls of the part (i.e. due to self-occlusions) is shown in figure 3. These two volumes were unioned to form the total volume of occlusion.

An approximation to the workspace of *Robot II*, the camera-carrying robot, (called the robot's *reachability* volume) was generated. The total occlusion volume was subtracted from this reachability volume giving the reachable/visible volume. This volume, which contains all points in space where the robot can position the camera such that the target can be seen without occlusion, was used in the optimization stage of MVP in order to compute a viewpoint.

Since MVP was unable to find a valid viewpoint for the entire task, the temporal interval search was used to find subintervals for which we can find valid viewpoints. Instead of recomputing the swept volumes for each subinterval examined, the discrete approximation allows us to union the appropriate subset of volumes of occlusion. The subintervals found for this task are shown in figures 6 and 7. The generated volumes of occlusion due to the robot's motion during each sub-interval are shown in figures 8 and 9. These volumes were again unioned with the self-occlusion volume and subtracted from the reachability volume forming the volumes of reachability/visibility shown in figure 10 and 11. These volumes were used in the optimization, and MVP was able to compute a viewpoint for each interval. Simulated views from these viewpoints are shown in figures 12 and 13.

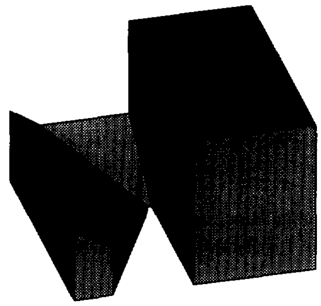

Figure 3: Volume of occlusion caused by other features on the object itself (i.e. self-occlusions).

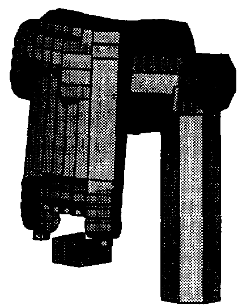

Figure 4: Swept Volume showing the robot's motion over the entire task.

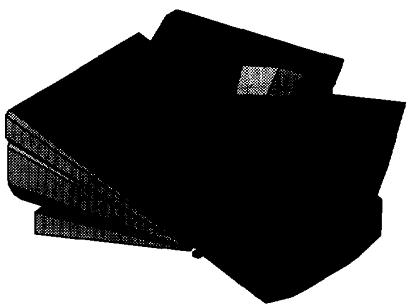

Figure 5: Volume of occlusion caused by the robot's motion during the entire task.

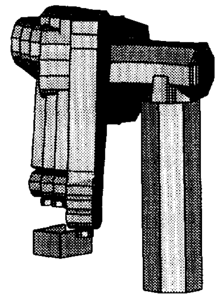

Figure 6: First task interval.

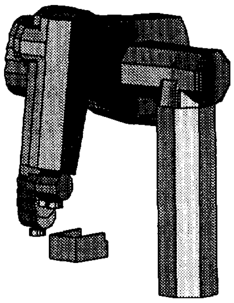

Figure 7: Second task interval.

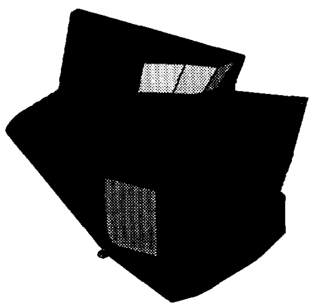

Figure 8: Occlusion due to the robot's motion during the first task interval, shown with object.

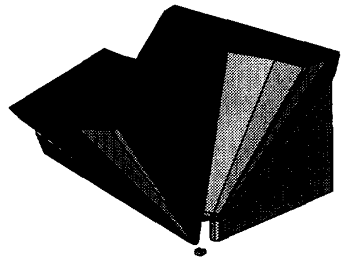

Figure 9: Occlusion due to the robot's motion during the second task interval, with object.

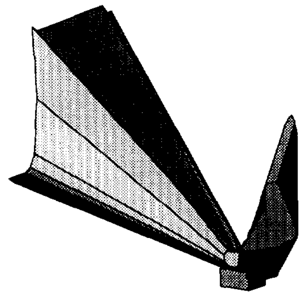

Figure 10: Intersection of reachable and visible volumes for first task interval

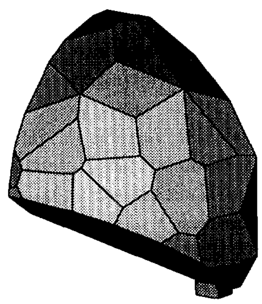

Figure 11: Intersection of reachable and visible volumes for second task interval

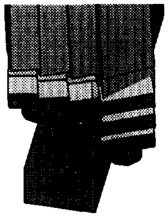

Figure 12: Simulated view from first computed viewpoint.

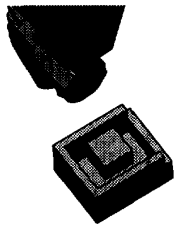

Figure 13: Simulated view from second computed viewpoint.

7 Conclusion

In conclusion, we have successfully extended our MVP system to plan sensor locations in a time-varying environment. The use of swept volumes which provides a useful way to extend static planning problems to dynamic domains. We have presented a convenient way to recover enough temporal information from swept volumes to use them in planning tasks. Our immediate research plans are to bring the results of this paper into our laboratory and execute the task with the planned viewpoints. Also, we will be examining the alternative sweeping techniques presented to see if they offer any performance improvements.

There are several open issues in dynamic sensor planning. There is work to be done in computational geometry to characterize the changes in a volume of occlusion as the target and occluding bodies move with respect to each other. A similar characterization of how the optical constraints vary with the target's motion is also important. Finally, it is hoped that these various characterizations can be combined to plan a continuous path through the sensor's parameter-space, rather than computing a series of viewpoints and critical times. This would allow more useful solutions to be found to dynamic sensor planning problems.

References

[1] Steven Abrams and Peter K. Allen. Sensor planning in an active robotic work cell. In *DARPA 1992 Image Understanding Workshop. Also in Proceedings SPIE Intelligent Robotic Systems Conference on Sensor Fusion IV: Control Paradigms and Data Structures*, Boston, MA, November 1991.

[2] Stephen Alan Cameron. *Modelling Solids in Motion.* PhD thesis, Department of Computer Science, University of Edinburgh, 1984.

[3] C. K. Cowan and A. Bergman. Model based synthesis of sensor locations. In *Proceedings 1989 IEEE International Conference on Robotics and Automation*, December 1989.

[4] S. A. Hutchinson and A. C. Kak. Planning sensing strategies in robot work cell with multi-sensor capabilities. In *Proceedings 1989 IEEE International Conference on Robotics and Automation*, December 1989.

[5] K. Ikeuchi and T. Kanade. Modeling sensors: Towards automatic generation of object recognition programs. *Computer Vision, Graphics, and Image Processing*, 48:50–79, 1989.

[6] James Korein. *A Geometric Investigation of Reach.* MIT Press, Cambridge, MA, 1985.

[7] Konstantinos Tarabanis. *Sensor Planning and Modeling for machine vision tasks.* PhD thesis, Department of Computer Science, Columbia University, October 1991.

[8] Konstantinos Tarabanis, Roger Y. Tsai, and Steven Abrams. Planning viewpoints that simultaneously satisfy several feature detectability constraints for robotic vision. In *Proceedings Fifth Inernational Conference of Advanced Robotics*, 1991.

[9] Kostantinos Tarabanis, Roger Y. Tsai, and Peter K. Allen. Automated sensor planning and modeling for robotic vision tasks. In *Proceedings 1991 IEEE International Conference on Robotics and Automation*, April 1991.

[10] John D. Weld and Ming C. Leu. Geometric representation of swept volumes with application to polyhedral objects. *International Journal of Robotics Research*, 9(5):105–117, October 1990.