Visually Built Task Models for Robot Teams in Unstructured Environments

Vivek A. Sujan and Steven Dubowsky
(vasujan|dubowsky@mit.edu}
Department of Mechanical Engineering
Massachusetts Institute of Technology
Cambridge, MA 02139

Abstract

In field environments it is not usually possible to provide robotic systems with valid geometric models of the task and environment. The robot or robot teams will need to create these models by performing appropriate sensor actions. Here, an algorithm based on iterative sensor planning and sensor redundancy is proposed to enable them to efficiently build 3D models of the environment and task. The method assumes stationary robotic vehicles with cameras carried by articulated mounts. The algorithm uses the measured scene information to find new camera mount poses based on information content. Issues addressed include model-based multiple sensor data fusion, and uncertainty and vehicle suspension motion compensation. Simulations show the effectiveness of this algorithm.

1. Introduction

An important goal of robotics research is to develop mobile robot teams that can work cooperatively in unstructured field environments, such as shown conceptually in Figure 1 [2, 7]. Potential tasks include explosive ordinance removal, de-mining and hazardous waste handling, exploration/development of space, environment restoration, and construction [2, 7, 15].

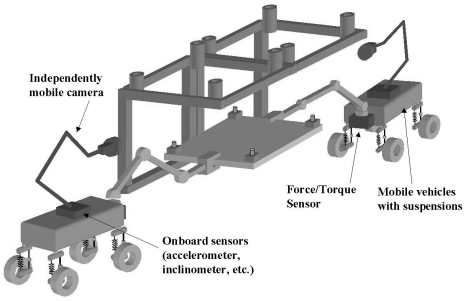


Figure 1: Representative physical system

The control of such systems typically requires models of the environment and task. In unstructured field environments it is often not possible to have such a-priori models. In such cases, the robot needs to construct these from sensory information. A number of problems can make this non-trivial. These include the uncertainty of the task in the environment, location and orientation uncertainty in the individual robots, and occlusions (due to obstacles, work piece, other robots). If the systems are equipped with cameras mounted on articulated mounts, intelligent planning of the camera motion can alleviate problems of the occlusions, providing an accurate geometrical model of the task and environment. If the system consists of more than one robot, planning the behavior of these multi-information sharing systems can further improve the system performance.

Previous work in planning of visual sensing strategies can be divided into two areas [12, 18]. One of these is concerned with sensor positioning—placing a sensor so that it can best observe some feature and selecting a sensing operation which will prove the most useful in object identification and localization. Researchers have limited their work to model-based approaches, requiring previously known environments [4, 6, 9]. Target motions (if any) are assumed to be known. Brute force search methods divide the entire view volume into grids, octrees, constraint sets, and search algorithms for optimum sensor location, are applied [5, 6, 12]. Additionally, they require a-priori knowledge of object/target models [18]. Such methods can be effective but are computationally expensive and not practical for many real field environments, where occlusions and measurement uncertainties are present.

The other direction of research in planning of sensors is sensor data fusion—combining complementary data from either different sensors or different sensor poses to get an improved net measurement [17, 18]. The main advantages of multi-sensor fusion are the exploitation of data redundancy and complementary information. Common methods for sensor data fusion are primarily heuristic (Fuzzy logic) or statistical in nature (Kalman and Bayesian filters) [3, 12, 14].

For environment and target model building both areas play key roles. However, little work has been done in effectively combining the capabilities of sensor placement planning and sensory fusion, to develop a sensing strategy for model building to be used by robots and robot teams in unstructured environments. Some studies have considered cooperative robot mapping of the environment [8, 13, 19]. Novel methods of establishing/identifying landmarks and dealing with cyclic environments have been introduced for indoor environments [8, 19]. In some cases, observing robot team members as references to develop accurate maps is required [13]. Mapping has been done in sequential brute force fashion [13, 19]. Researchers have addressed the concept of map building using a single mobile vision system [1, 4, 10]. Often, sensor models and data uncertainty are not fully considered [1, 10, 11] or exploration schemes are not developed [3, 14]. Structured environment and partial knowledge assumptions are also made [4, 14].

This paper proposes an environment and task model building algorithm, to overcome the uncertainties in robot and camera location and orientation, for robot teams cooperatively working in an unstructured field environment. It is assumed that dimensional geometric information is relevant and required for robots to perform their operations, such as the construction of field

facilities. It is also assumed that the system consists of two (or more) mobile robots working in an unknown environment (such as constructing a planetary structure—see Figure 2). There are no physical interactions between the robots. The vehicles and target are static. Each has a 3D vision system mounted on an articulated arm. Sensing and sensor placement is limited, resulting in occlusions and uncertainties. Again, the objective is to efficiently build a geometrically consistent dimensional model of the environment and target, available to all robots, to allow for tasks to be performed. This involves locating the robots and mapping a region around a target with respect to some target fixed reference frame. The key idea is that the algorithm builds an environment and task model by fusing the data available from each individual robot, providing both improved accuracy as well as knowledge of regions not visible by all robots. Using this algorithm, the individual robots can also be positioned "optimally" with respect to the target [20]. However, this is beyond the scope of this paper.

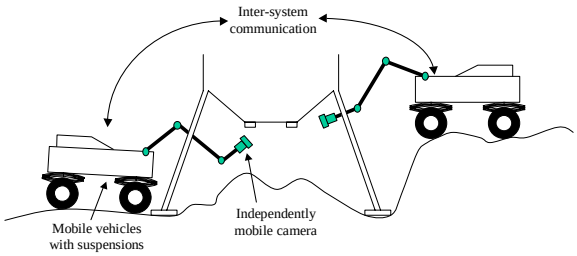


Figure 2: Cooperative mapping by robots

2. Algorithm description

2.1. Overview

The first step in cooperative model building is to visually construct a model of the local environment, including the locations of the task elements and the robots themselves. We assume that only the geometry of the task elements (such as the parts of a solar panel that needs to be assembled [7]) is well known. Obstacles and robot positions are not known.

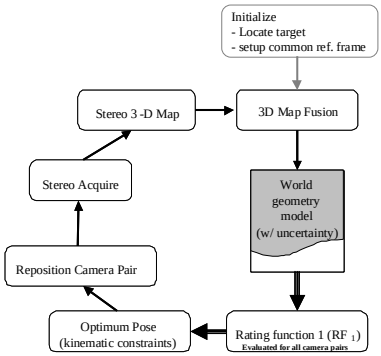


Figure 3: Outline of model building and placement algorithm

Figure 3 outlines the map building algorithm. The algorithm consists of two major parts. In the first part, the articulated cameras cooperatively scan the region around a target generating a 3D geometric model, so that the robots can locate themselves and the obstacles in the target reference frame. The second stage consists of using this model to find an optimum pose for the multiple camera systems to view the target(s). The 3D map is modeled as a probabilistic discretized occupancy grid. Every voxel in the map has a value for probability-of-occupancy that ranges from 0 (empty) to 1 (occupied). A value of 0.5 indicates maximum uncertainty in occupancy of the voxel. The process is initialized by visually finding the target and robots in a common reference frame. This is done by "looking around" and matching the known target element geometric model with visual data. Next, a new camera pose is found for each of the cameras by defining and evaluating a rating function (RF) over the known environment map subject to kinematic constraints of the sensor placement mechanisms for the individual robots. Then, the cameras move to their new poses and acquire 3D data. Based on the camera mount kinematics, the motions of the cameras are known. Note, it is assumed that the vehicles don't move, as large camera motions would be difficult to measure. Small motions of the robot base (due to suspension compliance) and errors in camera mounts lead to additional uncertainties. These are accounted for by measuring common features during the camera motion (section 2.5). Finally, the new data and its associated uncertainty are fused with the current environment map resulting in an updated probabilistic environment map.

2.2. Algorithm initialization

As described above, a common target is to be located to establish a common inertial reference frame between the robots and the environment. Searching for the target by moving the robot cameras can be done in many ways (depending on the target properties), such as exhaustive raster scanning, random walking, tracking "space filling curves", and model-based image understanding methods [12, 18]. In this study, camera positioning for target searching is done in the same way as camera positioning for environment model building (described in sections 2.4).

2.3. Data modeling and fusion

At any time, the cameras on each mobile robot can only observe a small part of their environment. However, measurements obtained from multiple viewpoints can provide reduced uncertainty, improved accuracy, and increased tolerance in estimating the location of the observed object [17]. To fuse multiple range measurements of a feature by sensors, a statistical model of sensor uncertainty is employed (see Figure 4). Current and previous range sensor measurements and their uncertainty models can be integrated to give an updated probabilistic geometric model of the environment.

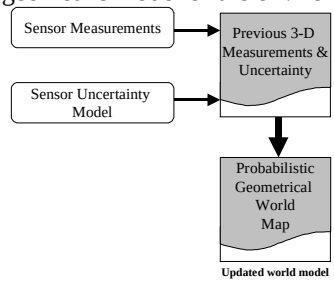


Figure 4: 3-D range measurement fusion with sensor uncertainty

A single range observation of a point ($\bar{x}$) is modeled as a 3-D Gaussian probability distribution centered at $\bar{x}$, based on two important observations. First, the use of the

mean and covariance of a probability distribution function is a reasonable form to model sensor data and is a second order linear approximation [17]. This linear approximation corresponds to the use of a Gaussian (having all higher moments of zero). Second, based on the central limit theorem, the sum of a number of independent variables has a Gaussian distribution regardless of their individual distributions.

The standard deviations along the three axes of the distribution correspond to estimates of the uncertainty in the range observation along these axes. These standard deviations are a function of intrinsic sensor parameters (such as camera lens shape accuracy) as well as extrinsic sensor parameters (such as the distance to the observed point or feature). This model can be theoretically approximated as [20]:

$$\begin{aligned}\sigma_{x,y,z} &= f(\text{extrinsic parameters, intrinsic parameters}) \\ &\approx S \cdot T_{x,y,z} \cdot d^{n}\end{aligned} \quad (1)$$

where S is an intrinsic parameter uncertainty constant, $T_{x,y,z}$ is an extrinsic parameter uncertainty constant, d is the distance to the feature/environment point, and n is a constant (typically 2). Provided two observations are drawn from a normal distribution, the observations can be merged into an improved estimate by multiplying the distributions. Since the result of multiplying two Gaussian distributions is another Gaussian distribution, the operation is symmetric, associative, and can be used to combine any number of distributions in any order. The canonical form of the Gaussian distribution in n dimensions depends on the standard distributions, $\sigma_{x,y,z}$, a covariance matrix (C) and the mean ($\bar{x}$) [17]:

$$p(\vec{x}') = \frac{1}{(2\pi)^{n/2}\sqrt{|C|}} \exp\left(-\frac{1}{2}(\vec{x}-\vec{x}')^{T} C^{-1} (\vec{x}-\vec{x}')\right), \text{ where } C = \begin{bmatrix} \sigma_x^2 & \rho\sigma_x\sigma_y & \rho\sigma_x\sigma_z \\ \rho\sigma_x\sigma_y & \sigma_y^2 & \rho\sigma_y\sigma_z \\ \rho\sigma_x\sigma_z & \rho\sigma_y\sigma_z & \sigma_z^2 \end{bmatrix} \quad (2)$$

where the exponent is called the Mahalanobis distance. For un-correlated measured data $\rho=0$. The formulation in Equation 2 is in the spatial coordinate frame. However, all measurements are made in the camera (or sensor) coordinate frame. This problem is addressed through a transformation of parameters from the observation frame to the spatial reference frame as follows:

$$C_{\text{transformed}} = R(-\bar{\theta})^{T} \cdot C \cdot R(-\bar{\theta}) \quad (3)$$

where $R(\boldsymbol{\theta})$ is the rotation matrix between the two coordinate frames. The angle of the resulting principal axis can be obtained from the merged covariance matrix:

$$C_{\text{merged}} = C_1\left(1 - \frac{C_1}{C_1 + C_2}\right) \quad (4)$$

where C_i is the covariance matrix associated with the i^{th} measurement. Additionally, a translation operation is applied to the result from Equation 2, to bring the result into the spatial reference frame.

2.4. Definition of the rating function (RF)

A rating function is used to determine the next pose of the camera from which to look at the unknown environment. The aim is to acquire new information of the environment that would lead to a more detail and more extensive environment map. In selecting this new camera state the following four constraints are considered:

(i) *Goal configuration is collision free*—from the probabilistic geometric environment model, (x,y,z) locations with $P_{x,y,z} < P_{empty} = 0.05$ (2σ) are considered as unoccupied. Such points form candidate configuration space camera pose coordinates.

(ii) *Goal reached by a collision free path*—this is a function of the camera manipulator kinematics and the known environment model.

(iii) *Goal configuration should not be far from the current one*—a Euclidean metric in configuration space, with individual weights α_i on each degree of freedom of the camera pose $\bar{c}$, is used to define the distance moved by the camera:

$$d = \left(\sum_{i=1}^{n} \alpha_i (c_i - c_i')^2\right)^{1/2} \quad (5)$$

where $\bar{c}$ and $\bar{c}'$ are vectors of the new and current camera poses respectively.

(iv) *Measurement at the goal configuration should maximize information intake*—Specifically, the new information (NI) is equal to the expected information of the unknown/partially known region viewed from the camera pose under consideration. This is based on the known obstacles from the current environment model, the field of view of the camera (see Figure 5) and a framework for entropic thresholding of information. Shannon showed that a definition of entropy, similar in form to a corresponding definition in statistical mechanics, can be used to measure the information gained from the selection of a specific event among an ensemble of possible events [16]. This entropy function, H, can be represented as:

$$H(q_1, q_2, ..., q_n) = -\sum_{k=1}^{n} q_k \ln q_k \quad (6)$$

where q_k represents the probability of occurrence for the k^{th} event, and uniquely satisfies the following three conditions [16]:

- $H(q_1, q_2, \ldots, q_n)$ is a maximum for $q_k=1/n$ for k=1…n. This implies that a uniform probability distribution possesses the maximum entropy.
- $H(AB)=H(A)+H_A(B)$ where A and B are two finite schemes. H(AB) represents the total entropy of schemes A and B and $H_A(B)$ is the conditional entropy of scheme B given scheme A.
- $H(q_1, q_2, \ldots, q_n, 0) = H(q_1, q_2, \ldots, q_n)$ or any event with zero probability of occurrence in a scheme does not change the entropy function.

Shannon's emphasis was in describing the information content of 1-D signals. In 2-D the gray level histogram of an image can be used to define a probability distribution:

$$q_i = f_i / N \text{ for } i = 1...N_{\text{gray}} \quad (7)$$

where f_i is the number of pixels in the image with gray level i, N is the total number of pixels in the image, and N_{gray} is the number of possible gray levels. With this definition, the entropy of an image for which all the q_i are

the same—corresponding to a uniform gray level distribution or maximum contrast—is a maximum. The less uniform the histogram, the lower the entropy.

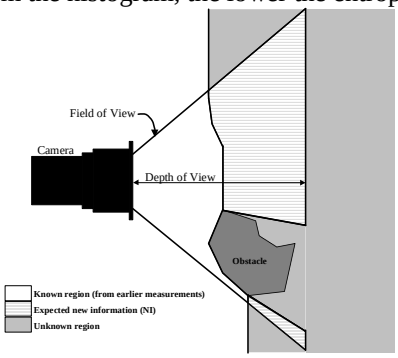


Figure 5: Evaluation of expected new information

It is possible to extend this idea of entropy to a 3-D signal—the environment model. In such an instance the scene probability distribution for entropy (information) analysis is still given by Equation 7. However, N is the maximum number of voxels visible by the vision system (limited by the depth of field and the field of view), and f_i is the number of voxels in the scene with gray level i. The possible gray values are defined as follows. For any previously sampled spatial voxel, a gray (probabilistic) occupancy value between 0 and 1 is found. Next the value, $p(\underline{x})$, is modified as follows:

$$\text{stretching}: p'(\bar{x}) = \begin{cases} \dfrac{1}{1-p(\bar{x})} \cdot \dfrac{1}{d_{voxel}} & \forall\, p(\bar{x}) < 0.5 \\ \dfrac{1}{p(\bar{x})} \cdot \dfrac{1}{d_{voxel}} & \forall\, p(\bar{x}) \geq 0.5 \end{cases} \quad (8a)$$

$$\text{scaling}: p''(\bar{x}) = \begin{cases} \dfrac{p'(\bar{x})-1}{2} & \forall\, p(\bar{x}) < 0.5 \\ 1-\dfrac{p'(\bar{x})-1}{2} & \forall\, p(\bar{x}) \geq 0.5 \end{cases}$$

where d_{voxel} is the Euclidean distance of the voxel from the camera coordinate frame. This process causes regions with probability densities closer to 0 or 1 (regions of most certainty) to have a reduced effect on the new information expected. Regions that have a probability density closer to 0.5 (regions of least certainty of occupancy) are stretched out in the scene probability distribution, thus increasing the new expected information associated with those regions. Additionally, for all unknown/unsampled voxels a gray value between 1 and 2 is defined:

$$p(\bar{x}) = 1 + \frac{d_{voxel}}{d_{\max}} \quad (8b)$$

where d_{max} is the maximum distance of any voxel in the camera field of view to the camera (equal to the depth of field). A uniform discretization of this range of gray values may be performed to define N_{gray}. With these definitions q_k (Equation 7) is evaluated and the results applied to Equation 6 resulting in a metric for new information (NI). Note that by applying the three conditions described above, this definition for NI does behave in an intuitively correct form. For example, for a given camera pose, if the field of view is occluded then NI decreases. If every point in the field of view is known and is empty then NI=0. NI increases as the number of unknowns in the field of view increases. Further, Equation 8a results in increasing the new information expected with regions that are known with median probabilistic values i.e. values that indicate with least amount of certainty whether a voxel is occupied or not. On the other hand, regions with high probabilistic values for occupancy result in reduced associated information.

To plan the motion consistently, constraints (iii) and (iv) are merged into a unique rating function (RF):

$$RF = (NI - K \,.\, d^n) \,.\, (1 - P_{x,y,z}) \quad (9)$$

where K, n are scaling constants. Shorter distances exhibit a higher rating. This rating function can be evaluated and optimized to find the next most promising camera configuration from which to make future measurements of the environment. Although this choice of rating function is somewhat arbitrary, good results were obtained. Additional constraints can also be accommodated.

2.5. Suspension motion correction

A final step is to identify the motion of the camera to allow for data fusion. This process eliminates manipulator positioning errors and vehicle suspension motions. A single spatial point in the base frame, $\bar{r}_i$, is related to the image point (u_i, v_i) by the 4x4 transformation matrix $\mathbf{g}_{01}$ (see Figure 6).

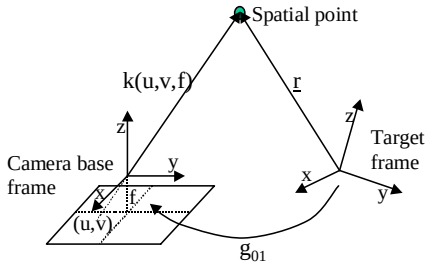


Figure 6: Relationship of camera and target frames

For motion calibration we need to identify $\mathbf{g}_{01}$:

$$\begin{bmatrix} k_i u_i \\ k_i v_i \\ k_i f \\ 1 \end{bmatrix} = \mathbf{g}_{01} \cdot \bar{r}_i = \begin{bmatrix} [\mathbf{R_{01}}]_{3x3} & \bar{x}_{3x1} \\ \bar{0} & 1 \end{bmatrix} \cdot \begin{bmatrix} r_i^x \\ r_i^y \\ r_i^z \\ 1 \end{bmatrix} \quad (10)$$

where $\mathbf{R}_{01}$ is the rotational matrix, $\bar{x}$ is the translation vector, f is the camera focal length, and k_i is a scaling constant. For computational reasons it is more convenient to treat the 9 rotational components of $\mathbf{R}_{01}$ as independent (rather than a transcedental relation of 3 independent parameters). Each spatial point gives 3 algebraic equations, but also introduces a new variable, k_i—multiplicative constant to extend the i^{th} image point vector $(u,v,f)_i$ to the i^{th} spatial point in the camera coordinate frame. k_i may be found from the disparity pair of the stereo images. For n points we have:

$$\mathbf{u} = \mathbf{g}_{01}\mathbf{r} \Rightarrow \begin{bmatrix} k_1u_1 & k_2u_2 & \cdots & k_nu_n \\ k_1v_1 & k_2v_2 & \cdots & k_nv_n \\ k_1f & k_2f & \cdots & k_nf \\ 1 & 1 & \cdots & 1 \end{bmatrix} = \mathbf{g}_{01} \begin{bmatrix} r_1^x & r_2^x & \cdots & r_n^x \\ r_1^y & r_2^y & \cdots & r_n^y \\ r_1^z & r_2^z & \cdots & r_n^z \\ 1 & 1 & \cdots & 1 \end{bmatrix} \quad (11)$$

This set of linear equations can be readily solved using conventional techniques. A least mean square error

solution is given by:

$$\mathbf{g_{01}} = \mathbf{u}\left(\mathbf{r}^{\mathbf{T}}\mathbf{r}\right)^{-1}\mathbf{r}^{\mathbf{T}} \tag{12}$$

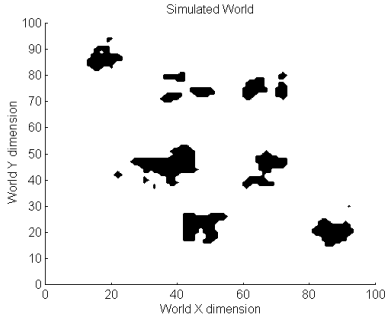
The rotation matrix, $\mathbf{R}_{01}$, and the translation vector, $\underline{\mathbf{x}}$, of the camera frame with respect to the base frame are extracted directly from this solution of $\mathbf{g}_{01}$. However, for real measured data and associated uncertainty, a larger number of scene points are required to more correctly identify the geometric transformation matrix, $\mathbf{g}_{01}$. Given the $(i+1)^{st}$ scene and image point, from Equation 12 $\mathbf{R}_{i+1}$ and $\underline{\mathbf{x}}_{i+1}$ can be obtained. A recursive method can be used to determine the mean and covariance of $\underline{\mathbf{x}}$ and $\mathbf{R}_{01}$ based on the previous i measurements as follows:

$$\begin{aligned}
\hat{\bar{x}}_{i+1} &= \frac{\left(i\hat{\bar{x}}_i + \bar{x}_{i+1}\right)}{i+1} \\
C_{i+1}^{\bar{x}} &= \frac{iC_i^{\bar{x}} + \left[\bar{x}_{i+1} - \hat{\bar{x}}_{i+1}\right]\left[\bar{x}_{i+1} - \hat{\bar{x}}_{i+1}\right]^T}{i+1} \\
\hat{\mathbf{R}}_{i+1}^{(l,m)} &= \frac{\left(i\hat{\mathbf{R}}_i^{(l,m)} + \mathbf{R}_{i+1}^{(l,m)}\right)}{i+1} \\
C_{i+1}^{\mathbf{R}(l,m)} &= \frac{iC_i^{\mathbf{R}(l,m)} + \left[\mathbf{R}_{i+1}^{(l,m)} - \hat{\mathbf{R}}_{i+1}^{(l,m)}\right]\left[\mathbf{R}_{i+1}^{(l,m)} - \hat{\mathbf{R}}_{i+1}^{(l,m)}\right]^T}{i+1}
\end{aligned} \tag{13}$$

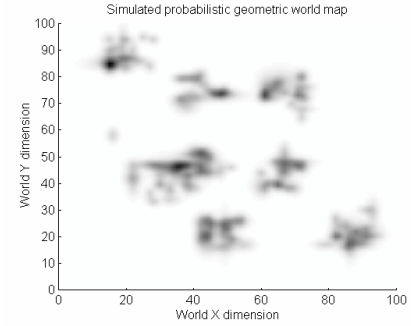
Finally, the issue of obtaining appropriate spatial points for vehicle motion compensation is addressed. Spatial points are obtained by maintaining a finite set of fiducials that are tracked during map building and visible by the cameras. As the camera moves, the fiducials move relative to the camera, eventually moving out of the camera view. This requires methods to track and identify new fiducials. First, for the purposes of this study fiducials are tracked using a computationally fast region growing method. Second, new fiducials are selected from the probabilistic environment model based on the degree of certainty with which a sampled point is known. Specifically, all local peaks in the probabilistic geometric environment map (potential fiducials) are identified. Next, at each local peak a process called spherical expansion is performed. In spherical expansion, using local gradient decent on neighboring voxels, the largest spherical region around a local peak, beyond which the voxel values increase, is found. Lastly, expanded spheres are scored based on the product of their radii and magnitude of local peak. These scores are normalized based on their distance to the current camera position. Higher scoring peaks form better fiducials and are selected accordingly. Although, alternative scoring functions may be employed, this simple one proves highly effective. Note that, by knowing the camera position and the camera arm kinematics, the robot base position can be easily extrapolated.

3. Simulation results

Results using the rating function, to explore a planar environment and develop a probabilistic geometric environment model, are given here. Figure 7 shows the results obtained of scanning a planar environment of random obstacles. Two vision systems fuse 200 samples each to give the probabilistic map seen in Figure 7(b). By increasing the number of scans taken, the uncertainty in this probabilistic map decreases.



(a) Simulated environment



(b) Probabilistic environment map after 400 scans

Figure 7: Mapping a simulated planar environment

Table 1: Comparison of explored space after 32 scans

Search method	Farthest visual point from start	Average radial distance from start	σ of viewed space
Random (m=10)	54.89	26.74	11.48
Random (m=50)	47.34	23.79	9.91
Random (m=100)	46.69	23.57	9.17
Random (m=200)	46.01	22.81	9.42
Random (m=300)	36.62	20.69	7.85
Exhaustive	34.59	19.23	7.52

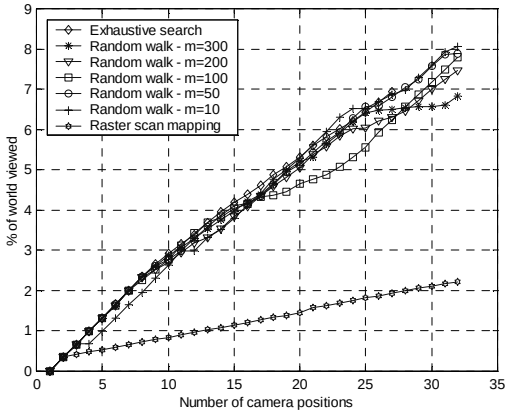


Figure 8: Percentage of environment viewed

The rating function (RF) cannot be optimized analytically. In practice, finding an optimum value for RF requires exhaustive searching though the entire known configuration space—a process that takes 0(n) time, where n is the number of discrete points in the configuration space. One way to reduce the search time is

to employ a finite random selection of goal configurations. For m possible configurations, this process takes 0(m) time—m is a constant. Thus, while the best goal configuration would be the one maximizing RF, any configuration with a high value for RF should suffice. Such a configuration can be found with reasonable effort. For comparison, results from random sample selection using 10, 50, 100, 200, and 300 points are presented along with an exhaustive search, in Figure 8 and table 1. Note that as the number of search points in the random selection increases, the explored/viewed space grows more uniformly (measured as the standard deviation of the radius of every point in the viewed environment space). This reaches a threshold as the search becomes more exhaustive in nature. Figure 8 shows the percentage increase of the environment viewed as a function of the number of scans. From this it appears that the effects of random walk searches produce equivalent results as an exhaustive search. Additionally, for comparison Figure 8 presents the results of modeling the environment using a conventional raster scan (where the next viewing position is selected sequentially from the available poses of the known environment). Clearly, there is significant decrease in performance efficiency. The specific numbers presented here are a function of camera properties (such as the FOV and the DOF) and the environment obstacles, and should be used to reflect the trend, not the exact behavior.

4. Conclusions

In field environments it is often not possible to provide robotic teams with detailed a priori environment and task models. In such unstructured environments, cooperating robots will need to create a dimensionally accurate 3-D geometric model by performing appropriate sensor actions. However, uncertainties in robot locations and sensing limitations/occlusions make this difficult. A new algorithm based on iterative sensor planning and sensor redundancy is proposed to build a geometrically consistent dimensional map of the environment for mobile robots that have articulated sensors. This algorithm is unique in that it uses a metric of the quality of information previously obtained by the sensors to find new viewing positions for the cameras. Simulations show promising results.

5. References

1. Asada, M. *Map building for a mobile robot from sensory data*. IEEE Transactions on Systems, Man, and Cybernetics. Vol. 37, no. 6, November/December 1990.

2. Baumgartner, E.T., P. S. Schenker, C. Leger, and T. L. Huntsberger. *Sensor-fused navigation and manipulation from a planetary rover*. Proceedings SPIE Symposium on Sensor Fusion and Decentralized Control in Robotic Systems, Vol. 3523, Boston, MA, Nov 1998.

3. Betge-Brezetz, S., Hebert, P., Chatila, R., and Devy, M. *Uncertain map making in natural environments.* Proceedings of the 1996 IEEE International Conference on Robotics and Automation, Minneapolis, Minnesota, April, 1996.

4. Burschka, D. Eberst, C. and Robl, C. *Vision based model generation for indoor environments.* Proceedings of the 1997 IEEE International Conference on Robotics and Automation, Albuquerque, New Mexico, April, 1997

5. Connolly, C.I. *The determination of the next best views.* Proceedings of the IEEE International Conference on Robotics and Automation, pp. 432-435, 1985.

6. Cowan, G.K. and Kovesi, P.D. *Automatic sensor placement from vision task requirements.* IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 10, no. 3, pp. 407-416, May 1988.

7. Huntsberger, T.L. *Autonomous multi-rover system for complex planetary retrieval operations.* Proceedings. SPIE Symposium on Sensor Fusion and Decentralized Control in Autonomous Robotic Agents, Pittsburgh, PA, Oct 1997, pp. 220-229.

8. Jennings, C., Murray, D., and Little, J. *Cooperative robot localization with vision-based mapping.* Proceedings of the 1999 IEEE International Conference on Robotics and Automation, Detroit, Michigan, May 1999.

9. Kececi, F., Tonko, M., Nagel, H.H., and Gengenbach, V. *Improving visually servoed disassembly operations by automatic camera placement.* Proceedings of the 1998 IEEE International Conference on Robotics and Automation, Leuven Belgium. May 1998.

10. Kruse, E., Gutsche, R., and Wahl, F.M. *Efficient, iterative, sensor based 3-D map building using rating functions in configuration space.* Proceedings of the 1996 IEEE International Conference on Robotics and Automation, Minneapolis, Minnesota, April, 1996.

11. Lumelsky, V., Mukhopadhyay, S. and Sun, K. *Sensor-based terrain acquisition: the "sightseer" strategy.* Proceedings of the 28th Conference on Decision and Control. Tampa, Florida. December 1989.

12. Luo, R.C. and M.G. Kay. *Multisensor integration and fusion in intelligent systems.* IEEE Transactions on Systems, Man, and Cybernetics, Vol. 19, No. 5, September, 1989.

13. Rekleitis, I., Dudek, G., and Milios, E. *Multi-robot collaboration for robust exploration.* Proceedings of the 2000 IEEE International Conference on Robotics and Automation, San Francisco, CA. April, 2000.

14. Repo, T. and Roning, J. *Modeling structured environments by a single moving camera*. Proceedings. Second International Conference on 3-D Digital Imaging and Modeling, 1999. On page(s): 340 - 347,4-8 Oct. 1999

15. Shaffer, G.; Stentz, A. *A robotic system for underground coal mining.* Proceedings of the IEEE International Conference on Robotics and Automation, 1992. Pages(s): 633 -638 vol.1

16. Shannon, C. E. *A mathematical theory of communication.* The Bell System Technical Journal, vol. 27, pp. 379-423 and 623-656, July and October, 1948.

17. Smith, R.C. and Cheeseman, P. *On the representation and estimation of spatial uncertainty*. International Journal of Robotics Research. 5(4): 56-68

18. Tarabanis, K.A., Allen, P.K. and Tsai, R.Y. *A survey of sensor planning in computer vision.* IEEE Transactions on Robotics and Automation, Vol. 11 no. 1, Pp. 86-104. February, 1995.

19. Thrun, S., Burgard, W., and Fox, D. *A real-time algorithm for mobile robot mapping with applications to multi-robot and 3D mapping.* Proceedings of the 2000 IEEE International Conference on Robotics and Automation. San Francisco, CA, April, 2000.

20. Sujan, V.A. *Compensating for Uncertainty in the Control of Cooperative Robots in Field Environments.* Ph.D. Thesis, Department of Mechanical Engineering, Massachusetts Institute of Technology, June 2002.