Journal of Intelligent and Robotic Systems (2005) 43: 217–253
DOI:10.1007/s10846-004-3063-y

Intelligent and Efficient Strategy for Unstructured Environment Sensing Using Mobile Robot Agents

VIVEK A. SUJAN
Air Handling and Combustion Control Division, Cummins Engine Company, Columbus, IN 47201, U.S.A.; e-mail: vivek.a.sujan@cummins.com

MARCO A. MEGGIOLARO
Department of Mechanical Engineering, Pontifical Catholic University of Rio de Janeiro, Rio de Janeiro, RJ, Brazil 22453-900; e-mail: meggi@alum.mit.edu

(Received: 9 December 2003; in final form: 19 July 2004)

Abstract. In field environments it is not usually possible to provide robots in advance with valid geometric models of its task and environment. The robot or robot teams need to create these models by scanning the environment with its sensors. Here, an information-based iterative algorithm to plan the robot's visual exploration strategy is proposed to enable it to most efficiently build 3D models of its environment and task. The method assumes mobile robot (or vehicle) with vision sensors mounted at a manipulator end-effector (eye-in-hand system). This algorithm efficiently repositions the systems' sensing agents using an information theoretic approach and fuses sensory information using physical models to yield a geometrically consistent environment map. This is achieved by utilizing a metric derived from Shannon's information theory to determine optimal sensing poses for the agent(s) mapping a highly unstructured environment. This map is then distributed among the agents using an information-based relevant data reduction scheme. This method is particularly well suited to unstructured environments, where sensor uncertainty is significant. Issues addressed include model-based multiple sensor data fusion, and uncertainty and vehicle suspension motion compensation. Simulation results show the effectiveness of this algorithm.

Key words: visual mapping, cooperative robots, information theory, unstructured environments, data fusion.

1. Introduction

An important goal of robotics research is to develop mobile robot teams that can work cooperatively in unstructured field environments (Baumgartner et al., 1998; Huntsberger et al., 2000). Figure 1 shows an example of a representative system and task, however several other tasks and systems can be considered. Potential tasks include explosive ordinance removal, de-mining and hazardous waste handling, exploration/development of space, environment restoration, and construction (Baumgartner et al., 1998; Huntsberger et al., 2001). For instance, space and planetary robotic missions will require robot scouts to lead the way, by exploring, mapping, seeking or extracting soil and rock samples and eventually constructing

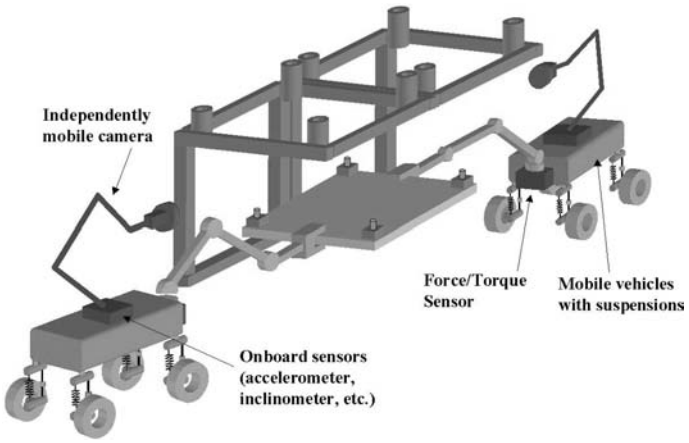


Figure 1. Representative physical system.

facilities in complex terrains. Multiple cooperating robots will be required to set up surface facilities in challenging terrain for in-situ measurements, communications, and to pave the way for human exploration of planetary surfaces. This will require the handling of relatively large objects, such as deploying of solar panels and sensor arrays, anchoring of deployed structures, movement of rocks, and clearing of terrain.

The use of robot teams working cooperatively to acquire and share data has been proposed to succeed in those missions (Huntsberger et al., 2001; Pirjanian et al., 2001; Schenker et al., 2001; Sujan and Dubowsky, 2002; Trebi-Ollennu et al., 2002). 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, usually from vision systems. A number of problems can make this nontrivial. These include the uncertainty of the task in the environment, location and orientation uncertainty in the individual robots, and occlusions, e.g., due to obstacles, work piece, or other robots. If the systems are equipped with vision sensors mounted at a manipulator end-effector (eye-in-hand systems), intelligent planning of the sensor 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.

Environment mapping by mobile robots falls into the category of Simultaneous Localization and Mapping (SLAM). In SLAM a robot is localizing itself as it maps the environment. Researchers have addressed this problem for well-structured (indoor) environments and have obtained important results (Anousaki and Kyriakopoulos, 1999; Burschka et al., 1997; Castellanos et al., 1998; Choset and Nagatani, 2001; Kruse et al., 1996; Thrun et al., 2000; Tomatis et al., 2001; Victorino et al., 2000; Yamauchi et al., 1998). These algorithms have been implemented for several

different sensing methods, such as camera vision systems (Castellanos et al., 1998; Hager et al., 1997; Park et al., 1999), laser range sensors (Tomatis et al., 2001; Yamauchi et al., 1998), and ultrasonic sensors (Anousaki and Kyriakopoulos, 1999; Choset and Nagatani, 2001). Sensor movement/placement is usually done sequentially (raster scan type approach), by following topological graphs or using a variety of *greedy* algorithms that explore regions only on the extreme edges of the known environment (Anousaki and Kyriakopoulos, 1999; Choset and Nagatani, 2001; Reklaitis et al., 2000; Victorino et al., 2000; Yamauchi et al., 1998). Geometric descriptions of the environment are modeled in several ways, including generalized cones, graph models and voronoi diagrams, occupancy grid models, segment models, vertex models, convex polygon models (Choset and Nagatani, 2001). The focus of these works is accurate mapping. They do not address mapping efficiency. Researchers have addressed mapping efficiency to a limited amount (Kruse et al., 1996). However, sensing and motion uncertainties are not accounted for. They also generally assume that the environment is effectively flat (e.g., the floor of an office or a corridor) and readily traversable (i.e. obstacles always have a route around them) (Anousaki and Kyriakopoulos, 1999; Choset and Nagatani, 2001; Thrun et al., 2000; Yamauchi et al., 1998) and have not been applied to robot teams working in rough planetary environments. Also, prior work has not addressed optimizing the communication between agents for both multi-agent planning and cooperative map-building.

To achieve the localization function, landmarks and their relative motions are monitored with respect to the vision systems. Several localization schemes have been implemented, including topological methods such as generalized voronoi graphs and global topological maps (Choset and Nagatani, 2001; Tomatis et al., 2001; Victorino et al., 2000), extended Kalman filters (Anousaki and Kyriakopoulos, 1999; Park et al., 1999), and robust averages (Park et al., 1999). Although novel natural landmark selection methods have been proposed (Hager et al., 1997; Simhon and Dudek, 1998; Yeh and Kriegman, 1995), most SLAM architectures rely on identifying landmarks as corners or edges in the environment (Anousaki and Kyriakopoulos, 1999; Castellanos et al., 1998; Choset and Nagatani, 2001; Victorino et al., 2000). This often limits the algorithms to structured indoor-type environments. Others have used human intervention to identify landmarks (Thrun et al., 2000).

Some studies have considered cooperative robot mapping of the environment (Jennings et al., 1999; Reklaitis et al., 2000; Thrun et al., 2000). Novel methods of establishing/identifying landmarks and dealing with cyclic environments have been introduced for indoor environments (Jennings et al., 1999; Thrun et al., 2000). In some cases, observing robot team members as references to develop accurate maps is required (Reklaitis et al., 2000). While the work done in this field has had significant impact on robot control architectures, these results largely do not address the problem of cooperative sensing in the context of mobile robots in unknown, unstructured environments. The methods developed to date generally

rely on assumptions that include: simple well-known terrains; accurate knowledge of the environment; little or no task uncertainty; sufficient sensing capability and sufficient communication capabilities. For real field environments, these assumptions are often not valid. In general, current research has not solved the problem of controlling multiple mobile robots performing cooperative tasks in unknown environments, where uncertainty, limited sensing capabilities, and incomplete physical models of the system(s)/environment dominate the problem.

This paper presents a cooperative multi-agent algorithm for the visual exploration of an unknown environment. The basic approach to the algorithm is given in Figure 2 (Sujan and Dubowsky, 2002; Sujan et al., 2003). This algorithm fuses sensory information from one or multiple agents using physical sensor models, robot models, and environment maps to yield geometrically consistent surrogate information *in lieu* of missing data due to the environment, task, robot and sensor uncertainties. The multiple agents can have different shapes and kinematic constraints, as long as their models are known. The algorithm falls into the general category of SLAM. The mapping and localization process is as follows. First, each agent efficiently repositions its sensors using an information theoretic approach, in order to optimally fill in uncertain/unknown regions of the environment map, based on maximizing the expected new information obtained. Next, each agent fuses the data to its known environment model by localizing itself with respect to a global fixed reference frame. Finally, each agent shares its known environment map with its team, which is then integrated by the other agents into their own environment maps. The information is then used by the control and planning architecture to plan further movements of the sensors for each agent. Thus a common environment map is built by fusing the data available from the individual robots, providing improved accuracy and knowledge of regions not visible by all robots. The algorithm is

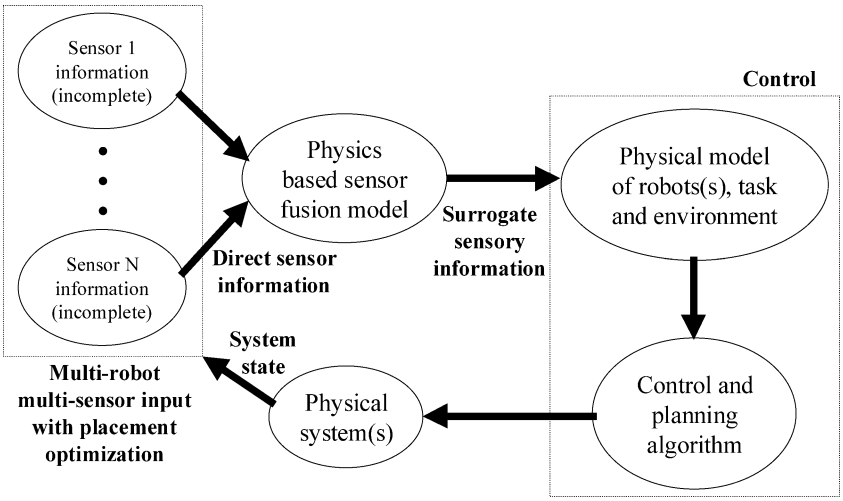


Figure 2. An architecture for multi-robot cooperative sensing.

unique in using the quantity of information of the environment that it currently has, predicting high information-yielding viewpoints from which to continue exploring the environment. This results in a significantly more efficient exploration process. This algorithm is directly applicable to a team of robots as it is to a single explorer. This generality also extends to data representation formats (e.g., 2-D, 2.5-D and 3-D representations). This is achieved through a unique information representation, sharing and fusion architecture.

2. Algorithm Description

The algorithm may be broken down as follows. First, the vision sensors cooperatively scan the region around a target, generating a local 3D geometric model. This allows the robots to locate themselves and the obstacles in the target reference frame. Next, these models are used to find optimum environment viewing poses for the multiple vision systems (Sujan and Dubowsky, 2002). The process is initialized by visually finding the target and robots in a common reference frame. Then, a new pose is found for each of the sensors by defining and optimizing a rating function (RF) over the possible sensor positions, subject to kinematic constraints of the sensor placement mechanisms for the individual robots. This rating function aims to acquire as much new information about the environment as possible with every sensing cycle, while maintaining or improving the map accuracy, and minimizing the exploration time. The process is constrained by selecting goal points that are not occluded and that can be reached by a collision-free traversable path. The sensors then move to their new poses and acquire 3D data. Based on the sensor mount kinematics, the motion of the sensor is known. However, small motions of the robot base (due to suspension compliance) and errors in sensor mounts lead to additional uncertainties. These are accounted for by measuring common features during the vision sensor motion. Finally, the new data and its associated uncertainty are fused with the current environment map, resulting in an updated probabilistic environment map, which may then be shared with the other sensing agents. In general, for each sensing agent the algorithm consists of four steps, described as follows (see Figure 3).

2.1. STEP 1: SYSTEM INITIALIZATION

Here the environment map is initialized, the robots are localized, and a first map is generated. The environment may be mapped using several different representations. In this paper, both 2.5-D elevation maps and 3-D occupancy grid maps are addressed. In 2.5-D elevation maps, the map is a plane of grid cells where each grid cell value represents the average elevation of the environment at that cell location. Uncertainty in elevation is also maintained with each grid cell. In 3-D occupancy grid maps, the map is modeled as a probabilistic discretized occupancy 3-D grid. Every voxel in the 3-D grid has a value for probability-of-occupancy that ranges

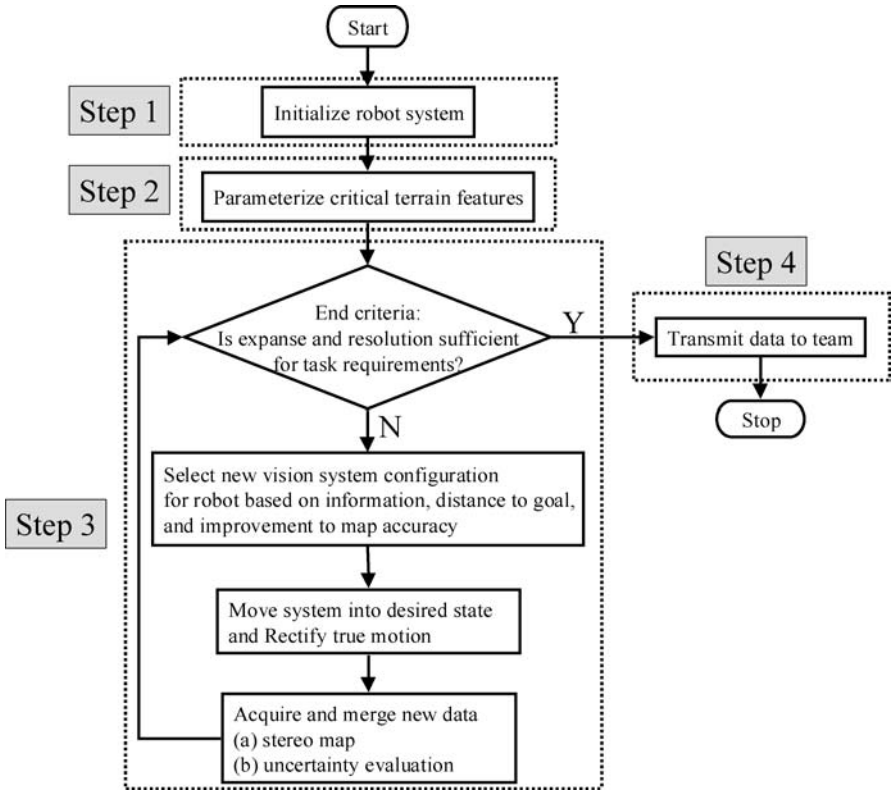


Figure 3. Outline of model building and placement algorithm.

from 0 (empty) to 1 (occupied). A value of 0.5 indicates maximum uncertainty in occupancy of the voxel. Each representation has its advantages, as follows. The 2.5-D elevation maps are particularly useful for terrain profiling where there are little or no terrain overhangs. On the other hand, the 3-D occupancy grid maps are more useful in areas that do have a large number of significant overhangs, such as caves, rooms, etc, since 2.5D maps only provide one data point, the elevation, for each planar grid cell.

The map is built in a fixed reference frame defined by a well-known landmark measurable by all the sensing agents. All robots contributing to or requiring use of the map are localized with respect to the initial map. Localization may be achieved by either:

(a) absolute localization, achieved by mapping a common environment landmark that is visible by all robots; or
(b) relative localization, done by mapping fiducials on all robots by other robot team members where one robot is selected as the origin.

Absolute localization is used in this application, although relative localization has been applied in the past (Sujan et al., 2003). Other methods can be used for either absolute or relative localization (Yamauchi et al., 1998). Searching for the

target (which forms the absolute origin) by moving the robot sensors can be done in many ways, such as exhaustive raster scanning, random walking, tracking "space filling curves", and model-based image understanding methods (Tarabanis et al., 1995). In this study, sensor positioning for target searching is done in the same way as sensor positioning for environment model building (described in Step 3, Section 2.3). The absolute origin target is located by matching the known target element geometric CAD model with visual data (Lara et al., 1998). At this stage, the environment model is considered empty, i.e. no points are known. The first stereo range map (including the common target and all objects within the field of view) is taken by each agent. It is assumed that only the geometry of the task elements – such as parts of a solar panel that is to be assembled (Huntsberger et al., 2001) – are well known. Obstacles and robot positions are unknown.

2.2. STEP 2: CRITICAL TERRAIN FEATURE IDENTIFICATION

In some applications, certain regions of the terrain may be critical, requiring early identification, mapping and monitoring as the environment model grows (Huntsberger et al., 2003; Schenker et al., 2001; Sujan et al., 2003). An example is determining regions of safe travel for the sensing agents. For this process, the incomplete environment model is temporarily completed by a Markovian approximation for unknown environment grid point values. A nearest measured neighbor average is performed and iterated till convergence.

2.3. STEP 3: OPTIMUM INFORMATION GATHERING POSE SELECTION

A rating function is used to determine the next location (position and orientation) of the sensing agent from which to explore the unknown environment. The objective is to acquire as much new information about the environment as possible with every sensing cycle, while maintaining or improving the map accuracy, hence minimizing the exploration time. The process is constrained by selecting goal points that are not occluded and that can be reached by a collision-free feasible path.

In this paper the sensors are CCD stereo cameras. The new information (NI) is equal to the expected information of the unknown/partially known region viewed from the sensor pose under consideration. This is based on the known obstacles from the current environment map, the field of view of the sensor and a framework for quantifying information. Shannon showed that the information gained by observing a specific event among an ensemble of possible events may be described by the following function:

$$H(q_1, q_2, \ldots, q_n) = -\sum_{k=1}^{n} q_k \log_2 q_k, \tag{1}$$

where q_k represents the probability of occurrence for the kth event. This definition of information may also be interpreted as the minimum number of states (bits)

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

$$q_i = \frac{n_i}{N} \quad \text{for } i = 1, \ldots, N_{\text{gray}}, \tag{2}$$

where n_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 information 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 information. Although this is generally true, it is critical to note that images with ordered patterns may result in the same information content as one with no order. For example, a uniform histogram may be mapped to two very different images, such as a random arrangement of intensity values and a (uniform) smooth color gradient. Intuitively, the former would be expected to contain more information but using Equations (1) and (2), they result in the same value. This anomaly exists for both the 2.5-D and 3-D models. However, this is readily rectified using conventional lossless image compression algorithms, defined below. Thus, before the information content of a data set can be evaluated, it must be processed by a compression algorithm.

A *compression program* is used to convert data from an easy-to-use format to one optimized for compactness. Likewise, an *uncompression program* returns the information to its original form. Only compression is addressed here (since a measure on the information present after compression is required, with no needs for decompression of the data). Decompression techniques can be inferred from the compression methods. Refer to (Smith, 1999) for complete descriptions.

There are many different forms of compression, classified in various ways. One way to classify the compression techniques is lossless vs. lossy. A lossless technique means that the restored data file is *identical* to the original. This is absolutely necessary for many types of data, for example: executable code, word-processing files, tabulated numbers, etc. In comparison, data files that represent images and other required signals do not have to be kept in perfect condition for storage or transmission. All real world measurements inherently contain a certain amount of noise. If the changes made to these signals resemble a small amount of additional noise, no harm is done. Compression techniques that allow this type of degradation are called lossy. This distinction is important since lossy techniques are much more effective at compression than lossless methods. The higher the compression ratio, the more noise added to the data. A few common methods of lossless compression are Simple Run-length compression, Lossless JPEG, Huffman coding, and Lempel–Ziv–Welch (LZW) compression. An ideal compression algorithm would remove all traces of any pattern in the data. Such an algorithm currently does not exist, however the LZW compression algorithm is well recognized to approach this limit. A thorough review is beyond the scope of this paper, but it can be found in

(Smith, 1999). Limited studies on several of the above methods have been carried out and results will be presented.

This concept of information content may now be extended to both 2.5-D environment elevation maps as well as 3-D environment occupancy grid maps. The process is constrained by selecting goal points that are not occluded and that can be reached by a collision free traversable path.

2.3.1. *Extension to 2.5-D Environment Elevation Map*

The new information content for a given sensor (camera) view pose is given by

$$H(\mathrm{cam}_{x,y,z,\theta_p,\theta_y}) = \sum_i \frac{n^{\max}_{\mathrm{grid}} - n^i_{\mathrm{grid}}}{n^{\max}_{\mathrm{grid}}} \left\{ \left(\frac{P^i_V}{2} \log_2 \frac{P^i_V}{2} \right) + \left(1 - \frac{P^i_V}{2} \log_2 \left(1 - \frac{P^i_V}{2} \right) \right) \right\}, \quad (3)$$

where H is an information measure summed over all grid cells i visible from camera pose $\mathrm{cam}_{x,y,z,\theta p,\theta y}$; n^i_{grid} is the number of environment points measured and mapped to cell i; $n^{\max}_{\mathrm{grid}}$ is the maximum allowable mappings to cell i; and P^i_V is the probability of visibility of cell i from the camera test pose. Note in Equation (3) that when the number of environment points measured and mapped to cell i reaches the maximum allowable mappings, its contribution to the new information content is zero, as expected. Locations where the elevation is greater than the local average elevation ($+2\sigma$) are considered as unoccupied and form candidate test poses for the vision system, since these locations are least likely to be obstructed.

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 it is a second order linear approximation (Smith and Cheeseman, 1986). This linear approximation corresponds to the use of a Gaussian (having all higher moments of zero). Second, from 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). For most range sensing systems, this model can be approximated as (Sujan and Dubowsky, 2002):

$$\sigma_{x,y,z} \approx S \cdot T_{x,y,z} \cdot L^n, \quad (4)$$

where S is an intrinsic parameter uncertainty constant, $T_{x,y,z}$ is an extrinsic parameter uncertainty constant, L is the distance to the feature/environment point, and n is a constant exponent (typically 2).

The probability of visibility P^i_V of a target cell (Δx, Δy) from the camera location is evaluated by computing the likelihood of occlusion of a ray $ray_{x,y,z}$ using all

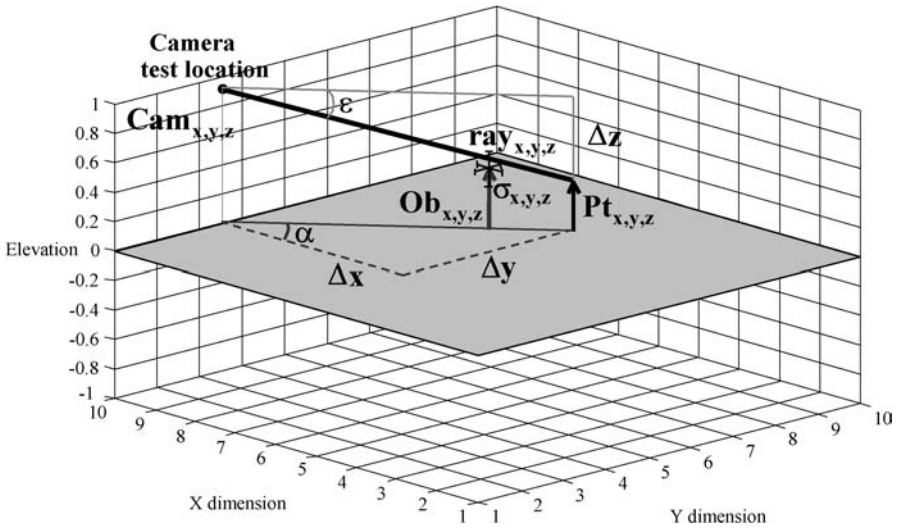


Figure 4. Ray tracing to determine probability of visibility of a grid cell from a given camera configuration.

the elevations of the obstructions, $Ob_{x,y,z}$, and the associated uncertainties, $\sigma_{x,y,z}$, at cells lying along the ray path shot through each target in the environment grid to the camera center. From Figure 4, if grid cell i falls within the camera field of view, then its average elevation, $Pt_{x,y,z}$ (obtained either as an average of all measured points mapped to cell i, or as the Markovian approximation of its neighborhood if no points have currently been mapped to cell i) traces a ray to the camera center, $\text{cam}_{x,y,z}$. Assuming z as the vertical direction, P_V^i is given by the product of the probability of visibility over all cells lying under the light ray from the target cell $(\Delta x, \Delta y)$ to the camera location:

$$P_V^i = \prod_{ray} \left\{ \text{sgn}(ray_z - Ob_z) \int_0^{ray_z - Ob_z} \frac{1}{\sigma_z \sqrt{2\pi}} \exp\left(-\frac{z^2}{2\sigma_z}\right) \text{d}z + 0.5 \right\}. \quad (5)$$

Note that an exact solution for P_V^i should be given by the volumetric integral about $\text{d}x$, $\text{d}y$ and $\text{d}z$. However, for computational simplicity we have only applied an approximation by considering the single dimension, $\text{d}z$.

2.3.2. *Extension to 3-D Environment Occupancy Grid*

From the probabilistic geometric environment model, (x, y, z) locations with probability of occupancy smaller than 0.05 (2σ) are considered as unoccupied. Such points form candidate configuration space camera pose coordinates. In such an instance the scene probability distribution for information analysis is still given by Equation (2). 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 equation is evaluated separately for

mapped (known) versus unmapped (unknown) regions:

$$H(\mathrm{cam}_{x,y,z,\theta_p,\theta_y}) = -\left(\left(\sum_{k=1}^{n} q_k \log_2 q_k\right)_{\text{known}} + \left(\sum_{k=1}^{n} q_k \log_2 q_k\right)_{\text{unknown}}\right). \tag{6}$$

The possible gray level values are defined as follows. For all unknown/unsampled voxels, an occupancy value $p(\bar{x})_{\text{unknown}}$ may be defined in the form of a Markovian chain, i.e., $p(\bar{x})$ of a particular voxel is the average value of $p(\bar{x})$ of the neighboring voxels. Intuitively, this results in unknown regions that are mapped as averages of closest known regions. Thus, for all spatial voxels, a gray (probabilistic) occupancy value between 0 and 1 is found. Next, the values for $p(\bar{x})$ are modified as follows:

$$\begin{aligned} \text{stretching:}\quad p'(\bar{x}) &= \begin{cases} \dfrac{1}{1-p(\bar{x})} \cdot \dfrac{1}{d_{\text{voxel}}} & \forall p(\bar{x}) < 0.5, \\ \dfrac{1}{p(\bar{x})} \cdot \dfrac{1}{d_{\text{voxel}}} & \forall p(\bar{x}) \geqslant 0.5, \end{cases} \\ +\,\text{scaling:}\quad p''(\bar{x}) &= \begin{cases} \dfrac{p'(\bar{x})-1}{2} & \forall p(\bar{x}) < 0.5, \\ 1 - \dfrac{p'(\bar{x})-1}{2} & \forall p(\bar{x}) \geqslant 0.5, \end{cases} \end{aligned} \tag{7}$$

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. A uniform discretization of this range of $p''(\bar{x})$ values may be performed to define the gray level values i, f_i, N_{gray} and N in Equation (2). With these definitions, q_k of Equation (2) is evaluated and the results applied to Equation (6), resulting in a metric for new information (NI). Alternatively, a possibly better choice is a uniform discretization of $p(\bar{x})$ to define the gray level values i, f_i, N_{gray} and N. To increase the contribution of regions with higher occupancy uncertainty to the information metric, the term $q_k \log_2 q_k$ of Equation (6) is premultiplied by $-(p_k \log_2 p_k + (1-p_k)\log_2(1-p_k))$, reflecting the greater expected information available in such regions, resulting in

$$\begin{aligned} H(q) = &-\left(\sum_{k=1}^{n}\left\{-[p_k \log_2 p_k + (1-p_k)\log_2(1-p_k)]\right\} \cdot q_k \log_2 q_k\right)_{\text{known}} \\ &-\left(\sum_{k=1}^{n}\left\{-[p_k \log_2 p_k + (1-p_k)\log_2(1-p_k)]\right\} \cdot q_k \log_2 q_k\right)_{\text{unknown}}. \end{aligned} \tag{8}$$

Therefore, voxels with $p(\bar{x})$ close to zero or one will result in less expected information in the modified expression above, due to their greater level of certainty. The definitions for NI shown in Equations (3), (6) and (8), do behave in an intuitively correct form. For example, for a given sensor pose, if the field of view is occluded then NI decreases. If every point in the field of view is known to be empty, then NI is equal to zero. NI increases as the number of unknowns in the field of view increases. Further, the new expected information also increases in the presence of 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.

2.3.3. *Compensating for Poor Data Quality*

During the mapping process, some regions that are expected to be visible may not be, because of sensor characteristics (e.g., lack of stereo correspondence due to poor textures or lighting conditions), and inaccuracies in the data model (e.g., expected neighboring cell elevations and uncertainties/occlusions), resulting in an unsuccessful measurement for that specific cell. However, after repeated unsuccessful measurements of cells expected to be visible, it becomes more likely that sensor characteristics are the limitation. In this case, it may be impossible to improve the measurements only by increasing their number. Therefore, instead of dedicating too much time trying to unsuccessfully measure those cells, the robot should lose interest in them and try to explore other areas. This is represented as a data quality function that reduces as the number of unsuccessful measurements of the visible grid cell increases. The information metric associated with such regions are pre-multiplied by an "interest function" IF for the grid cell i given at the kth unsuccessful measurement by

$$\begin{aligned} \mathrm{IF}_i^0 &= 1, \\ \mathrm{IF}_i^k &= \frac{1}{\mathrm{e}^{\beta P_V^i}} \cdot \mathrm{IF}_i^{k-1}, \end{aligned} \tag{9}$$

where β is a scaling constant determined empirically, with larger values resulting in faster decrease of IF. Note that occluded regions do not translate to low data quality regions. This permits future "interest" in such regions that may be explored later from another view location.

2.3.4. *Data Fusion*

The next step in environment map building is to fuse the newly acquired data by each agent with the environment model currently available to that agent. At any time, the sensors on each mobile robot can only observe a small part of their environment. However, measurements obtained from multiple view-points can provide reduced uncertainty, improved accuracy, and increased tolerance in estimating

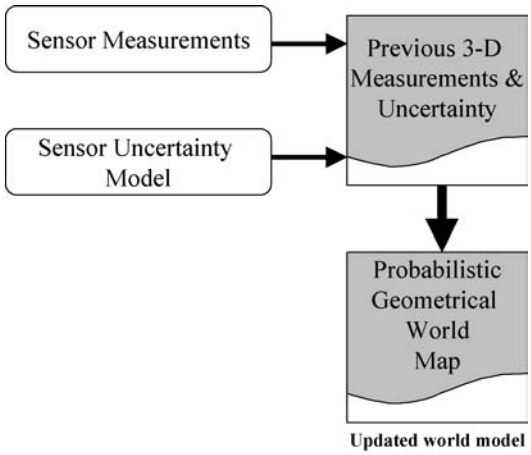


Figure 5. Data fusion with sensor uncertainty.

the location of observed objects (Smith and Cheeseman, 1986). To fuse multiple range measurements of a feature by sensors, a statistical model of sensor uncertainty is employed (see Figure 5). Current and previous range sensor measurements and their uncertainty models can be integrated to give an updated probabilistic geometric model of the environment.

Each agent only fuses its own newly acquired data to the environment map stored in its memory. Thus, as the environment map develops on an individual agent level, it needs to be shared and integrated among the team to keep each agent updated. Optimal map sharing protocols for multi agent systems is currently work in progress, i.e., decentralized protocols instructing the team members when and how to share their individual environment maps. However, once an agent shares its map, the other agents fuse this shared map into their own environment maps using the same method to fuse directly measured data, as described below.

Since the environment model has been developed in a fixed frame (see Step 1), all agents contributing to the environment map require identification of their vision system motion with respect to the fixed coordinate frame, i.e., the agents require global localization. This compensation process during the coordinate transformation reduces robot positioning errors, such as sensor motion errors, and vehicle suspension motions, and allows for accurate data fusion from multiple sources. The process for data fusion is as follows. A single spatial point in the fixed reference frame, $\bar{r}_i$, is related to the image point (u_i, v_i) in the sensor frame by the 4×4 transformation matrix $\mathbf{g}_{01}$, see Figure 6.

For motion calibration of a sensor, $\mathbf{g}_{01}$ needs to be identified:

$$\begin{bmatrix} k_i u_i \\ k_i v_i \\ k_i f \\ 1 \end{bmatrix} = \mathbf{g}_{01} \cdot \bar{r}_i = \begin{bmatrix} [\mathbf{R}_{01}]_{3\times 3} & \overline{\mathbf{X}}_{3\times 1} \\ 0 \quad 0 \quad 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} r_i^x \\ r_i^y \\ r_i^z \\ 1 \end{bmatrix}, \tag{10}$$

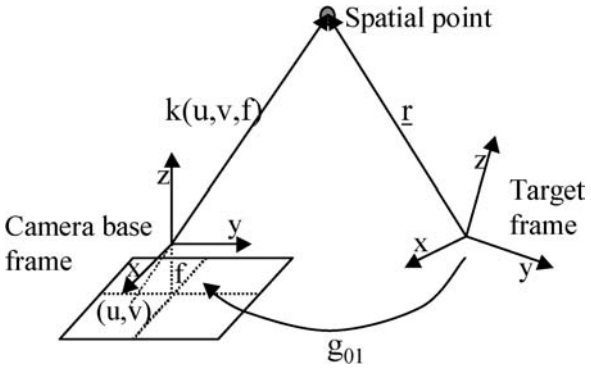


Figure 6. Relationship between sensor and target frames.

where $\mathbf{R}_{01}$ is the rotational matrix, $\overline{\mathbf{X}}$ is the translation vector, f is the sensor (camera system) focal length, and k_i is a scaling constant. For computational reasons, it is more convenient to treat the nine rotational components of $\mathbf{R}_{01}$ as independent, rather than a transcendental relation of three independent parameters. Each spatial point gives three algebraic equations, but also introduces a new variable k_i. This variable is a multiplicative constant to extend the ith image point vector $(u, v, f)_i$ to the ith spatial point in the sensor coordinate frame. These k_i may be found from the disparity pair of the stereo images. For n points it is found that

$$\mathbf{u} = \mathbf{g}_{01} \cdot \mathbf{r} \quad \Rightarrow \quad \begin{bmatrix} k_1u_1 & k_2u_2 & \dots & k_nu_n \\ k_1v_1 & k_2v_2 & \dots & k_nv_n \\ k_1f & k_2f & \dots & k_nf \\ 1 & 1 & \dots & 1 \end{bmatrix}$$

$$= \mathbf{g}_{01} \cdot \begin{bmatrix} r_1^x & r_2^x & \dots & r_n^x \\ r_1^y & r_2^y & \dots & r_n^y \\ r_1^z & r_2^z & \dots & r_n^z \\ 1 & 1 & \dots & 1 \end{bmatrix}. \tag{11}$$

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

$$\mathbf{g}_{01} = \mathbf{u} \cdot \left(\mathbf{r}^{\mathrm{T}}\mathbf{r}\right)^{-1}\mathbf{r}^{\mathrm{T}}. \tag{12}$$

The rotation matrix $\mathbf{R}_{01}$ and the translation vector $\overline{\mathbf{X}}$ of the sensor 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 spatial points are required to more correctly identify the geometric transformation matrix $\mathbf{g}_{01}$. Given the $(i + 1)$st spatial and image point, from the above equations, the new estimates of the rotational matrix and translational vector, $\mathbf{R}_{i+1}$ and $\overline{\mathbf{X}}_{i+1}$, can be obtained. A recursive method is used to determine the mean and covariance $\mathbf{C}$ of

$\overline{\mathbf{X}}$ and $\mathbf{R}_{01}$ based on the previous i measurements as follows:

$$\begin{aligned}
\widehat{\overline{\mathbf{X}}}_{i+1} &= \frac{(i\widehat{\overline{\mathbf{X}}}_i + \overline{\mathbf{X}}_{i+1})}{i+1}, \\
\mathbf{C}_{i+1}^{\overline{\mathbf{X}}} &= \frac{i\mathbf{C}_i^{\overline{\mathbf{X}}} + [\overline{\mathbf{X}}_{i+1} - \widehat{\overline{\mathbf{X}}}_{i+1}][\overline{\mathbf{X}}_{i+1} - \widehat{\overline{\mathbf{X}}}_{i+1}]^{\mathrm{T}}}{i+1}, \\
\widehat{\mathbf{R}}_{i+1}^{(l,m)} &= \frac{(i\widehat{\mathbf{R}}_i^{(l,m)} + \mathbf{R}_{i+1}^{(l,m)})}{i+1}, \\
\mathbf{C}_{i+1}^{R(l,m)} &= \frac{i\mathbf{C}_i^{R(l,m)} + [\mathbf{R}_{i+1}^{(l,m)} - \widehat{\mathbf{R}}_{i+1}^{(l,m)}][\mathbf{R}_{i+1}^{(l,m)} - \widehat{\mathbf{R}}_{i+1}^{(l,m)}]^{\mathrm{T}}}{i+1}.
\end{aligned} \tag{13}$$

This method essentially maintains a measure on how certain the sensor motion is with respect to its original configuration (assuming the original configuration is known very precisely with respect to the common reference frame). This sensor pose uncertainty must be accounted for to obtain an estimate on the position uncertainty of a measured point in the environment. Let the measurement $\bar{\mathbf{z}}$ be related to the state vector (actual point position) $\bar{\mathbf{x}}$ by a nonlinear function, $\mathbf{h}(\bar{\mathbf{x}})$. The measurement vector is corrupted by a sensor noise vector $\bar{\mathbf{v}}$ of known covariance matrix $\mathbf{R}$, giving

$$\bar{\mathbf{z}} = \mathbf{h}(\bar{\mathbf{x}}) + \bar{\mathbf{v}}. \tag{14}$$

Assume that the measurement of the state vector $\bar{\mathbf{x}}$ is done multiple times. In terms of the current measurement, a Jacobian matrix of the measurement relationship evaluated at the current state estimate is defined as

$$\mathbf{H}_k = \left.\frac{\partial \mathbf{h}(\bar{\mathbf{x}})}{\partial \bar{\mathbf{x}}}\right|_{\bar{\mathbf{x}}=\bar{\mathbf{x}}_k}. \tag{15}$$

The state (or position) may then be estimated as follows:

$$\begin{aligned}
\mathbf{K}_k &= \mathbf{P}_k\mathbf{H}_k^{\mathrm{T}}\left[\mathbf{H}_k\mathbf{P}_k\mathbf{H}_k^{\mathrm{T}} + \mathbf{R}_k\right]^{-1}, \\
\hat{\bar{\mathbf{x}}}_{k+1} &= \hat{\bar{\mathbf{x}}}_k + \mathbf{K}_k[\bar{\mathbf{z}}_k - \mathbf{h}(\bar{\mathbf{x}}_k)], \\
\mathbf{P}_{k+1} &= [1 - \mathbf{K}_k\mathbf{H}_k]\mathbf{P}_k,
\end{aligned} \tag{16}$$

where $\mathbf{P}_k$ and $\mathbf{R}_k$ are the covariance of the state error and measurement noise after k measurements.

This estimate is known as the Extended Kalman Filter (Gelb, 1974). Using this updated value for both the measured point $\bar{\mathbf{x}}$ and the absolute uncertainty $\mathbf{P}$, the measured point may then be merged with the current environment model.

A method to obtain appropriate spatial points is now addressed. Spatial points are a visible set of fiducials that are tracked during sensor motion. As the sensor moves, the fiducials move relative to the sensor, eventually moving out of the sensor view. This requires methods to identify and track new fiducials. Fiducials

are selected from the probabilistic environment model based on three criteria: the degree of certainty with which a sampled point is known, the visual contrast of the sampled point with its surroundings, and depth contrast of the sampled point with its surroundings. These are combined into a single fiducial evaluation function (FEF):

$$\text{FEF} = f_1(P(\mathbf{x})) + f_2(C(u, v)) + f_3(H(\mathbf{x})), \tag{17}$$

where:

- $f_1(P(\mathbf{x})) \sim P(\mathbf{x})/r$ is the fiducial certainty, where r is the radius of a sphere centered at the potential fiducial within which neighboring voxels have descending certainty levels (outside this sphere, voxel certainty levels increase, and lower values for r suggest that the region surrounding a potential fiducial is well known, a desirable property);
- $f_2(C(u, v)) \sim$ contrast $(C) \times$ window size (w) is the fiducial visual contrast, with contrast is defined as:

$$C(u, v) = \frac{I(\mathbf{x}) - \bar{I}_w}{\bar{I}_\text{w}}, \tag{18}$$

  where $I(\mathbf{x})$ is the 2D image intensity value of the potential fiducial at $\mathbf{x}$, $\bar{I}_w$ is the average intensity of a window centered at the potential fiducial in the 2D image, and w is the maximum window size after which the contrast starts to decrease; and
- $f_3(H(\mathbf{x})) \sim H(\mathbf{x}) \times$ window size (w) is the fiducial depth contrast, where $H(\mathbf{x})$ is the maximum spatial frequency (from a 3D Fourier transform) at the potential fiducial within a window, and w is the maximum window size after which the power spectrum (of the 3D Fourier transform) starts shifting to higher frequencies (to simplify computation, this may be approximated with some heuristics).

Additionally, a penalty is added if a potential fiducial is too close to other identified fiducials. Using the identified fiducials, sensor motion can be obtained. Fiducials can be tracked with simple methods such as region growing or image disparity correspondence. An example of this process is shown in Figure 7. The flow diagram of visual system motion identification is shown in Figure 8.

Once the sensor motion has been identified using spatial mapped and tracked fiducials, the newly acquired data may be merged with the existing model as follows. 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 deviations $\sigma_{x,y,z}$, a covariance

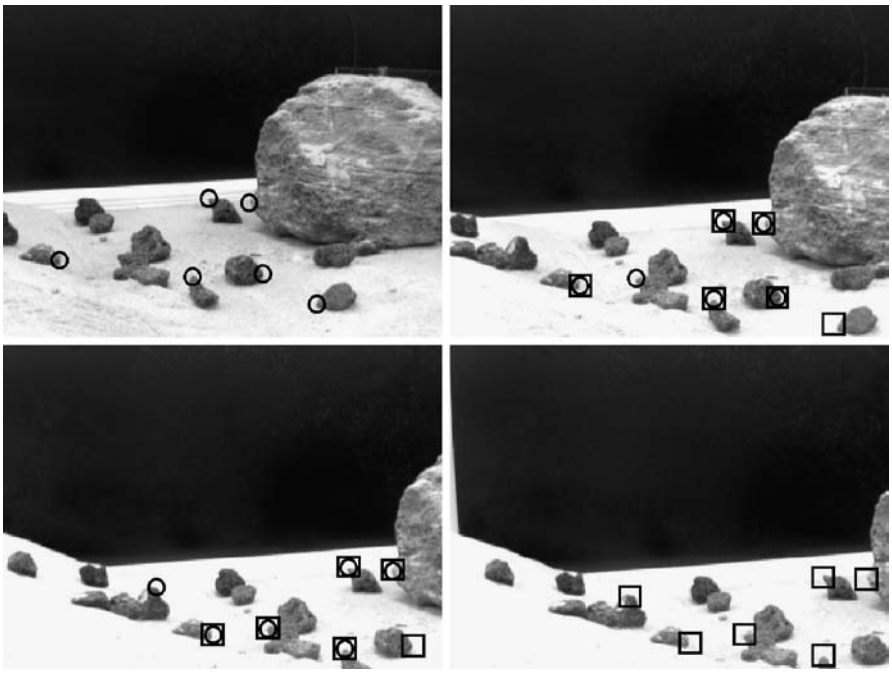


Figure 7. Identification and tracking of 6 fiducials (□ = tracked with previous image, ∘ = tracked with next image).

matrix ($\mathbf{C}$) and the mean ($\bar{x}$) (Stroupe et al., 2000, Smith and Cheeseman, 1986):

$$p(\bar{\mathbf{x}}'|\bar{\mathbf{y}}) = \frac{1}{(2\pi)^{n/2}\sqrt{|\mathbf{C}|}} \exp\left(-\frac{1}{2}(\bar{\mathbf{y}} - \bar{\mathbf{x}}')^{\mathrm{T}}\mathbf{C}^{-1}(\bar{\mathbf{y}} - \bar{\mathbf{x}}')\right), \quad \text{where}$$
$$\mathbf{C} = \begin{bmatrix} \sigma_x^2 & \rho_{xy}\sigma_{xy}\sigma_{xy} & \rho_{zx}\sigma_{zx}\sigma_{zx} \\ \rho_{xy}\sigma_{xy}\sigma_{xy} & \sigma_y^2 & \rho_{yz}\sigma_{yz}\sigma_{yz} \\ \rho_{zx}\sigma_{zx}\sigma_{zx} & \rho_{yz}\sigma_{yz}\sigma_{yz} & \sigma_z^2 \end{bmatrix}, \tag{19}$$

where the exponent is called the Mahalanobis distance. For uncorrelated measured data, it is found that $\rho = 0$. The formulation in Equation (19) is in the spatial coordinate frame. However, all measurements are made in the sensor coordinate frame. This problem is addressed through a transformation of parameters from the observation frame to the spatial reference frame as follows:

$$\mathbf{C}_{\text{transformed}} = \mathbf{R}(-\bar{\theta})^{\mathrm{T}} \cdot \mathbf{C} \cdot \mathbf{R}(-\bar{\theta}), \tag{20}$$

where $\mathbf{R}(\theta)$ is the rotation matrix between the two coordinate frames. The angle of the resulting principal axis can be obtained from the merged covariance matrix (Smith and Cheeseman, 1986; Stroupe et al., 2000):

$$\mathbf{C}_{\text{merged}} = \mathbf{C}_1\big(\mathbf{I} - \mathbf{C}_1(\mathbf{C}_1 + \mathbf{C}_2)^{-1}\big), \tag{21}$$

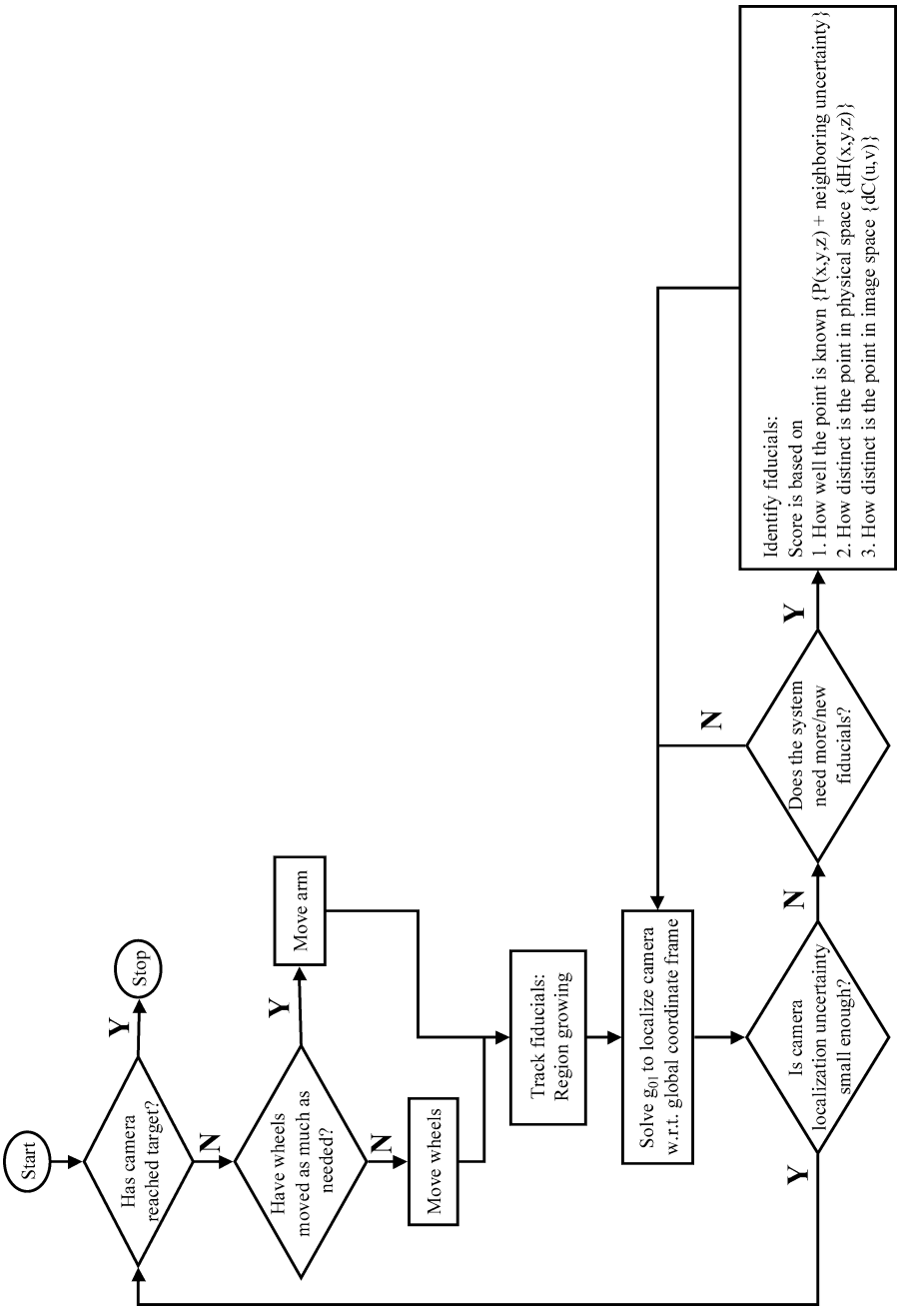


Figure 8. Flowchart for vision system motion identification using scene fiducials.

where $\mathbf{C}_i$ is the covariance matrix associated with the ith measurement. Additionally, a translation operation is applied to the result from Equation (19), to bring the result into the spatial reference frame. To contribute to the probabilistic occupancy environment model, all measured points corresponding to obstacles are merged. That is, all measured points falling in a particular grid cell contribute to the error analysis associated with that voxel.

Note that adding noisy measurements leads to a noisier result. For example, the sensor pose uncertainty increases as the number of sensor steps increase. With every new step, the current uncertainty is merged with the previous uncertainty to get an absolute uncertainty in sensor pose. However, by merging redundant measurements leads to a result with less noise.

2.3.5. *Final Definition of Camera Pose Rating Function*

In addition to maximizing information acquisition, it is also desirable to minimize travel distance and maintain/improve the map accuracy, while being constrained to move along feasible paths. An Euclidean metric in configuration space, with individual weights α_i on each degree of freedom of the sensor pose $\bar{c}$, is used to define the distance moved by the sensor:

$$d = \left(\sum_{i=1}^{n} \alpha_i (c_i - c_i')^2 \right)^{1/2}, \tag{22}$$

where c_i and c_i' are the components of the vectors $\bar{c}$ and $\bar{c}'$ of the new and current sensor poses respectively. Here, α_i is set to unity. In general, this parameter reflects the ease/difficulty in moving the vision system in the respective axis. Map accuracy is based on the accuracy of localization of each sensing agent. This may be obtained by adding the localization error (LE) of the agent along the path to the target. Paths containing more promising fiducials for localization result in higher utility in determining both the goal location and the path to the goal. The new information, the travel distance and the net improvement of map accuracy are combined into a single utility rating function (RF) that may be optimized to select the next view pose:

$$\mathrm{RF} = (w_{\mathrm{NI}} \cdot \mathrm{NI} - w_d d - w_{\mathrm{LE}} \cdot \mathrm{LE})(1 - P_{x,y,z}), \tag{23}$$

where w_{NI}, w_d and w_{LE} are scaling constants. This rating function can be evaluated and optimized to find the next sensor 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. The vision system pose selection algorithm is outlined in Figure 9. Note that the movement of the vision system may require motions by the mobile robot (in addition to manipulator motions). The flowchart in Figure 9 includes a simple path planning approach based on the principle of convex hulls (Sujan and Dubowsky, 2002).

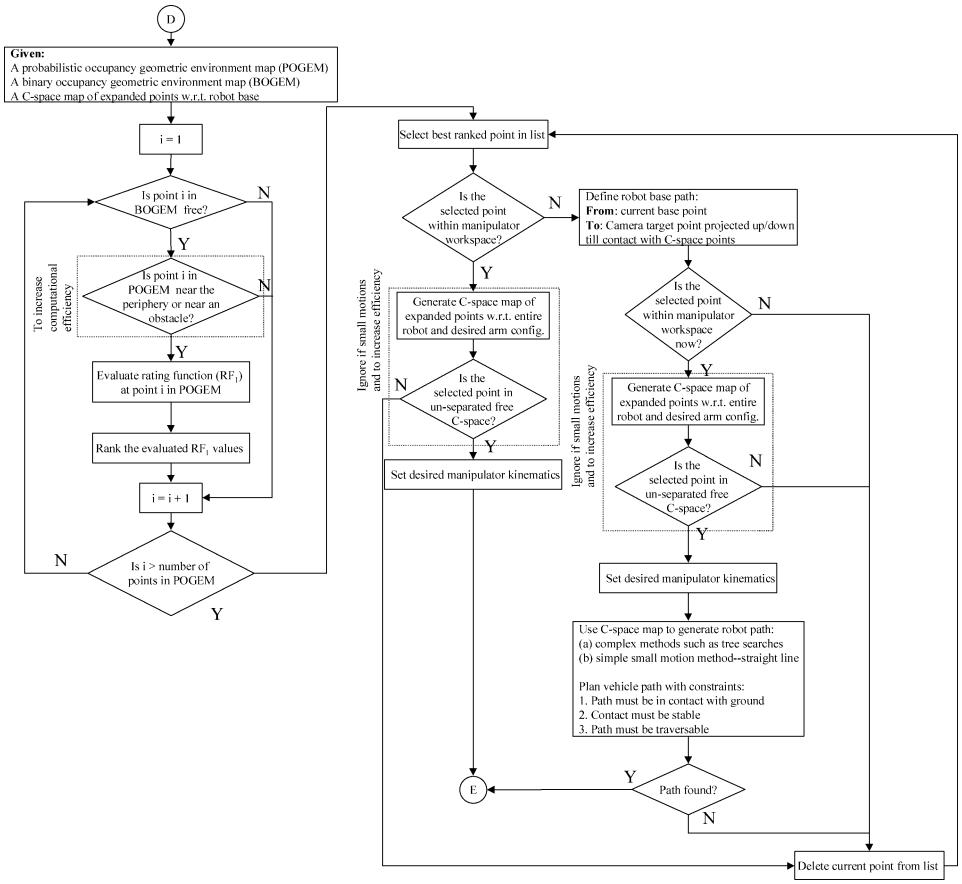


Figure 9. Flowchart for vision system pose selection of environment mapping algorithm.

2.4. STEP 4: MAP DISTRIBUTION

As each agent maps and fuses an environment section to the environment map, it needs to distribute this updated map among the other agents. This is required so that each agent may optimally plan its next move and add information to the map. Once completed, the environment map needs to be distributed to the team (Huntsberger et al., 2001, 2003; Sujan et al., 2003).

Due to communication bandwidth limitations of NASA/JPL present and near-term rovers, an appropriate data transfer algorithm needs to be developed. For example, during the 1997 Mars Sojourner mission, both the lander and rover carried 9600 baud radio modems, with an effective data rate of 2400 bps (NASA, 1997). For the 2003 Mars Exploration Rover (MER) mission the data transfer rates of MER-to-Earth is expected to vary from 3 Kbps to 12 Kbps and MER-to-orbiter is expected to stay constant at 128 Kbps (NASA, 2003). These communication limitations may be further exacerbated with multiple cooperating agents. Thus

successful communication requires the reduction of the data set into relevant data, i.e., only communicate data that is necessary for task execution.

The data reduction algorithm used here breaks down the environment map into a *quadtree of interest regions*. This is achieved by first reducing the entire elevation map with adaptive decimation. This removes highly insignificant objects, such as small pebbles. The resulting data set is divided into four quadrants. The information content of each quadrant is evaluated using Equations (1) and (2). This information content reflects the amount of variation in the terrain quadrant (where higher information content signifies higher variation in the terrain). Quadrants with high information content are further divided into sub-quadrants and the evaluation

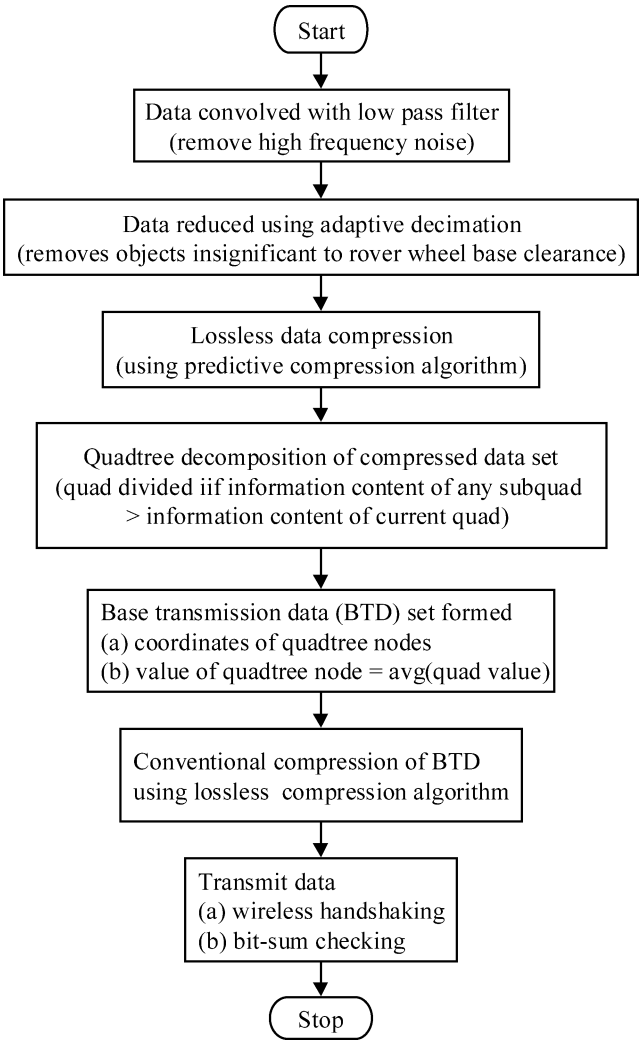


Figure 10. Inter-robot communication flow diagram.

process is continued. Once it is determined that a quadrant does not require further subdivision, an average elevation value of the particular quadrant is used for transmission (rather than the elevation of all grid cells within that quadrant). This cutoff threshold of information is based on a critical robot physical parameter (e.g., the wheel diameter). This results in a significantly reduced data set known as the *quadtree of interest regions*. Conventional lossless compression schemes may then be applied to the reduced data set to further reduce the number of transmission bits. The flow diagram of this process is given in Figure 10.

3. Results

3.1. LOSSLESS COMPRESSION ALGORITHMS COMPARISON RESULTS

Nine techniques for data compression are compared here. These methods are listed in Table I (Sujan and Dubowsky, 2002; Sujan et al., 2003). Five 2-D data set class sizes were used, with each class size comprising of 500 different 2-D data sets (formed from a variety of digital images). These sizes were: 16^2, 32^2, 64^2, 128^2, and 256^2. Further, each class size also comprised of 100 randomly (uniform distribution) generated 2-D data sets, used as control sets. Each of the methods presented in Table I were applied and tested for performance with the above data class sets. Comparison of information content before and after coding give the compression ratio, i.e., number of bits needed to represent a pixel before and after compression using the Shannon entropy function. These results are presented in Table II. As expected, the data sets generated randomly showed little or no compression.

Due to its complexity, the LZW algorithm has not been explicitly coded, but it was tested using available routines. It should be noted that for large data sets, this algorithm is more efficient than the others that were tested (Smith, 1999). However, this is not readily apparent for the tests conducted in this paper. Due to its inherent simplicity and high performance, method 5 was selected as the compression algo-

Table I. Nine lossless compression algorithms for 2-D data

Method 1	No compression
Method 2	LZW compression
Method 3	Predictor: $I(u, v) = I(u, v) - I(u-1, v)$
Method 4	Predictor: $I(u, v) = I(u, v) - I(u-1, v-1)$
Method 5	Predictor: $I(u, v) = I(u, v) - (I(u, v-1) + I(u-1, v) - I(u-1, v-1))$
Method 6	Predictor: $I(u, v) = I(u, v) - (I(u, v-1) + (I(u-1, v) - I(u-1, v-1))/2)$
Method 7	Predictor: $I(u, v) = I(u, v) - \{I(u-1, v) + [I(u, v-1) - I(u-1, v-1)]/2\}$
Method 8	Predictor: $I(u, v) = I(u, v) - [I(u-1, v) + I(u, v-1)]/2$
Method 9	Predictor: $I(u, v) = I(u, v) - [I(u-1, v) + I(u-1, v-1) + I(u, v-1)]/3$
Method 10	Huffman coding

rithm for the remaining work presented in this paper. From Tables I and II, predictor method 5 appears to give the highest compression ratios and is extended to the information theoretic vision planning algorithm for the remaining work presented in this paper.

3.2. ENVIRONMENT MAPPING STUDIES

Due to the qualitative similarity in performance, only the results using the rating function for vision system pose selection to develop a 3-D probabilistic occupancy model of a planar environment are given here. Two simulation results are presented: single sensor/robot modeling of an unstructured environment, and two cooperative sensors/robots modeling of an unstructured environment. Five sensor pose selection methods are compared:

(i) *random pose selection* – the next sensor pose is selected randomly within the known environment;
(ii) *sequential/raster pose selection* – the next sensor pose is selected as the next free location in the known environment from which measurements have not yet been made;
(iii) *pose with maximum expected unmapped (new) region* – the next sensor pose is selected as the location with the largest expected new region while accounting for known occlusions;
(iv) *pose with minimum mapped (old) region (also known as the Frontier strategy)* – the next sensor pose is selected as the location that will map the smallest previously mapped region; and
(v) *pose with maximum expected information*.

Table II. Comparison of compression methods on 2-D images

	Compression ratio (CR $\pm \sigma$)				
	Image: 16^2 pixels	Image: 32^2 pixels	Image: 64^2 pixels	Image: 128^2 pixels	Image: 256^2 pixels
Method 1	1.000 ± 0.000	1.000 ± 0.000	1.000 ± 0.000	1.000 ± 0.000	1.000 ± 0.000
Method 2	1.009 ± 0.038	1.142 ± 0.045	1.253 ± 0.063	1.375 ± 0.089	1.531 ± 0.146
Method 3	1.013 ± 0.033	1.142 ± 0.044	1.247 ± 0.064	1.358 ± 0.091	1.488 ± 0.147
Method 4	0.980 ± 0.025	1.060 ± 0.031	1.149 ± 0.046	1.250 ± 0.066	1.348 ± 0.101
Method 5	0.983 ± 0.041	1.143 ± 0.042	1.277 ± 0.068	1.416 ± 0.111	1.723 ± 0.234
Method 6	0.996 ± 0.041	1.156 ± 0.041	1.289 ± 0.063	1.427 ± 0.098	1.646 ± 0.184
Method 7	0.997 ± 0.039	1.157 ± 0.039	1.288 ± 0.064	1.422 ± 0.101	1.619 ± 0.181
Method 8	1.018 ± 0.035	1.159 ± 0.036	1.282 ± 0.056	1.414 ± 0.086	1.564 ± 0.145
Method 9	1.008 ± 0.031	1.126 ± 0.035	1.236 ± 0.052	1.358 ± 0.077	1.480 ± 0.123
Method 10	0.997 ± 0.001	0.996 ± 0.001	0.996 ± 0.001	0.996 ± 0.001	0.996 ± 0.001

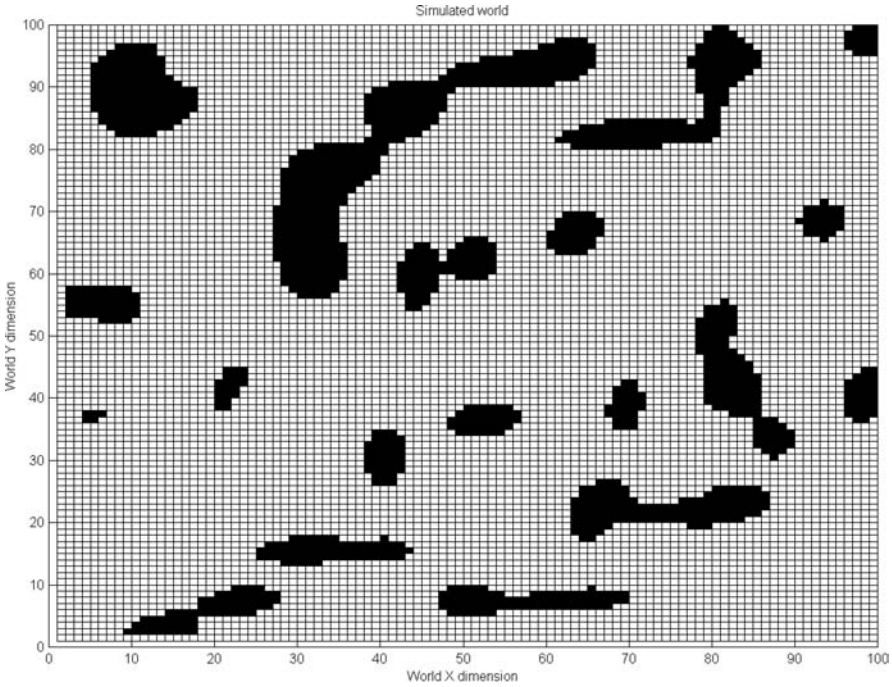
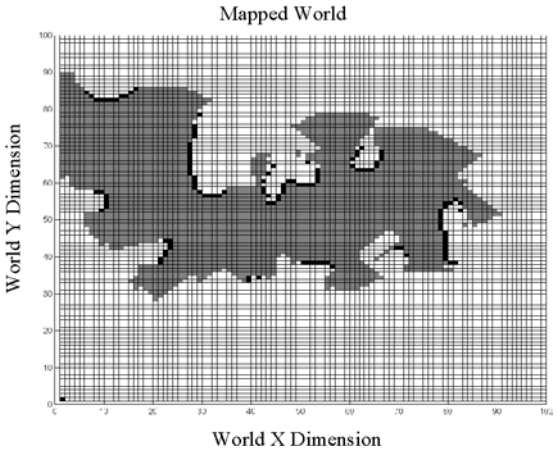


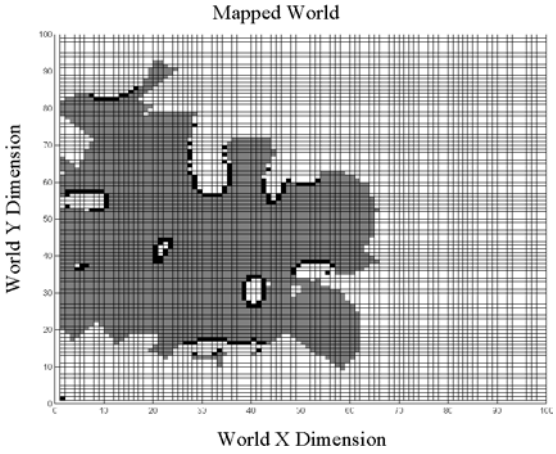
Figure 11. Unknown planar environment.

The first two methods reflect commonly used environment mapping schemes (Burschka et al., 1997; Castellanos et al., 1998; Choset and Nagatani, 2001; Reklaitis et al., 2000; Victorino et al., 2000). The latter three reflect with increased complexity the algorithm developed here. The rating function (RF) cannot be optimized analytically. Furthermore, exhaustive searching though the entire n-point configuration space (a process that takes $O(n)$ time) is computationally very expensive. Here, to reduce the search time, a finite set of goal configurations is employed. This set of goal configurations may be selected in several ways (random, closest to current pose, etc.). For m possible configurations, this process takes $O(m)$ time, where m often 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.

Figure 11 shows an unknown environment (100 m $\times$ 100 m) with occlusions (black) to be mapped/modeled. It is assumed that all mobile mapping sensing agents start at the center of this environment. Figures 12–14 show the results of exploring this environment with a single robot (with a 90° field of view, 15 m depth of field). Figure 12 shows examples of the area mapped using mapping/modeling methods (i) and (v). Figure 13 shows the average accumulated motion error of the vision system as it explores the environment as a function of traveled distance. Comparing Figures 13(a) and (b), it is seen that this error decreases substantially with redundancy of fiducials as well as with proximity of the fiducials.



(a)

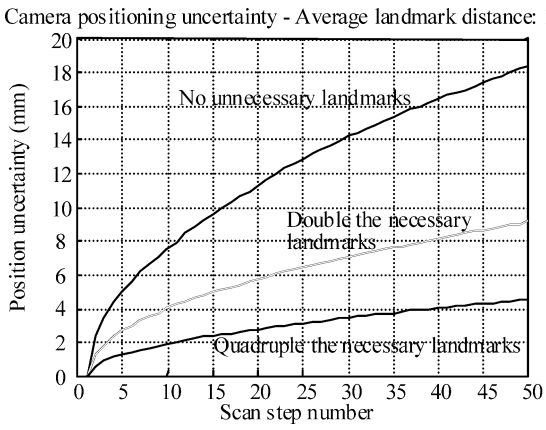


(b)

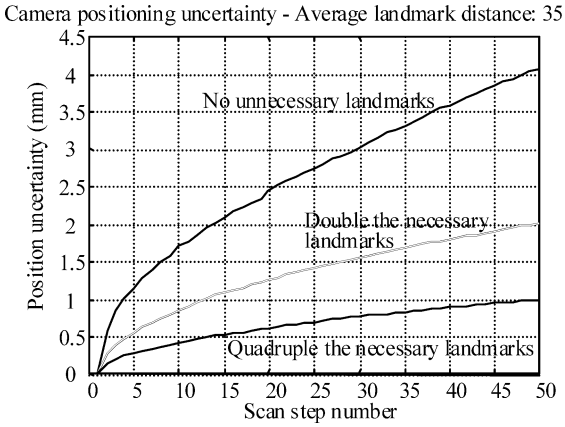
Figure 12. Mapped area by a single vision system (gray = empty space, black = obstacle, white = unknown): (a) random walk pose selection; (b) maximum information pose selection.

Figures 14(a) and (b) show respectively the fraction of the environment mapped and the net distance moved by the vision system for the five mapping methods, using a single mobile vision system. The energy consumption by the system is proportional to the net distance moved by the vision system. Hence it is desirable to have large fraction of the environment mapped with small net displacements.

Note that with random walking you have the likelihood of ending up with a solution that is not as good as the information theory approach, but it is better than conventional methods such as (iii) or (iv). The problem with random walk is the huge amount of energy spent in moving around and the computational load in maintaining position information. The maximum expected unmapped method



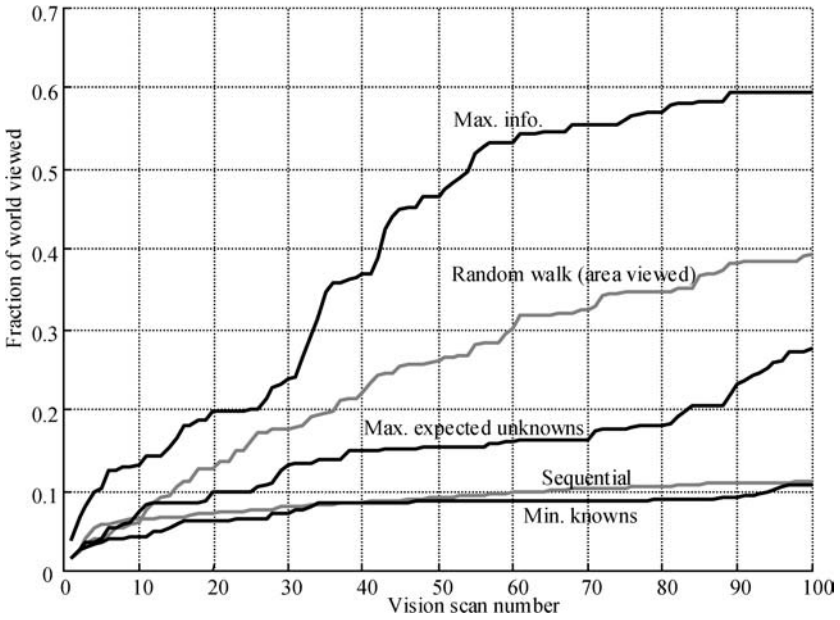
(a)



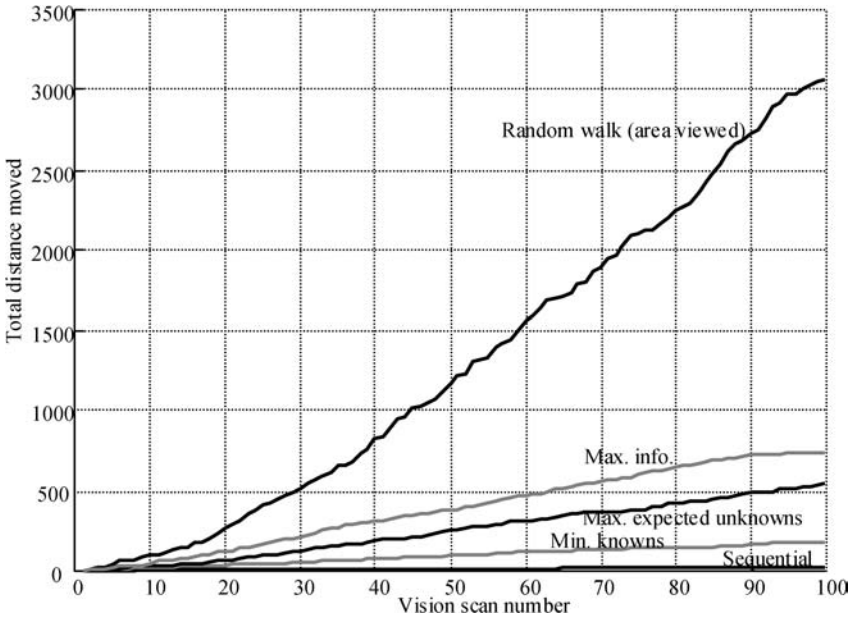
(b)

Figure 13. Accumulated r.m.s. translation error of vision system. Avg. fiducial distance (a) 1100 mm, (b) 350 mm.

and minimum mapped region methods do not account for extrapolations on known obstacles. Hence although they do consider the obstacle that has been mapped, they do not try to figure out where the obstacle may exist. Additionally, when deciding on amount of world mapped, the information theory approach accounts for the certainty to which it is known. Hence, if certain areas have lower certainty (due to distance) these are often remapped with the random walk and the maximum information methods, but not so with methods (iii) or (iv). The max information approach also results in a slightly higher distance traveled compared to, e.g., the max expected unknowns, however the fraction of world viewed is so much higher that this turns out to be an advantage. If comparing the ratio between fraction of world viewed and distance traveled from these figures, the proposed approach turns out to be the best and random walk the worst based on the obtained results.

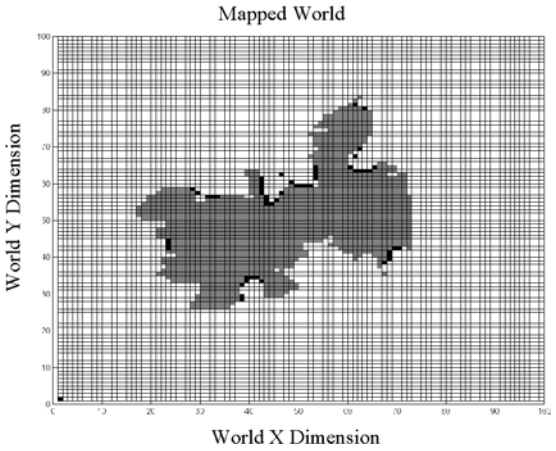


(a)

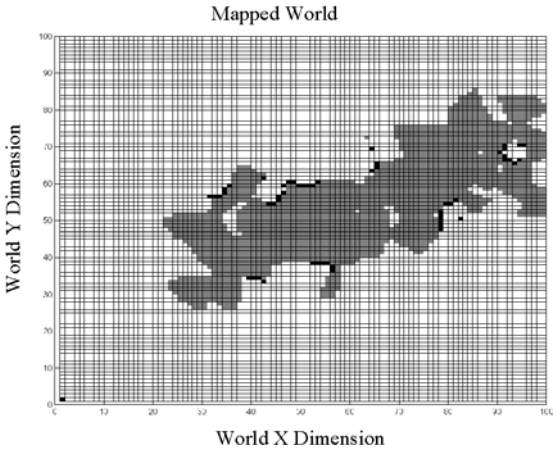


(b)

Figure 14. Results of single vision system modeling an unknown environment: (a) fraction of environment modeled as a function of scan number; (b) distance moved by vision system as a function of scan number.



(a)

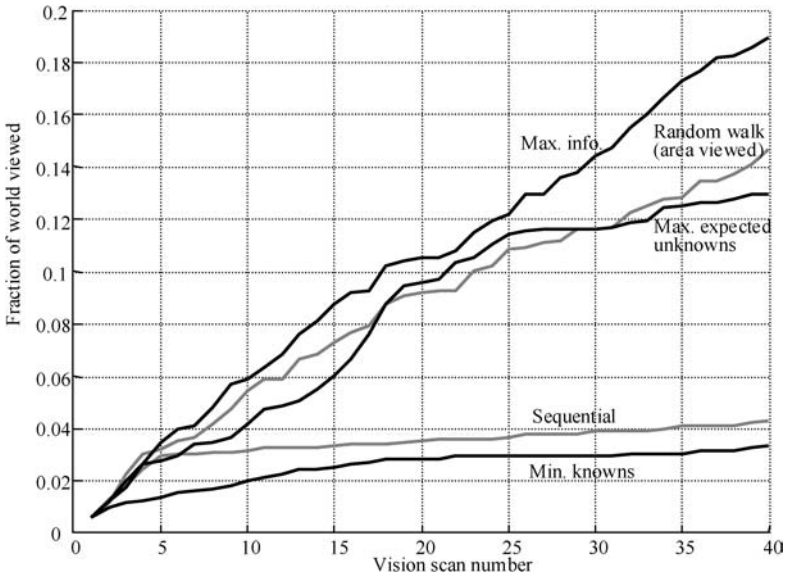


(b)

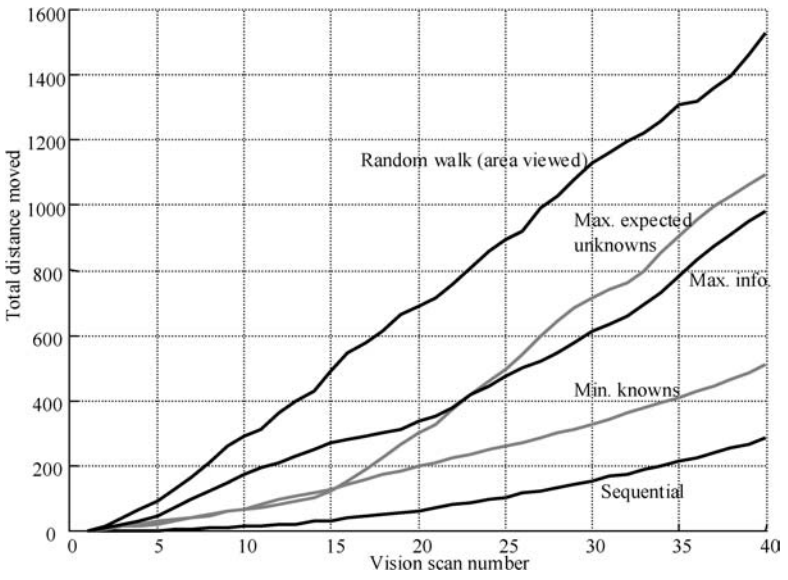
Figure 15. Mapped area by two cooperating vision systems (gray = empty space, black = obstacle, white = unknown): (a) random walk pose selection; (b) maximum information pose selection.

Figures 15, 16 show the results of exploring the same environment cooperatively using two robots (each with 75° field of view, 10 m depth of field). Figure 15 shows examples of the area mapped using mapping/modeling methods (i) and (v). Figure 16 shows the fraction of the environment mapped in Figure 11 and the net distance moved by the vision system for the five mapping methods, using the two cooperating mobile vision systems. These results show the effectiveness of the information theoretic approach to vision system pose selection in environment modeling.

Figure 17 shows an unstructured indoor-type environment (100 m × 100 m) with rooms to be mapped/modeled. Figure 18 shows the fraction of the environment



(a)



(b)

Figure 16. Results for two vision systems modeling an unknown environment: (a) fraction of environment modeled as a function of scan number; (b) distance moved by vision system as a function of scan number.

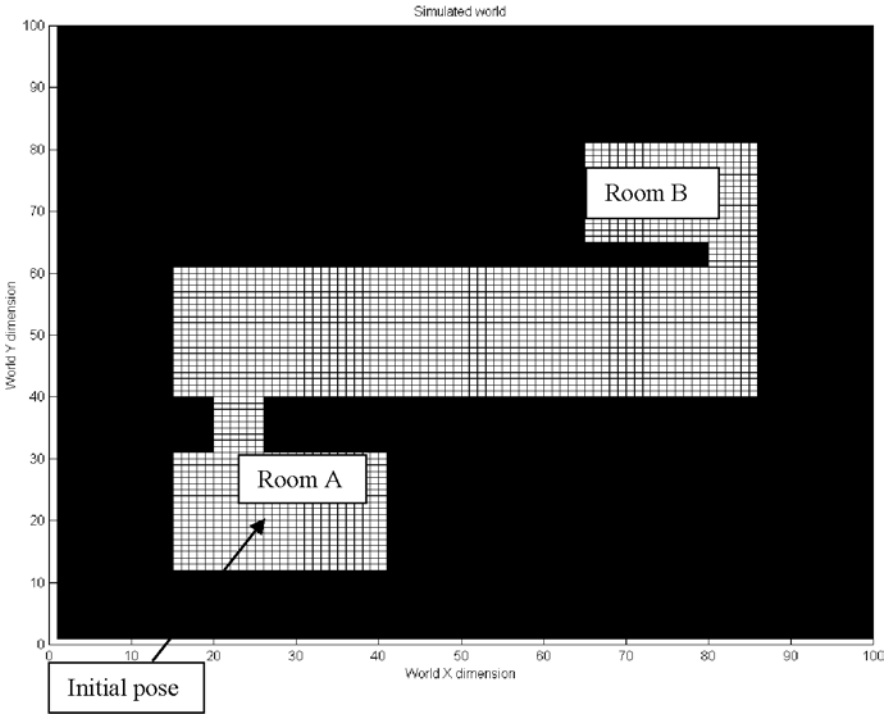
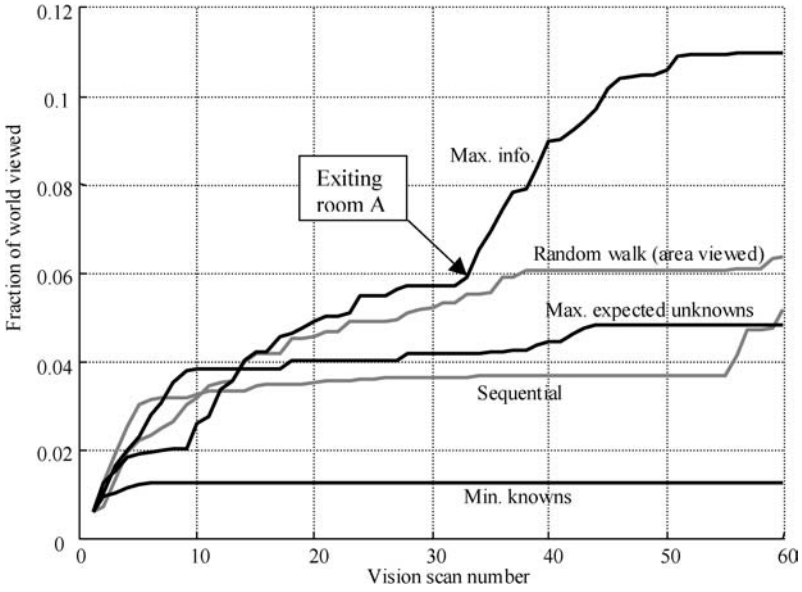


Figure 17. Unstructured indoor-type planar environment.

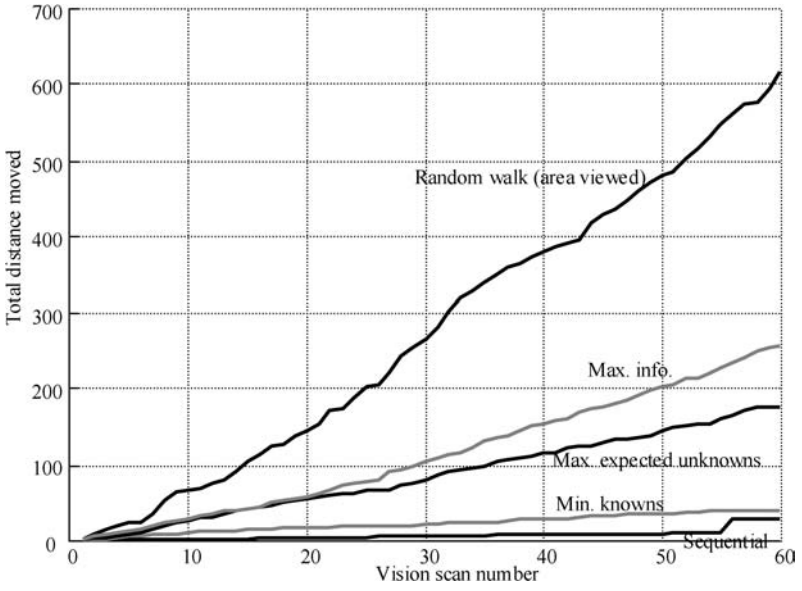
mapped and the net distance moved by the vision system for the five mapping methods, using a single mobile vision system (with 75° field of view, 10 m depth of field). These results again show the effectiveness of the information theoretic approach to vision system pose selection in environment modeling. It is important to note that the information theoretic vision algorithm is the first to map the room and find its way out into the corridor.

3.3. INTER-ROBOT COMMUNICATION STUDIES

The results in the previous sections demonstrate the effectiveness of the multi-agent environment mapping architecture developed in this paper. To demonstrate the effectiveness of the map reduction and distribution algorithm for robots in real planetary exploration field environments, 32 different Mars elevation maps of fixed dimensions, based on the statistics of Viking I/II Mars lander data, were tested. The data of each elevation map was reduced with respect to a robot with varying wheel diameter. To compare the data reduction, a terrain variation parameter *dh* is defined as the terrain elevation variation normalized by the robot wheel diameter. Thus, it is expected that robots with smaller wheel diameters (higher *dh*) require a greater amount of terrain detail for navigation than those with larger wheel diameters for the same terrain map. Figure 19 confirms this expectation. It shows the data



(a)



(b)

Figure 18. Results of single vision system modeling an unknown environment: (a) fraction of environment modeled as a function of scan number; (b) distance moved by vision system as a function of scan number.

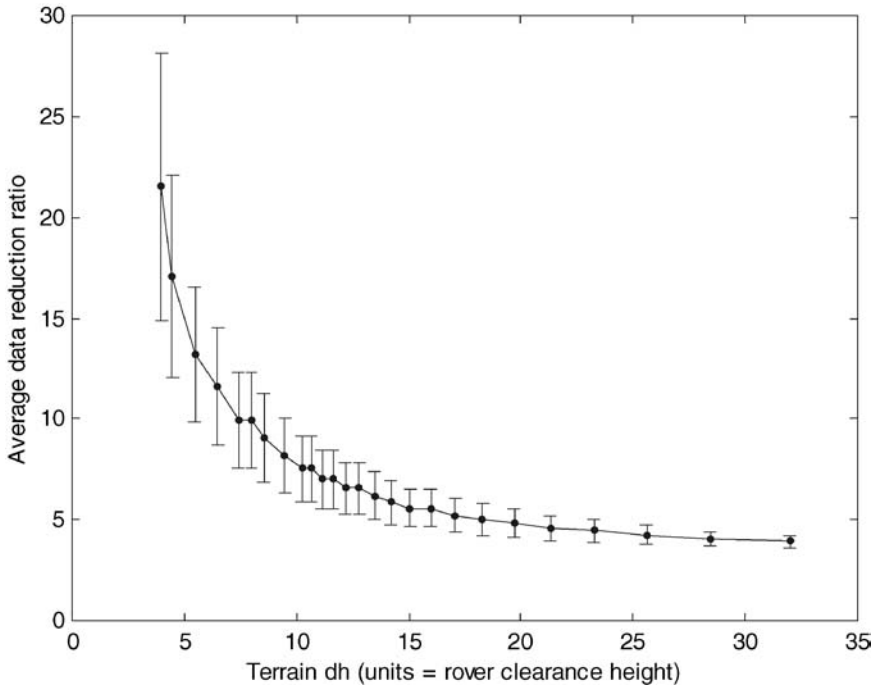


Figure 19. Elevation map data reduction for transmission as a function of terrain elevation range.

reduction factor as a function of *dh* using the algorithm described above (without conventional lossless compression added at the end). The variation at each data point represents the variation in data reduction expected for a given elevation map.

An example of this data reduction process is shown in Figures 20–23. It compares the grayscale elevation map before and after the data reduction process – lighter regions indicate higher elevations. Figure 20 shows a contour map of the simulated environment. Figure 21 shows the quad-tree decomposition of the environment. Note that higher divisions of regions indicate more complex terrain and consequently higher relevant information content. Figure 22 shows the contour map overlaid on the original grayscale elevation map. Finally, Figure 23 compares the elevation map before and after the data reduction process.

For this example, a data reduction factor of approximately 10 was achieved with a $dh = 8$. Although visually the left and right images of Figure 23 may appear the same, closer inspection reveals regions in the transmitted image (such as the bottom right corner) to contain very little information content. This indicates that the region in the original elevation contained insignificant terrain variation with respect to the particular wheel diameter. However, other regions such as the boulders, indicated in the original elevation map, that are critical with respect to the wheel diameter, are faithfully transmitted. It is seen that using this method, significant data reduction can be achieved while maintaining the overall map structure.

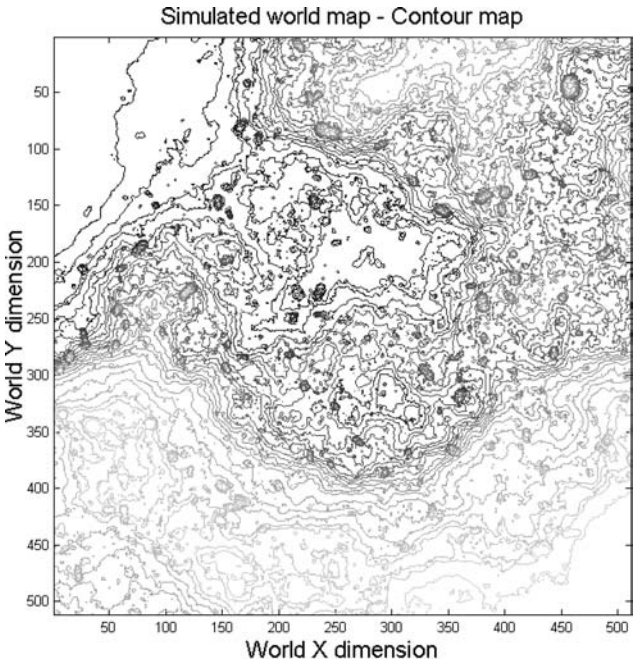


Figure 20. Simulated world contour map – gray color code indicates lighter regions to be of higher elevation.

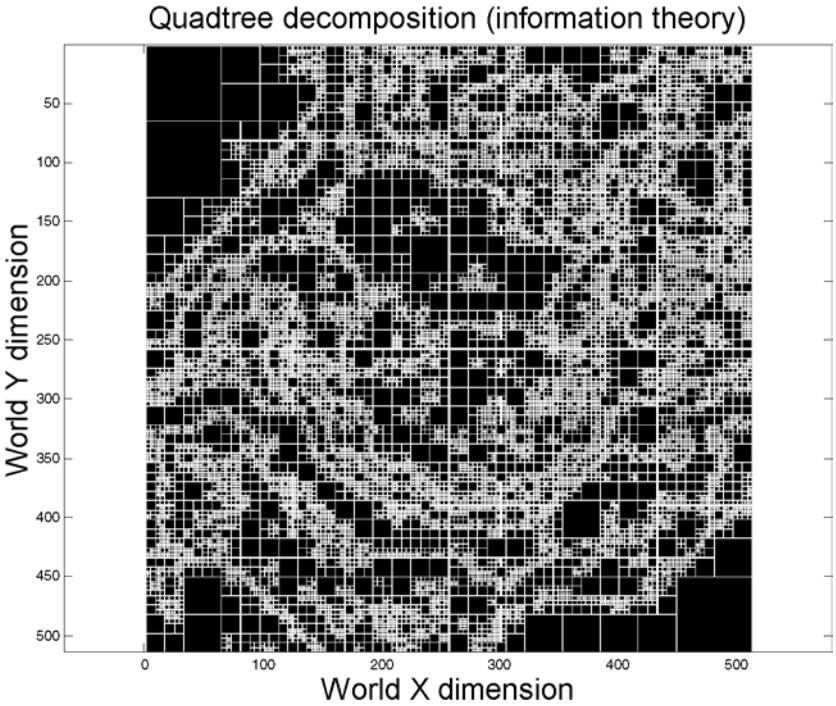


Figure 21. Quadtree decomposition of elevation map.

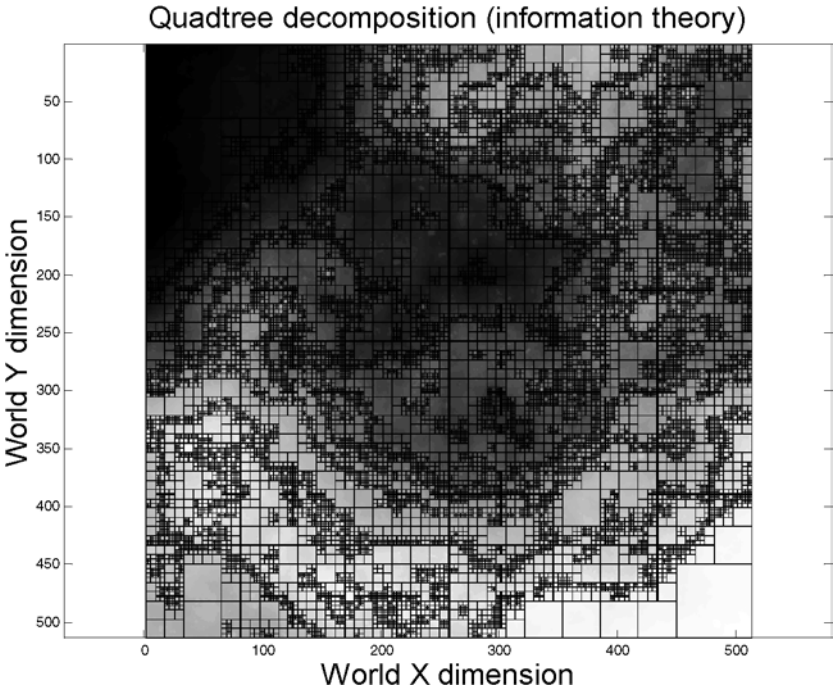


Figure 22. Quadtree decomposition overlaid on original elevation map.

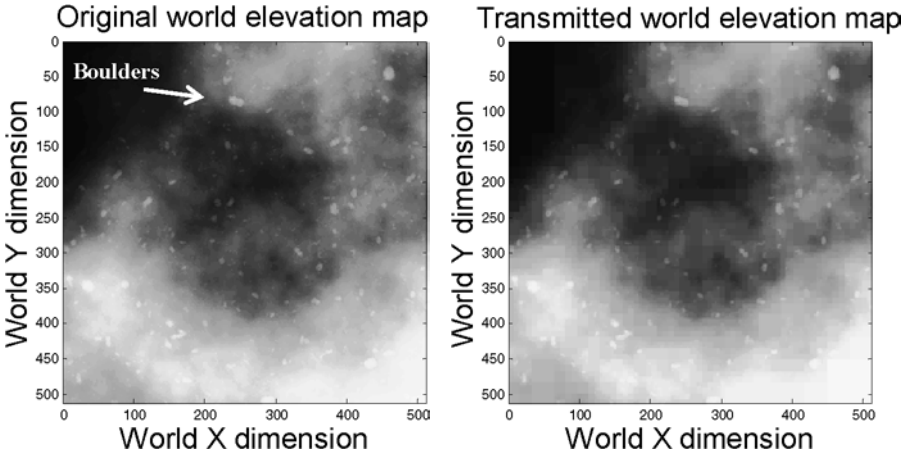


Figure 23. The original (a) and the process/transmitted (b) environment elevation maps.

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 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 with eye-in-hand systems. The aim is to acquire new information that leads to more detailed and complete knowledge of the environment. Controlling robots to maximize knowledge is performed using Shannon's information theory-based evaluation functions. The work applies information theory to enhance the performance of cooperative sensing robot teams compared with traditional pose selection methods. It may be used by multiple distributed and decentralized sensing agents for efficient and accurate environment modeling. The algorithm makes no assumptions of the environment structure. Hence it is robust to robot failure since the environment model being built is not dependent on any single agent frame, but is set in an absolute reference frame. It accounts for sensing uncertainty, robot motion uncertainty, environment model uncertainty and other critical parameters. It allows for regions of higher interest getting more attention by the agents. Simulations show promising results.

References

Anousaki, G. C. and Kyriakopoulos, K. J.: 1999, Simultaneous localization and map building for mobile robot navigation, *IEEE Robotics Automat. Mag.* **6**(3), 42–53.

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

Burschka, D., Eberst, C., and Robl, C.: 1997, Vision based model generation for indoor environments, in: *Proc. of the 1997 IEEE Internat. Conf. on Robotics and Automation*, Vol. 3, Albuquerque, NM, pp. 1940–1945.

Castellanos, J. A., Martinez, J. M., Neira, J., and Tardos, J. D.: 1998, Simultaneous map building and localization for mobile robots: A multisensor fusion approach, in: *Proc. of 1998 IEEE Internat. Conf. on Robotics and Automation*, Vol. 2, Leuven, Belgium, pp. 1244–1249.

Choset, H. and Nagatani, K.: 2001, Topological simultaneous localization and mapping (SLAM): Toward exact localization without explicit localization, *IEEE Trans. Robotics Automat.* **17**(2), pp. 125–137.

Gelb, A.: 1974, *Applied Optimal Estimation*, MIT press, Cambridge, MA.

Hager, G. D., Kriegman, D., Teh, E., and Rasmussen, C.: 1997, Image-based prediction of landmark features for mobile robot navigation, in: *Proc. of the 1997 IEEE Internat. Conf. on Robotics and Automation*, Vol. 2, Albuquerque, NM, pp. 1040–1046.

Huntsberger, T. L., Pirjanian, P., and Schenker, P. S.: 2001, Robotic outposts as precursors to a manned Mars habitat, in: *Proc. of the 2001 Space Technology and Applications Internat. Forum (STAIF-2001)*, Albuquerque, NM, pp. 46–51.

Huntsberger, T. L., Rodriguez, G., and Schenker, P. S.: 2000, Robotics: Challenges for robotic and human Mars exploration, in: *Proc. of ROBOTICS 2000*, Albuquerque, NM, pp. 299–305.

Huntsberger, T. L., Sujan, V. A., Dubowsky, S., and Schenker, P. S.: 2003. Integrated system for sensing and traverse of cliff faces, in: *Proc. of Aerosense'03: Unmanned Ground Vehicle Technology V*, Orlando, FL, SPIE, Vol. 5083.

Jennings, C., Murray, D., and Little, J.: 1999, Cooperative robot localization with vision-based mapping, in: *Proc. of the 1999 IEEE Internat. Conf. on Robotics and Automation*, Detroit, MI.

Kruse, E., Gutsche, R., and Wahl, F. M.: 1996, Efficient, iterative, sensor based 3-D map building using rating functions in configuration space, in: *Proc. of the 1996 IEEE Internat. Conf. on Robotics and Automation*, Vol. 2, Minneapolis, MN, pp. 1067–1072.

Lara, B., Althoefer, K., and Seneviratne, L. D.: 1998, Automated robot-based screw insertion system, in: *Proc. of the IEEE Conf. of the 24th Annual Industrial Electronics Society, IECON '98*, Vol. 4, pp. 2440–2445.

NASA: 1997, `http://mars.jpl.nasa.gov/MPF/rover/faqs_sojourner.html`

NASA: 2003, `http://mars.jpl.nasa.gov/mer/mission/comm_data.html`

Park, J., Jiang, B. and Neumann, U.: 1999, Vision-based pose computation: Robust and accurate augmented reality tracking, in: *Proc. of the 2nd IEEE/ACM Internat. Workshop on Augmented Reality*, San Francisco, CA, pp. 3–12.

Pirjanian, P., Huntsberger, T. L., and Schenker, P. S.: 2001, Development of CAMPOUT and its further applications to planetary rover operations: A multirobot control architecture, in: *Proc. of SPIE Sensor Fusion and Decentralized Control in Robotic Systems IV*, Vol. 4571, Newton, MA, pp. 108–119.

Reklaitis, I., Dudek, G. and Milios, E.: 2000, Multi-robot collaboration for robust exploration, in: *Proc. of the 2000 IEEE Internat. Conf. on Robotics and Automation*, Vol. 4, San Francisco, CA, pp. 3164–3169.

Schenker, P. S., Baumgartner, E. T., Lindemann, R. A., Aghazarian, H., Ganino, A. J., Hickey, G. S., Zhu, D. Q., Matthies, L. H., Hoffman, B. H., and Huntsberger, T. L.: 1998, New planetary rovers for long-range Mars science and sample return, in: *Proc. of SPIE Symposium on Intelligent Robots and Computer Vision XVII: Algorithms, Techniques, and Active Vision*, Vol. 3522, Boston, MA.

Schenker, P. S., Pirjanian, P., Huntsberger, T. L., Trebi-Ollennu, A., Aghazarian, H., Leger, C., Dubowsky, S., McKee, G. T.: 2001, Robotic intelligence for space: Planetary surface exploration, task-driven robotic adaptation, and multirobot cooperation, in: *Proc. of SPIE Symposium on Intelligent Robots and Computer Vision XX: Algorithms, Techniques, and Active Vision*, Vol. 4572, Newton, MA.

Simhon, S. and Dudek, G.: 1998, Selecting targets for local reference frames, in: *Proc. of the 1998 IEEE Internat. Conf. on Robotics and Automation*, Vol. 4, Leuven, Belgium, pp. 2840–2845.

Smith, S. W.: 1999, *The Scientist and Engineer's Guide to Digital Signal Processing*, 2nd edn, California Technical Publishing, San Diego, CA.

Smith, R. C. and Cheeseman, P.: 1986, On the representation and estimation of spatial uncertainty, *Internat. J. Robotics Research* **5**(4), pp. 56–68.

Stroupe, A. W., Martin, M. and Balch, T.: 2000, Merging Gaussian distributions for object localization in multi-robot systems, in: *Proc. of the Seventh Internat. Symposium on Experimental Robotics, ISER'00*, Hawaii.

Sujan, V. A. and Dubowsky, S.: 2002, Visually built task models for robot teams in unstructured environments, in: *Proc. of the 2002 IEEE Internat. Conf. on Robotics and Automation*, Vol. 2, Washington, DC, pp. 1782–1787.

Sujan, V. A., Dubowsky, S., Huntsberger, T., Aghazarian, H., Cheng, Y., and Schenker, P.: 2003, Multi-agent distributed sensing architecture with application to cliff surface mapping, in: *Proc. of the 11th Internat. Symposium of Robotics Research (ISRR)*, Siena, Italy.

Tarabanis, K. A., Allen, P. K., and Tsai, R. Y.: 1995, A survey of sensor planning in computer vision, *IEEE Trans. Robotics Automat.* **11**(1), 86–104.

Thrun, S., Burgard, W., and Fox, D.: 2000, A real-time algorithm for mobile robot mapping with applications to multi-robot and 3D mapping, in: *Proc. of the 2000 IEEE Internat. Conf. on Robotics and Automation*, Vol. 1, San Francisco, CA, pp. 321–328.

Tomatis, N., Nourbakhsh, I., and Siegwar, R.: 2001, Simultaneous localization and map building: a global topological model with local metric maps, in: *Proc. of IEEE/RSJ Internat. Conf. on Intelligent Robots and Systems*, Vol. 1, Maui, HI, pp. 421–426.

Trebi-Ollennu, A., Das, H., Aghazarian, H., Ganino, A., Pirjanian, P., Huntsberger, T. and Schenker, P.: 2002, Mars rover pair cooperatively transporting a long payload, in: *Proc. of 2002 IEEE Internat. Conf. on Robotics and Automation*, Vol. 4, Washington, DC, pp. 3136–3141.

Victorino, A. C., Rives, P., and Borrelly, J.-J.: 2000, Localization and map building using a sensor-based control strategy, in: *Proc. of 2000 IEEE/RSJ Internat. Conf. on Intelligent Robots and Systems*, Vol. 2, Takamatsu, Japan, pp. 937–942.

Yamauchi, B., Schultz, A., and Adams, W.: 1998, Mobile robot exploration and map-building with continuous localization, in: *Proc. of 1998 IEEE Internat. Conf. on Robotics and Automation*, Vol. 4, Leuven, Belgium, pp. 3715–3720.

Yeh, E. and Kriegman, D. J.: 1995, Toward selecting and recognizing natural landmarks, in: *Proc. of the 1995 IEEE/RSJ Internat. Conf. on Intelligent Robots and Systems*, Vol. 1, Pittsburgh, PA, pp. 47–53.

Yu, Y.: 2002, Information theoretical incremental approach to sensor-based motion planning for eye-in-hand systems, PhD Thesis, Simon Fraser University.