Autonomous Robots manuscript No.
(will be inserted by the editor)

3D Mapping with Multi-Resolution Occupied Voxel Lists

Julian Ryde · **Huosheng Hu**

Received: date / Accepted: date

Abstract Most current navigation algorithms in mobile robotics produce 2D maps from data provided by 2D sensors. In large part this is due to the availability of suitable 3D sensors and difficulties of managing the large amount of data supplied by 3D sensors. This paper presents a novel, multi-resolution algorithm that aligns 3D range data stored in occupied voxel lists so as to facilitate the construction of 3D maps. Multi-resolution occupied voxel lists (MROL) are voxel based data structures that efficiently represent 3D scan and map information. The process described in this research can align a sequence of scans to produce maps and localise a range sensor within a prior global map. An office environment (200 square metres) is mapped in 3D at 0.02m resolution, resulting in a 200,000 voxel occupied voxel list. Global localisation within this map, with no prior pose estimate, is completed in 5 seconds on a 2GHz processor. The MROL based sequential scan matching is compared to a standard iterative closest point (ICP) implementation with an error in the initial pose estimate of plus or minus 1m and 90 degrees. MROL correctly scan matches 94% of scans to within 0.1m as opposed to ICP with 30% within 0.1m.

Keywords Mobile robotics · localisation · 3D mapping · 3D laser scanner · SLAM

Julian Ryde
Autonomous Systems Laboratory, CSIRO ICT Centre, PO Box 883, Kenmore, QLD 4069, Australia
Tel.: +123-45-678910 Fax: +123-45-678910
E-mail: julian.ryde@csiro.au

Huosheng Hu
Department of Computing & Electronic Systems, University of Essex, Wivenhoe Park, Colchester CO4 3SQ, U.K.
E-mail: hhu@essex.ac.uk

1 Introduction

Reliable localisation is a clear prerequisite of the mapping process. Once accurate localisation has been achieved, one can progress to building maps. However, mapping raises its own issues, such as what is the best internal representation or map format? And how can new sensor data best be incorporated into the map? If it is accepted that probabilistic depictions are appropriate for both maps and poses then information theory shows how the value of estimators can be extracted from unknown probability distributions without complete knowledge of these distributions. Moreover such sample based approaches converge to the actual values surprisingly quickly. Particle filters are an example of such an approach and describe distributions by samples or particles as in Dellaert et al (1999); Fox (2003); Fox et al (1999); Gustafsson et al (2002); Jensfelt et al (2000); Lenser and Veloso (2000).

Particle filters track a robot's true pose by representing it as a set of possibilities with each assigned a weight. This weighted sample of possible poses tends towards a probability distribution function as the number of particles increases. The particle filter develops the pose distribution over time by updating the poses of the individual particles according to a prediction model. Information from observations is then used to adjust the weights of the new particle set. The computational overhead required by particle filters is orders of magnitude less than that required by grid based implementations as they ensure calculations focus on significant regions of the state space.

In the field of robotics navigation, much emphasis has been placed upon the idea that localisation and mapping have to be performed in parallel. Advocates of simultaneous localisation and mapping (SLAM) or concurrent localisation and mapping (CML) use sophisticated probabilistic analysis to derive maps from uncertain location information.

Efforts to separate the localisation and mapping processes (Wang et al, 2007) go some way to reduce the computational burden of conventional SLAM and this becomes especially important when large numbers of landmarks are considered. In practice navigation usually depends on two types of landmarks, the artificial (Hu and Gu, 2000) and the natural (Wang et al, 2002). The use of natural landmarks in navigation confers a degree of robustness but can lead to errors in situations where landmarks are few. Additionally some knowledge of the local environment is necessary at the outset. Although approaches utilising artificial landmarks are less flexible, their improved ability to locate landmarks simplifies the process of map building.

In the past, navigation algorithms designed for mobile robotics have usually exploited 2D mapping irrespective of whether they use natural or artificial landmarks. This follows from 3D navigation sensors being expensive and requiring greater computational resources. It is far from easy to manage the large amounts of data that a 3D navigation sensor generates and this paper seeks to address this problem. It does so by presenting a novel, multi-resolution algorithm for aligning the 2D range data that has been generated, and then stored in occupied voxel lists, to produce 3D maps. The scan and map data stored in occupied voxel lists are combined by the sampling algorithm to produce the 3D map. The map enables global localisation, without the need for a prior estimate, over a 200m^2 area to an accuracy of $\pm$0.02m and to within one degree. The algorithm accomplishes this task in 5 seconds on a 2GHz processor. Although the process described can both localise and generate maps without recourse to odometry data, such data can be incorporated into the process, thereby shortening the computation time required by reducing the initial pose search volume.

The rest of the paper is organized as follows. Section 2 inspects the research that others have undertaken to represent environments digitally for mobile robots. Section 3 contains the methods for storing and merging 3D data. It includes the representation of occupied voxel lists and contains a detailed description of the multi-resolution, pseudo particle filter solution for aligning multiple occupied voxel lists in real-time. A theoretical analysis of the significance of the overlap value returned by the scan matching algorithm is also presented. Section 4 includes results of single robot mapping experiments and scan matching at various resolutions to test the theoretical work. Finally a brief conclusion, with suggestions for future work, is given in Section 5.

2 Related Work

2.1 Environment Representations

The question of how best to represent the surroundings of a robot in a manner that is easily stored, manipulated and processed has received much attention and two main paradigms have emerged that use either geometric or topological representations. In this work the focus is on geometric mapping and its associated paradigm.

Upon first sight of the laser range data returned by the robot, one is drawn to consider geometric models to represent the environment as in Hager and Burschka (2000). In this paper the authors developed a computationally efficient algorithm for determining the rotation and translation between consecutive laser scans. Other useful approaches include González-Baños and Latombe (2002); Wetzler et al (1994). Map modelling with geometric primitives tends to be hard to implement and can be rather fragile especially in environments containing features inadequately described by the set of primitives. Upon consideration it became apparent that probabilistic approaches not only yield the best experimental results but seem readily scalable to problems involving data from different sensors and robots.

Extensions to straight line representations utilise other geometric primitives such as circles and conic sections (Lee, 1995; Vandorpe et al, 1996) to help describe a wider range of environments. The resulting maps require little storage, describe the environment well and are quick to process for the purpose of path planning. However they are not as popular as occupancy grids or sampling based approaches as they are difficult to update, express probabilities poorly and are more complicated to implement.

Grid based approaches to 2D mapping involving occupancy maps were first envisaged by Elfes (1987); Moravec (1989) and are powerful as well as capable of handling a large degree of uncertainty in position. This facility however requires the storage and modification of an excessively large state space containing all the possible positions and orientations of the robot throughout the environment. It seems that this full representation of the location probability distribution is not always necessary Fox (1998). The ease of implementing the grid based approach has encouraged many researchers (Burgard et al, 1996, 1998; Konolige and Gutmann, 2000; Olson, 2000) to adopt them. A modified $2\frac{1}{2}$ dimensional representation for exterior environments is introduced in Triebel et al (2006) and subsequently used by Kümmerle et al (2008); Pfaff et al (2007). These multi-level surface maps are a good compromise between computational requirements and environment expression.

Quantitative measures of map quality and localisation performance are notably absent from much of the literature as observed by Lee (1995). Some efforts (Elfes, 1991; Lim and Cho, 1992; Rocha, 2005; Rocha et al, 2005; Stachniss and Burgard, 2003a,b) assess the map quality using an average entropy measure.

Mobile robot navigation has evolved in a number of directions in the past couple of decades. Initial work with single robot SLAM has advanced to the investigation of multi-

ple robots operating in a cooperative manner, (Burgard et al, 2005; Rocha et al, 2005, 2008). Improvements in the sensors deployed have encouraged some researchers to embark upon 3D sensing using cameras or 3D laser range finders, (Thrun and Montemerlo, 2006).

2.2 Geometric Mapping

Geometric mapping algorithms attempt to survey the environment by producing detailed measurements, often from range sensors. If, like most indoor environments, the surroundings include many plane surfaces then the straight lines appearing in 2D range scans point us towards polygonal maps, González-Baños and Latombe (2002). Polygonal maps are good for path planning algorithms due to their inherent simplicity and because they have been extensively analysed as a result of their importance in the field of computer graphics. Unfortunately not all environments contain straight lines; cluttered environments such as disaster scenes may not have enough straight lines for effective mapping and navigation. These circumstances are more suited to grid based approaches which can handle almost all environments as they make no assumptions about the prevailing geometry.

Techniques involving probability grids have enjoyed much success over the years and have capitalised on the exponential growth in computer processing power. Some of the drawbacks of these grid techniques, also referred to as probabilistic grid maps, are detailed by Lee (1995). The most serious of these, concerns the probability meaning assigned to each cell and, especially, the assumption that the occupancy probability of a given cell is independent of that of neighbouring cells. A problem with cell probability updates arises in the situation where the robot is stationary but continues to update the map. In these circumstances whatever the detected state of an individual cell it continues to accumulate probability weight until either the robot moves or the map updates cease. This means the accuracy of such a map is particularly vulnerable to systematic errors, such as specular reflection, inherent with sonar sensors. Although this problem is diminished by the use of laser scanners, the assumption of cell occupancy independence still leads to less than optimal results. These problems are addressed by the work on coverage maps of Stachniss and Burgard (2003a,b) and Rocha et al (2005) in which partially occupied cells are correctly represented.

One apparently valid concern is that the accuracy in the pose estimate is fundamentally limited by the resolution of the grid. However this is dispelled by Olson (1999) who achieves a pose accuracy higher than the resolution of the occupancy grid by the insertion of a fitted parameterised surface in the region of the maximal pose likelihood. Researches have attempted to reduce the computational resource requirements of occupancy grids, even in 2D, by various multi-resolution approaches, Moravec (1989); Olson (1999); Yguel et al (2005).

Environmental representations using topology collapse the states representing robot position into a set of discrete, easily identifiable, locations or paths within the arena of operations. The obvious advantage of this is the vast reduction in computational resources required to store and manipulate the position probability distribution. The process of path planning is also greatly simplified by this method as the resulting networked nodes are fast to search. It should be noted that this solution requires easily identifiable features in the environment and, in addition, the algorithms developed for this method are situation specific. Implementation examples include Choset and Nagatani (2001); Kuipers and Beeson (2002) and a Voronoï based approach Victorino et al (2003).

2.3 3D Sensors

Recently considerable effort has been devoted to examining the advantages provided by 3D sensors. To this end some researchers have mounted 2D laser scanners on nodding or rotating mechanisms in order to obtain 3D scans, Cole and Newman (2006); Lingemann et al (2004); Nüchter et al (2005); Pfaff et al (2007). Another popular approach has been to mount two laser scanners so their scan planes are orthogonal, Howard et al (2004). In the latter case though it must be noted that whilst mounting the laser scanners in this way enables 3D maps to be produced they do not give the robot 3D sensory perception, something that is highly desirable for reliable obstacle avoidance. For these reason significant attention is being focused on laser mapping with 3D laser range sensors, Hähnel et al (2002); Howard et al (2004); Montemerlo and Thrun (2004).

A completely different approach is taken in Weingarten et al (2004). Here scans are obtained with modulated infra-red light and an image sensor. Four samples of the intensity of the reflected infra-red light are used to determine the phase difference of the reflected data for each image pixel. From this phase difference and the frequency of modulation of the infra-red source, the distance to image pixels can be ascertained unambiguously up to 7.5m. This sensor delivers 19480 range measurements at 30 frames per second (FPS), with no moving parts, and compares favourably with the accuracy of the SICK laser measurement system (LMS) 200. The main drawback of this approach is the price and low availability of the equipment. The relatively narrow field of view is also a disadvantage though this can be improved by altering the lens arrangement.

There are a small number of commercially produced 3D laser range finders but they are very expensive and the cost can become prohibitive if multiple robots are to be equipped. Companies such as Riegl and Deltasphere sell 3D scanners

that could be modified for robotic use but as they are designed for surveying, and typically tripod mounted, they have a narrow field of view and sacrifice scan rate for high point density.

This multitude of custom 3D range scanners illustrates the rapidly increasing interest in full 3D robotic sensory perception. The benefits of full 3D mapping are abundant and so the rapid expansion of this field is inevitable. The detection of negative and over-hanging obstacles greatly enhances avoidance behaviour. Once 3D maps of environments have been built they can be customised for and distributed to different robots. For instance various 2D occupancy grids may be built for robots of different sizes or with 2D sensors at different heights. Severely cluttered non-manifold environments such as search and rescue situations may be reliably mapped. Maps based upon the ceilings of rooms (Jeong and Lee, 2005) will remain accurate for longer and an unoccluded view of the ceiling is usually readily accessible to a robot even in crowded environments, Burgard et al (1999). An alternative approach taking advantage of full 3D perception (Wulf et al, 2004) uses virtual 2D scans produced by collapsing the 3D data vertically into a plane and then taking the furthest point for each 2D scan angle. This produces good 2D representations in cluttered environments.

Finally, Strand et al (2007) describes a sensor that delivers full colour and associated range data, the scans of which are merged by methods involving both octree spatial decomposition and conventional ICP. The disadvantages of 3D sensing technologies are slower scan acquisition time and the geometric increase in data that need be processed.

3 3D Mapping Method

It is very important to have an indication of the completeness and accuracy of the 3D representation that has been produced. Most authors produce side by side comparisons of photographs of the environment and the corresponding maps. Whilst such comparisons give a rough, qualitative indication of accuracy, a more rigorous, quantitative and reliable indicator is desirable. In this series of experiments the results are tested in two ways. The first is by comparing the product of the algorithm with maps built by manually aligning the 3D scans. The second is a relative measure of the map quality produced by different algorithms, each using the same data set, which can be calculated by (7) or alternatively by the number of voxels in the resulting global map. With a given set of scans, the algorithm that produces a map with the lowest entropy or the lowest global map voxel count is the best.

3.1 Occupied Voxel Lists

Ideally the map is a 3D probability density function and, as a scalar field, it is a function of position $p(x,y,z)$ describing the probability density of occupancy at a particular position in space. Practically however, the map is stored as a discrete representation at sequential times with the most recent map occupancy probabilities being

$$P_{ijk} = \int_z^{z+\varepsilon} \int_y^{y+\varepsilon} \int_x^{x+\varepsilon} p(x,y,z)\, dx\, dy\, dz \tag{1}$$

where the integers i, j, k are

$$i = \left\lfloor \frac{x}{\varepsilon} \right\rfloor, j = \left\lfloor \frac{y}{\varepsilon} \right\rfloor, k = \left\lfloor \frac{z}{\varepsilon} \right\rfloor. \tag{2}$$

The storage and processing requirements for such representations are demanding even in the case of moderately sized environments and, consequently, conventional 2D occupancy grids cannot be readily extended into 3D. As an illustration of the difficulty consider a typical room measuring 10m by 10m by 4m mapped at a resolution of 0.05m. In 2D this would require 40,000 cells to hold the mapping information; for a 3D representation however an 80 fold increase in the number of cells would be required. Such a large increase has great import for both memory requirements and performance and so compromises must be made if the map is to be compressed without an unacceptable reduction in accuracy.

Observations of the nature of the data typically gained from 3D mapping indicate that important reductions in the size of the representation can be achieved without significant loss of information (i.e. lossy compression). The number of cells required depends on the disorder of the room and increases with the number of observable surfaces. Clearly however the number of cells needed is most sensitive to cell size.

In navigable 3D environments there is a huge preponderance of unoccupied cells and so the relative scarcity of occupied cells means their observation is of greater significance. It is therefore important that such observations should be stored and manipulated as a matter of preference. The importance of occupied cells in relation to unoccupied ones is particularly acute when matching sensor scans with prior maps. Their combination of abundance and low significance means that unoccupied cells can usually be safely ignored. Historically unoccupied cells have been retained in occupancy grids because the grids were generally 2D, and therefore relatively small, and the sensors employed to make the observations (usually sonar) were unreliable. A high level of sensor unreliability meant it was unwise to discard cells reported as being unoccupied. Laser scanners though are not only very reliable but produce copious amounts of data thereby making the following lossy compression approach viable.

Rather than holding the cell probabilities in a 3D array, all the cells that have been observed to be occupied more frequently than not are maintained in a list. Each entry in the list contains the position of a voxel and an index of the number of times that voxel has been observed to be occupied. If the occupancy of a particular voxel falls below zero it means the voxel has been observed to be empty more often than occupied and it is removed from the list. Such a list is termed an occupied voxel list (Ryde and Hu, 2007) by analogy with conventional occupancy grids.

This additive map update is fast and is consistent with (10) if the map is regarded as storing a quantity proportional to the log odds occupancy. Compared with a 3D array, not only is there a considerable saving in storage and manipulation overhead but there is no need for strict bounds on the environment.

To get an idea of the benefits delivered by this lossy compression consider a single 3D laser scan with radius 10m and stored at a resolution of 0.02m, which is the smallest sensible resolution as the laser scanner itself has an error of $\pm$0.02m. Assuming a typical room height of 3m the number of voxels required for an occupancy grid would be approximately 100 million. A typical scan stored to a 0.02m resolution in an occupied voxel list results in about 5000 voxels. Evidently storing only occupied cells results in a compression factor of around 20,000. For all scans the additional information stored in the occupancy grid may be generated from that stored in the occupied voxel lists by ray tracing. This is necessary for path planning algorithms that need to know the free space volume just above the floor. The information can also be extracted from the map, as long as the scan poses are available. The computer iterates through nearby poses and, according to the sensor model, traces a ray from the pose to the nearest occupied voxel in each direction. The intervening, free, voxels are recorded in a conventional 2D occupancy grid which can then be used for path planning.

In this instance the representation chosen is a hash table with the voxel's integer position as the index key. The value associated with each position's index key is an integer count of the number of times the cell has been observed to be occupied which, as shown in Section 3.2, is indicative of the probability of occupancy under a number of assumptions. The hash structure means that the average time taken to find the probability at a given position is independent of list size and this procedure is the most common operation on the map. Each point is quantized using (2) and mapped back to real space by

$$(x,y,z) = (a\varepsilon, b\varepsilon, c\varepsilon). \tag{3}$$

Although there is a loss of precision in this process, repeated, alternate applications of (2) and its approximate inverse (3), leave the voxel coordinates unchanged. The integers a, b and c are hashed to produce the look up allowing the occupied voxel list to be stored as a compact list that still benefits from fast query times.

In this manner, the environment is represented as a dense feature set rather than an occupancy grid. Each feature is a voxel that is likely to produce a laser return and this representation scales much better as the size of the environment increases. Data points are recorded for voxels containing surfaces in the environment and, as these surfaces are 2D, the number of voxels required to represent them scales as the square of the linear size of the environment. Occupied voxel lists are superior to conventional feature sets in their generality because no feature extraction is required. This confers flexibility on occupied voxel lists enabling them to represent any environment without losing information as a consequence of featureless data.

Map quality may be determined by the information content as measured by the entropy (4), for example in Rocha et al (2005).

With the entropy of the map,

$$H(P) = -\sum_{x\in P} P_{ijk} \ln P_{ijk} \tag{4}$$

where P_{ijk} is the probability of occupancy for the voxel with indices i,j,k. The summation is done over the entire volume of the map. If the probability of occupancy of unlisted voxels is P_0 then the entropy of the occupied probability voxel list is

$$H(P) = -\left(\frac{M_v}{\varepsilon^3} - |P|\right) P_0 \ln P_0 - \sum_{i,j,k} P_{ijk} \ln P_{ijk} \tag{5}$$

where M_v is the volume of the corresponding occupancy grid, ε is the voxel size and P_{ijk} is a particular voxel occupancy probability. The number of occupied voxels in the list is $|P|$.

The occupied voxel list, M, stores the log odds of the probability therefore the probability that a listed map voxel is occupied is

$$P_{ijk} = \frac{1}{e^{-M_{ijk}} + 1} \tag{6}$$

This results in the final expression for the occupied voxel list entropy as

$$H(M) = \left(|M| - \frac{M_v}{\varepsilon^3}\right) P_0 \ln P_0 + \sum_{i,j,k} \frac{\ln\left(e^{-M_{ijk}} + 1\right)}{e^{-M_{ijk}} + 1}. \tag{7}$$

The main disadvantage of this metric is that it depends upon the dimensions of the probability array. Any increase in the size of the probability array results in an increase in the unmapped volume which, with its correspondingly high entropy, leads to an increase in the map entropy measure. An alternative method for assessing mapping performance is to consider the number of voxels required to represent

scans or maps at a particular resolution. Given a set of scans voxelised to a particular resolution, count how many voxels make up the map resulting from a combination of these scans. The algorithm that can generate the map with the fewest voxels can considered superior. An example of this assessment is included in Figure 5. In this way SLAM is recast as a compression problem. From the map, the sensor model and poses it should be possible to recreate a good approximation to the original scan voxel data for a static environment. Those mapping algorithms which produce lower global map voxel counts are better. This is because in order to achieve a good compression on a thoroughly explored environment, they must be able to encode repeated sections of raw data due to either the similar sections of the environment itself (perceptual aliasing) or due to data observed from a similar position. For a thoroughly explored environment in which the robot has a finite pose access space the map should converge to a finite number of voxels as the robot will not be able to adopt a pose from which it can perceive something new. Algorithms can then be compared when only an approximate ground truth is available.

3.2 Map Updating

Mapping and localisation may be modelled as a partially observable Markov decision process (POMDP). As it would be unwarranted to assume that the area under consideration can be completely observed from any one vantage point, under a first-order Markov assumption the map is assumed to contain the information revealed by all observations up to that point and so can be recursively updated.

For a particular map cell m and laser return cell l being either occupied in the map m or free in the map $\neg m$ then

$$p(m|l) = \left(1 + \frac{p(l|\neg m)p(\neg m)}{p(l|m)p(m)}\right)^{-1}. \tag{8}$$

The log odds approach to probabilities (Thrun et al, 2005) simplifies the update step and enhances the expressive power of the map which mostly contains very high probabilities (cells with probabilities under a threshold are removed from the map). The odds of an event A with probability $p(A)$ are

$$o(A) = \frac{p(A)}{p(\neg A)} = \frac{p(A)}{1 - p(A)}. \tag{9}$$

Thus (8) can be expressed in terms of log odds

$$\log o(m|l) = \log p(l|m) - \log p(l|\neg m) + \log o(m). \tag{10}$$

3.3 Scan Matching Probability

As new range scans are produced they need to be incorporated into the global map. One of the ways of finding the rotation and translation transformations necessary to merge scans, and thus the change in pose between them, is scan matching. This is also employed in localisation to determine the pose of the robot by matching the current range data to positions in a previously generated map.

The ideal matching algorithm returns not only an estimate of the pose, but also a probability distribution function describing the pose differences between successive scans. This estimate of the pose can be stored alongside the range data, together with the current map, and is composed of these pose-scan pairs. Storing the map in this way allows backward propagating corrections in the light of new observations. Unfortunately this violates the real-time requirement because, as more of these scans are taken, more data has to be processed. Ways of overcoming this drawback include allowing the scan data to decay over time or simply restricting processing to the time available.

As pointed out by Konolige and Gutmann (2000), when exploring areas already covered by the map care has to be taken when matching current poses with earlier ones and their associated range data. When aligning a scan with the map it is important to determine the significance of a particular overlap value in order to assess the reliability of that localisation. Determination of the confidence in the alignment pose is required for robust map building; only those scans with a high confidence value should be added to the map.

There are a number of methods for establishing the correlation metric between a scan at a hypothesised pose and the map. One as in Thrun (2001), the likelihoods fields approach is also sometimes referred to as the end point model. The model calculates the correlation as a function of the distance of the end point from the nearest obstacle in the environment. Whilst not physically correct it has proved to provide an adequate metric for many SLAM algorithms. There are two reasons for not incorporating this metric into the MROL algorithm. The first is execution speed and the second is that the reason it helps gradient descent type algorithms, is by smoothing out the pose probability distribution. However for the MROL algorithm it is better to have a sharper more discontinuous pose distribution as it tends to help it to isolate those poses that are better at coarser resolutions making it necessary to test fewer poses at the subsequent finer resolutions. This is because the correct poses stand out further and so at the pose culling stage more poses are removed. In this manner the smoothness of the pose probability distribution whilst required for many algorithms is neither necessary or desirable for MROL alignment. The shape of the probability distribution is useful in establishing this significance. For instance if the distribution is multimodal and the peaks are similar in height then there is doubt as to which modal peak is correct. Likewise the sharpness of the peaks indicates the accuracy of the corresponding

modal poses. These measures are all relative; they say nothing about the absolute value of the overlap which is addressed in the following analysis.

In order to determine significance of a given overlap, the probability of getting an overlap at least as high by chance alone must be considered. Let the volume covered by the map be M_v, the overlap o and the number of occupied voxels in the map and scan $|M|$ and $|N|$ respectively. The map and scan voxel sizes are both ε. The total number of possible voxels in the map volume is $|M_v| = M_v/\varepsilon^3$. Thus the proportion of the map that is occupied is

$$\frac{|M|}{|M_v|} = \frac{|M|\varepsilon^3}{M_v}. \tag{11}$$

Assuming the distribution of occupied voxels in the map is random, any particular voxel's chances of being occupied are independent of that of its neighbours and (11) expresses the probability of a single voxel picked at random being occupied. Given a scan of $|N|$ voxels what is the expected value of the overlap assuming that the scan voxel occupancy probabilities are independent? This can be re-expressed in standard probability terminology as picking $|N|$ voxels at random from the map without replacement. This means that the overlap counts may be modelled as a hypergeometric distribution the mean of which is given by $\bar{o} = |N||M|\varepsilon^3/M_v$, and the variance by

$$\sigma^2 = \frac{|N||M|}{|M_v|}\frac{|M_v|-|N|}{|M_v|-1}\left(1-\frac{|M|}{|M_v|}\right). \tag{12}$$

As an aid to simplification it is important to note that the hypergeometric distribution tends towards a normal distribution as $|N|/|M_v| \to 0$. The significance of the maximum overlap value may be determined by calculating the probability of its occurrence using the difference of o in standard deviations (12) from the mean. Figure 6 presents overlap distributions at a range of resolutions for a single scan matching.

3.4 Multi-Resolution Occupied Voxel Lists

Global localisation and scan matching are achieved by an approach that is similar to a multi-resolution particle filter. In this work a particle is considered as a pose with an associated probability weight w. In the implementation these weights, which are not normalised, are integers and considered to be proportional to the likelihood of the corresponding pose.

In Algorithm 1 the re-sampling step is akin to sampling importance re-sampling with a local uniform distribution. Typically in particle filters the initial distribution is randomly generated from a uniform distribution. The pose distribution is initialised by generating all poses at the coarsest resolution that lie within a region in pose space defined by an initial guess pose plus or minus a pose spread. The source of the guess pose may be odometry, inertial readings or simply the last pose of the robot. The spread pose is an indication of the error associated with the initial guess pose. The number of poses that need be generated at the initial coarse resolutions is greatly reduced by using large voxels as illustrated in Table 1. This is due to not only the reduction in translations that need to be tested but also the coarser representations which need to be rotated more before they differ at the alignment range, in this case 8m, and so fewer rotations are needed.

Algorithm 1 Multi-resolution sampling overview algorithm

```
OccupiedList map, OccupiedList scan, Integer factor f
generate uniform pose distribution sampleSet
for f is 2^i where i is 6 to 0 decrement 1 do
    down-sample map and local scan by f
    calculate overlaps for all poses in sampleSet
    remove from sampleSet all poses with certainty < 80% of the mode
    re-sample poses around remaining poses in sampleSet uniformly with width ε/2 spatially and Δθ/2
end for
```

The implementation of the particle set must satisfy two criteria. It must store only unique poses at the current spatial and angular resolutions and it must be able to iterate through the particle poses in angular order. The latter though is an optimisation consideration rather than a strict requirement for the success of the process. When calculating the overlaps, one of the slowest operations is the rotation step. Ordering the samples by orientation the rotation on the local scan can be done once only and then translated to poses with identical orientation at different positions. This is achieved by using a treeset structure which maintains the uniqueness and order of elements whilst guaranteeing insertion times of $O(\log n)$.

A typical multi-resolution distribution is visualised in Figure 1 where the local scan is globally aligned with the map. Both the local scan and the map are full 3D representations however, for the sake of clarity, they are shown as 2D projections in Figure 1 with ceiling and floor voxels removed. The initial particle distribution is uniform with a spatial separation of 1.28m. This initial resolution is dependent on the number of downsample levels and the final resolution, $1.28 = 0.02 \times 2^6$. The particles are re-sampled around the highest probability regions, progressively sampling the pose space at better resolutions as explained in Section 3.5. The final alignment is processed with a local scan and map resolution of 0.02m. This resolution is chosen because it is comparable to the random error inherent in

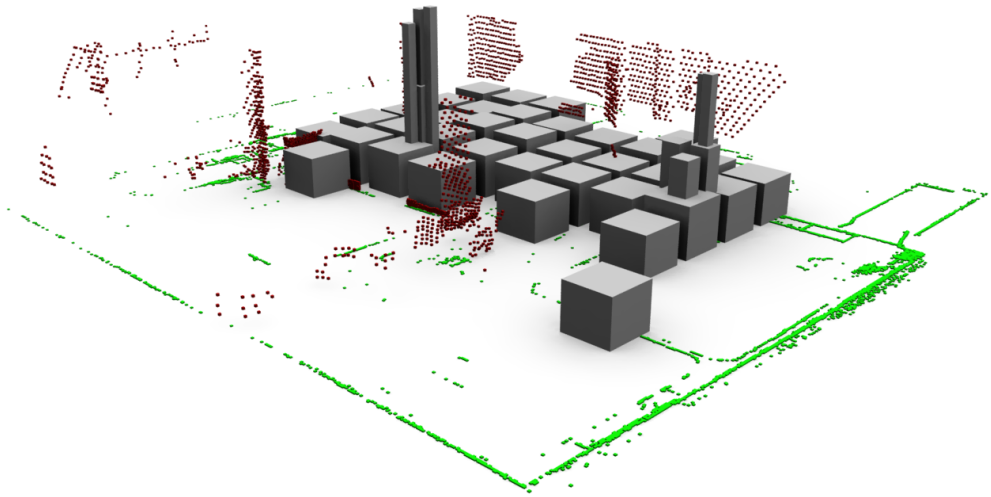

Fig. 1 Visualisation of scan and map with tested poses with $w > 0.8w_{max}$. Distribution of best orientation poses as a function of 2D position with height indicating probability of pose. The 3D scan consists of red voxels and the green voxels represent the map. Although the map is a full 3D map it has been projected onto the xy-plane to improve clarity of presentation. The multi-resolution nature of the solution is conveyed by the width of the pose blocks which give an indication of the 2D histogram of the pose distribution. Only those poses with $w > 0.8w_{\max}$ are shown.

the 3D laser scanner. The solution is indicated by the high cluster of poses shown in Figure 1.

The algorithm is similar to that for global localisation and tracking with the only difference being the first step of generating the initial particle distribution. For global localisation the particle distribution ranges over all possible pose space. The two main ways of generating the initial particle distribution are uniform and random. These two methods each have their advantages and disadvantages and in other work random initialisation has generally been preferred. However for this algorithm it is important to use a uniform initial particle distribution with the spatial and angular resolution of the poses carefully chosen. In this way it is possible to be sure of complete coverage of the pose space. Normally this would be prohibitively costly in terms of memory and computation time, but down-sampling makes it practicable. Other advantages of a uniform initial particle distribution are that it is deterministic and quick to generate.

The spatial Δx and angular $\Delta\theta$ resolutions need to be matched and are dependent upon the resolution ε of the occupied voxel lists and the scan range R. The spatial separation of the particles is dictated by the resolution of the occupied voxel list being processed. As $\Delta x = \varepsilon$ the angular separation is

$$\Delta\theta = \varepsilon / R \tag{13}$$

where R is the maximum range of points under consideration, measured from the origin of the local map, and ε is the voxel size. By effecting angles of rotation greater than (13) one ensures that successive rotations produce new local occupied voxel lists at the current resolution.

Because the particles are stored in sets, then upon refinement the particle distribution is still guaranteed to contain unique particles. Therefore no computational power is wasted in calculating the weighting for duplicate samples generated by pose refinement (as described in Algorithm 4). An important aspect of this process is the down-sampling of the occupied voxel lists. The 3D laser scanner returns 3D scans as point clouds and these are converted into an occupied voxel list of resolution $\varepsilon = 0.02$m.

The down-sampling process is described in Algorithm 2 and illustrated in Figure 2. In Figure 2 the point cloud and grid are represented in 2D for diagrammatic simplicity. In Algorithm 2 down-sampling is done at many resolutions however, to aid clarity, only two resolution grids are shown in Figure 2 and Figure 3. In Figure 2 the crosses represent laser return points and two resolution grids are superimposed. The number of points within each cell of the corresponding grid is labelled with numbers in the larger font corresponding to the grid with the larger cells.

A visualisation of the 3D down-sampling of a typical scan is depicted in Figure 3. This down-sampling mechanism is chosen rather than the more conventional one, in which voxels are simply subdivided. The main reason is due to the problem of simply subdividing the voxels with no offset. If the space is divided in such a manner then two points could lie on either side of a voxel boundary and, although they are very close, they would never be considered to be in the same voxel regardless of the voxel size. By down-sampling with an offset, points that are close will be registered in the same voxel at some resolution.

Down-sampling can take time, especially in the case of a large occupied voxel list representing a map which usually contains many more voxels than a local scan. In some instances, for example repeated localisation within a map, down-sampling needs to be performed only once for the map

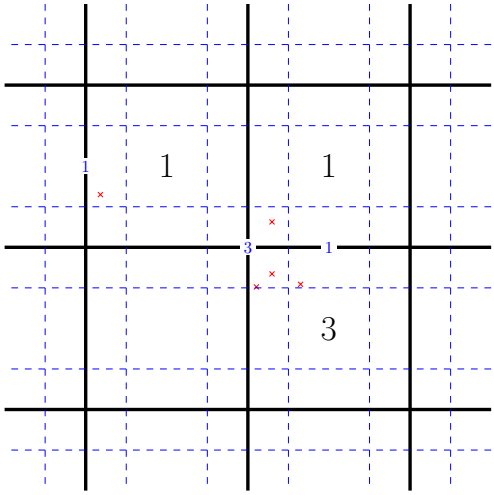

Fig. 2 The quantisation process for converting point clouds to the occupied voxel lattice. In the diagram there are two grids superimposed upon one another with resolutions differing by a factor of two.

as multiple localisations using the map do not have to recalculate it. This is especially useful if there are a number of scans to be merged into the map, in which case it is better to test them all and add the one with the highest overlap first.

Algorithm 2 Occupied voxel list down-sample algorithm.

```
occupiedList L, integer factor f
new resolution ε = resolution L*f
if occupiedList resolution ε exists in cache then
    return corresponding downSampled occupiedList
end if
new occupiedList l with resolution f*resolution of L
for all voxels in L do
    add new voxel(x/f, y/f, z/f) to l
end for
store f with l in cache
```

Once the occupied voxel lists have been down-sampled and the particles selected the probabilities or weights for each particle are estimated. This estimation, as it is carried out for each particle, is a fine balance between accurate weighting and computational speed. The process is described in Algorithm 3 and may be succinctly expressed as

$$o = |L \cap M|. \tag{14}$$

A single scan has a spatially uniform distribution of points. There is a higher density of points closer to the origin and so this has to be corrected by applying an R^{-2} weighting factor in 3D and R^{-1} for 2D to the scan points during the overlap calculation. The overlap count is o, the number of elements or cardinality of set A is $|A|$, the map is M and the local transformed scan is L. Thus o is not only a count of the overlapping voxels but also an indicator of the probability of the associated pose particle. A statistical analysis of the significance levels for o in various situations is presented in Section 3.3.

Algorithm 3 Process for updating the weights of the sample set.

```
sampleSet poseSample, occupiedList M, occupiedList local
for each sample in poseSamples do
    o = 0
    if sample pose angle is different from previous then
        calculate new rotated local L
    end if
    translate L by (p_x, p_y)
    for each voxel in L do
        if voxel is present in M then
            increment o
        end if
    end for
    set sample weight to o
    if o > o_max then
        o_max = o
    end if
end for
```

It is assumed that $|L| < |M|$ and so iteration is carried out over the elements of L. The particles are stored in a set sorted primarily by their angle. Transformations are performed by rotation matrix multiplication for the rotations and component wise addition for the translations. Rotation operations on the local occupied voxel list are slower than translation operations. Rotations in 2D Cartesian coordinates require four multiplications and two additions compared with translations which require two additions. This processing becomes even more demanding when the alignment algorithm is extended to allow rotations due to uneven floor surface. Sorting the list by gathering groups with similar rotations together, and accompanying this with a check to see if the rotation of the next particle is the same, guarantees that rotations are only performed once.

3.5 Pose Refinement

Once the particle weights have been updated, particles with a weight less than the modal fraction for that resolution of the maximum are removed. Example modal fractions are in Table 1. New particles are then generated around the remaining high-weight particles as described in Algorithm 4.

This new particle set is then submitted for the next round of tests at better resolution. In this manner about 10,000 particles per second can be assessed on a 2GHz computer, the precise rate depending upon the resolution and structure of the map and the local occupied voxel lists. The rate may be calculated for a given resolution by dividing the number of lookups per second by the product of the scan voxels at that resolution and the poses. The lookups per second is the number of times a single voxel can be transformed to a

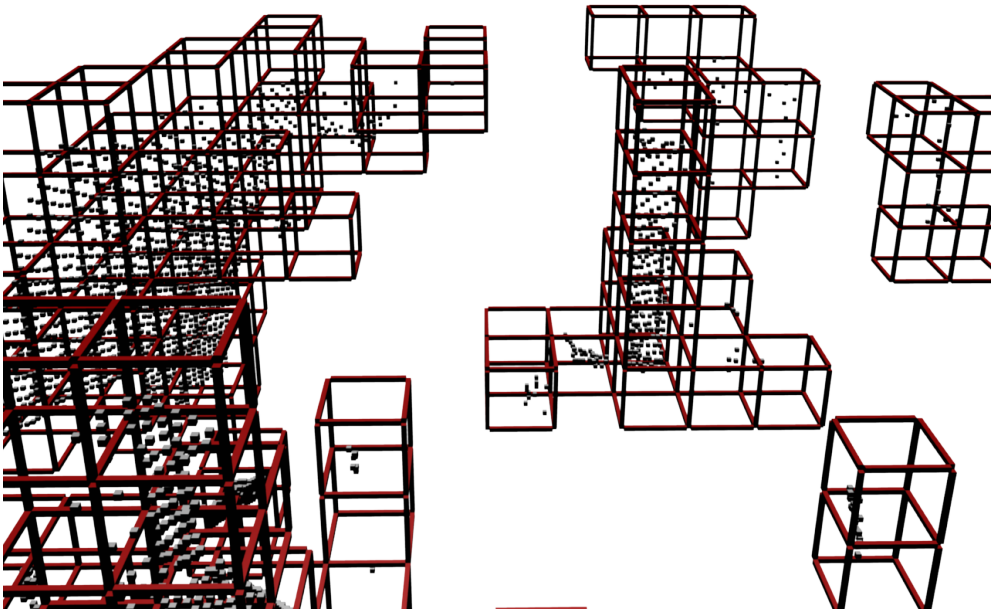

Fig. 3 Illustration of the down-sampling of a typical scan.

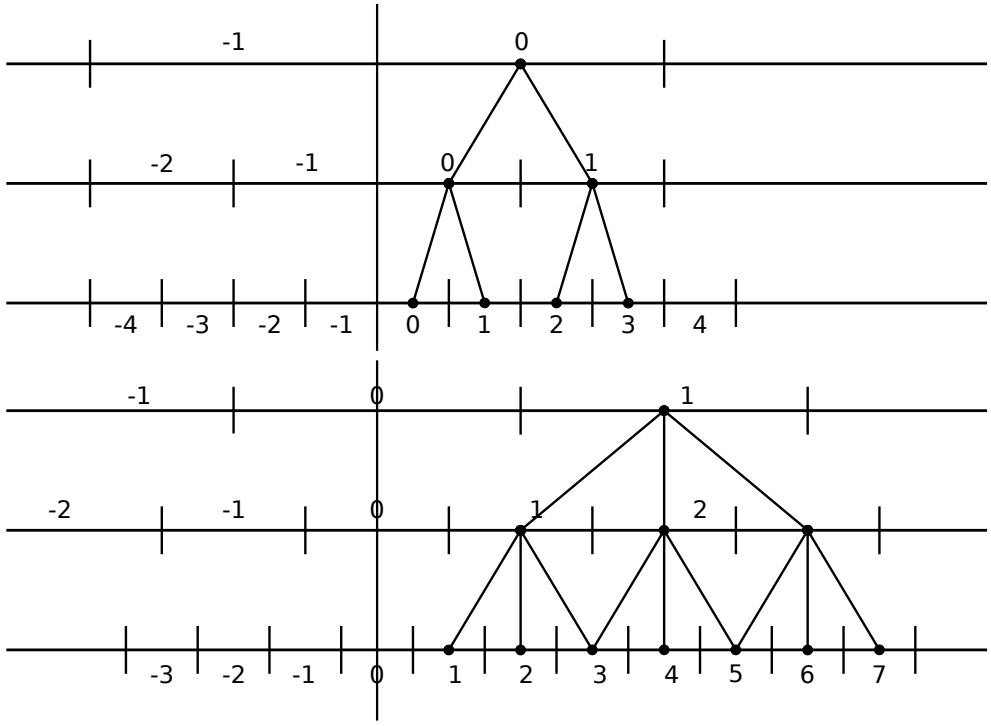

Fig. 4 Discrete single variable refinement propagated through multiple resolutions. The two quantization methods considered $\lfloor x/\varepsilon \rfloor$ and $\lfloor x/\varepsilon + 0.5 \rfloor$, of which the latter corresponds to conventional rounding.

Algorithm 4 Re-sampling at improved spatial and angular resolution around high weight samples. This algorithm will produce repeated samples for adjacent initial samples, however samples are placed in a set to ensure uniqueness.

```
sampleSet initialSamples
integers a,b,c
create new sampleSet P
for each sample in initialSamples do
    for −1 ≤ a ≤ 1 do
        for −1 ≤ b ≤ 1 do
            for −1 ≤ c ≤ 1 do
                new sample (x+aε/2, y+bε/2, θ+cΔθ)
                add sample to P
            end for
        end for
    end for
end for
```

new position and the resulting voxel presence checked in the map.

The pose refinement process in one dimension as it propagates through the multiple resolutions is illustrated in Figure 4 for two quantisation methods. The first quantisation method, $\lfloor x/\varepsilon \rfloor$, is quicker to compute. The second $\lfloor x/\varepsilon + 0.5 \rfloor$ is mathematically more convenient as it is symmetric about $x = 0$. However there is a stark difference in the number of children each particle produces at the next resolution. For a one dimensional variable the difference between two and three child particles is small but for a pose with 6 degrees of freedom (DOF) this becomes $2^6 = 64$ as compared to $3^6 = 729$. A problem with splitting into two is that for poses constrained to a plane, typical in indoor robotic operations, the returned poses will not be exactly aligned or coincident with the *xy*-plane.

4 Experimental Results and Analysis

Experiments are performed in various indoor environments, an office, a corridor and a lab.

Figure 5 plots the evolution of map voxel counts as each scan is incorporated into the map. The counts for the correct mapping solution, resulting in the map in Figure 10, are shown. Nearby is the plot for when the scan voxel count is capped at 1000. This results in a fivefold speed increase in the mapping of this data set. When the scan voxel count exceeds this cap value at a particular resolution a random sample of the cap value scan voxels is selected for alignment checking against the map. As can be seen from the plots in Figure 5 this has very little impact on the accuracy of the map produced. For context the data was processed and errors deliberately induced into the final alignment of the mapping algorithm and these can be clearly seen where the incorrect map voxel counts grows faster than the correct solution.

The actual distributions of overlaps for a number of resolutions are shown in Figure 6 in which the mean overlap stays relatively constant regardless of resolution, however the standard deviation grows as the resolution becomes finer. What is also clear is that the overlap count may be adequately represented by a normal distribution down to a resolution of 0.64m. The region of interest is the far right part of this graph where the significance of the maximum overlap may be established. The maximum count for the 0.16m resolution scan alignment shown in Figure 6 is 379 which is very significant and means the result may be regarded as reliable.

Independence of voxel occupancy was one of the original assumptions underlying this treatment. This assumption becomes increasingly valid as ε, the voxel size, increases. The impact of this assumption is to increase the significance level of the overlap by enlarging the spread of the distribution. Clustered voxels are more likely to produce higher overlaps. According to the distribution, assuming voxel independence, the maximum count of 379 for 0.16m has a vanishingly small probability of occuring by chance alone for random uniform spatial distributions of voxels. It is therefore safe to conclude that this is an actual match rather than one arising by chance.

The results of mapping a typical indoor office environment are displayed in Figure 7 and Figure 8. The data is gathered by the enhanced laser scanner developed in Ryde and Hu (2008). The map is created by aligning 37 scans without odometry. Once the map shown in Figure 8 is created global localisation of different scans is performed within this map. The robot is assumed to be completely lost and therefore the entire area (approximately 200m^2) is searched. Locating the 7000 voxel scan within the 200,000 voxel map takes 5 seconds on a 2GHz processor.

The high data quality along with the almost omnidirectional field of view leads to good localisation and mapping performance. The occupied voxel list map in Figure 8 has a resolution of 0.02m and the position error is 0.03m. The error in the orientation may be estimated by noting that the misalignment between the corridor and the office is less than 1 degree. The alignment is of comparable quality to the manual alignment technique making the goal of further improvements in accuracy difficult.

4.1 Comparison with ICP

In order to test the effectiveness of the MROL algorithm, the same data set is processed by both MROL and an ICP implementation. The 3D ICP implementation is an open source one from the Mobile Robotics Programming Toolkit (MRPT) (González et al, 2009) version 0.6.5. The resulting inter-scan pose transforms produced are compared against the ground truth. The true inter-scan pose transforms are obtained by

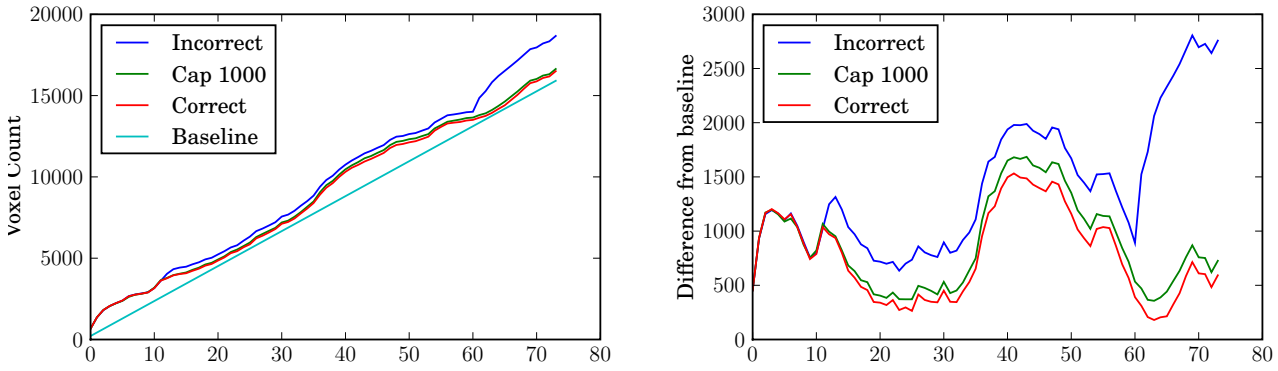

Fig. 5 Map voxel counts at a resolution 0.2m for the QCAT data set as a function of scan number. The voxel count for the correct solution as well as one with deliberately induced errors are shown. The third data is for mapping with the local voxel count capped to 1000 voxels. Also plotted is the count with the baseline subtracted to better illustrate the differences.

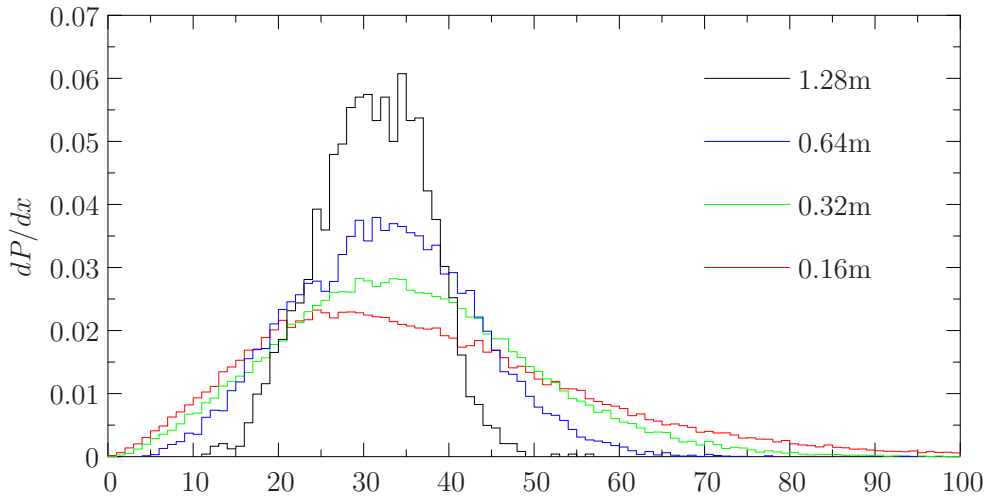

Fig. 6 Probability distribution of overlap counts for various resolutions of the map and local scan.

manually aligning all scans. This Queensland Centre for Advance Technology (QCAT) data set consists of around 80 3D scans taken in a stop and go manner inside an office environment with scans being taken every 2m or so. The resulting maps from this data set are displayed in Figure 10 from which it can be seen that the environment consists of a mixture of empty corridors, cluttered rooms, a small loop, orthogonal and non-orthogonal walls. This variety makes the data set a good test set for mapping algorithms. These maps are generated using the MROL algorithm on the raw scans using initial poses guesses which have plus or minus errors of 0.5m and 45 degrees. The environment was also dynamic during the experiment mainly due to the motion of people but this does not seem to have a detrimental impact on the effectiveness of the mapping approach presented.

The nature of ICP mean it is more suited for scan to scan matching rather than scan to map matching for the comparison. Normally MROL is best executed in scan to map matching mode. In this manner it can close small loops naturally and is generally more reliable as the intersection of a scan with a map is likely to be higher than that with the previous scan alone. Due to this nature of ICP and in the interests of a fair comparison both algorithms are executed in scan to scan matching mode.

The point clouds from the 3D scans are voxelised to a resolution of 0.05m. Both algorithms are given point lists of these voxel coordinates. Interestingly it is observed that ICP performs much better on the voxel coordinates than on the original raw scan point clouds. It is believed that this is due to the R^{-2} point density dependence typical of point clouds originating from 3D laser scanners. In this way the alignment tends to favour placing the two regions of dense points in both scans near each other rather than the correct alignment. Voxelising the point clouds in this way also accelerates the ICP algorithm as it has less points to consider.

Rather than presenting a single metric for the accuracy of the ICP versus MROL algorithms plots of the position and orientation errors have been generated in Figure 9. For these

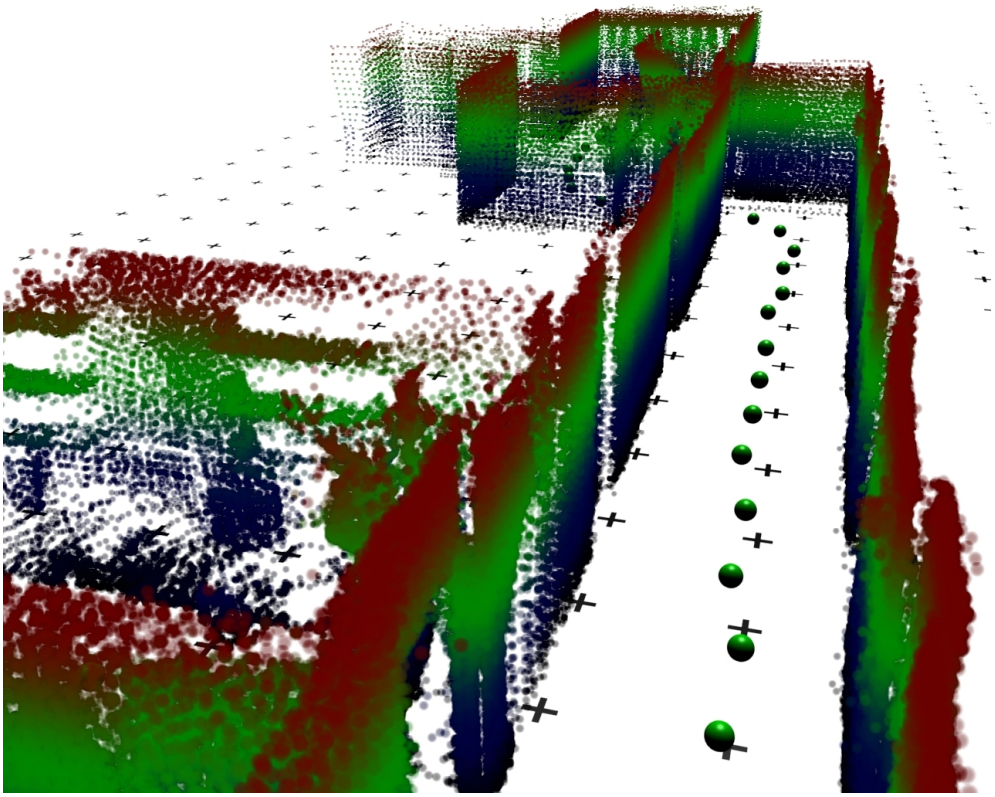

Fig. 7 Corridor view demonstrating the mapping ability even in a relatively featureless corridor. The green spheres are the observation positions and the map voxels (0.02m) are coloured by height.

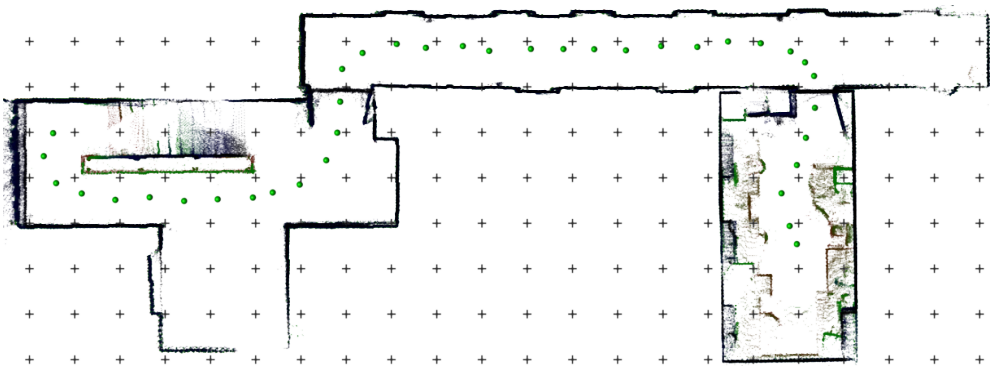

Fig. 8 Overhead view of a 3D map containing a stair well, corridor and office at a resolution of 0.02m. The floor and ceiling points have been removed for clarity. The positions of the robot are shown as green spheres. The points are coloured by height and the grid points are at 1m intervals. No odometry is required to build this map.

graphs both algorithms were given the same initial guess poses and the same point data. From this the inter-scan pose transform was estimated by ICP and MROL with 3 and 4 degrees of freedom. The MROL 3DOF estimated the x, y translations and the change in heading, whilst the MROL 4DOF estimated the z translation in addition. These plots show the sorted errors and times in ascending order similar to a cumulative frequency curve aiding a direct comparison of the error and execution time distributions of the ICP and MROL algorithms

As can be seen from these plots the MROL approach is more reliable than ICP returning poses close to the correct pose with only one misalignment out of 75. These graphs reflect a relatively well known problem with ICP that it can have a small region of convergence. If the initial pose estimate is within this region of convergence the ICP result will be very accurate. Once the initial pose estimate drifts outside of the region of convergence ICP tends to fail abruptly. The MROL approach does not have a region of convergence, as such, as it starts by testing all poses uniformly at a coarse resolution across the given pose range. The execution per-

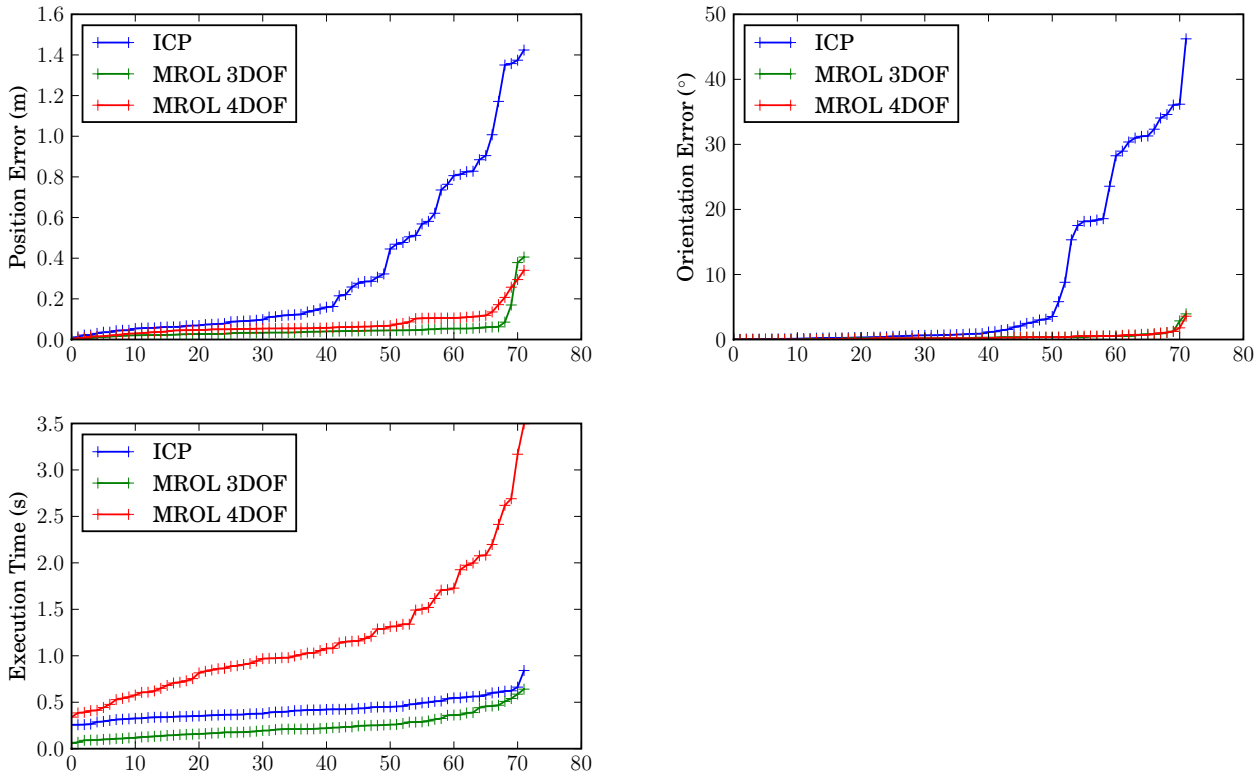

Fig. 9 Sorted position errors, orientation errors and execution times for both MROL and ICP algorithms sequential scan matching the QCAT data set (Figure 10). The initial pose estimate error was $\pm$1m in the x and y directions and $\pm$90 degrees for the heading.

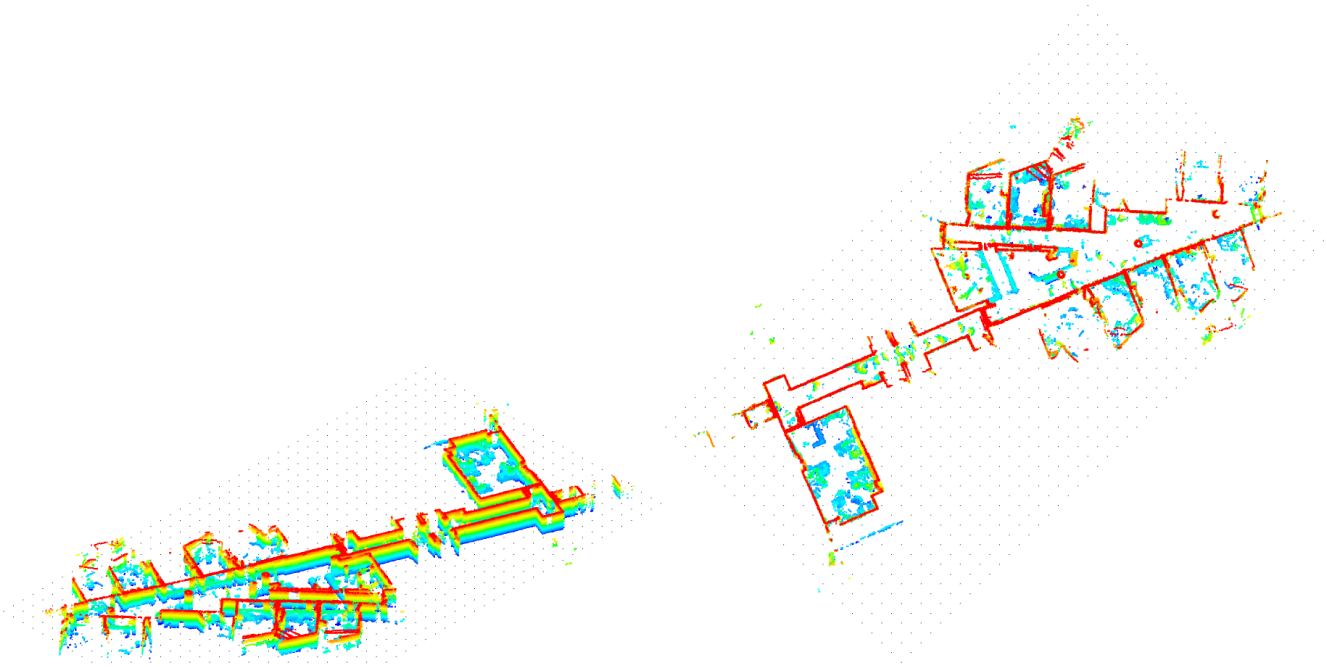

Fig. 10 Isometric and overhead views for the MROL generated map of the QCAT data set. Voxels are coloured by height and the grid points separated by 1m.

Res. (m)	Modal Frac.	Map Voxels	Scan Voxels	Poses Before	Poses After	Time (s)
0.8	0.6	134	122	68	12	0.0030
0.4	0.6	362	384	136	11	0.011
0.2	0.6	800	982	147	14	0.028
0.1	0.7	1836	994	196	14	0.039
0.05	0.8	3625	996	231	6	0.046

Table 1 Example output for aligning within in a pose region of $\pm$1m and $\pm$90 degrees. Pose counts are recorded for both the number of poses tested and the remaining poses after rejection of poses with overlaps less than the modal fraction.

formance of the MROL 3DOF compares favourably with ICP being somewhat three fold faster. However as the number of degrees of freedom is increased the MROL algorithm slows. This highlights perhaps the biggest disadvantage of this MROL based approach; its execution time is sensitive to the number of degrees of freedom that need to be estimated. However it is often possible to estimate the roll and pitch of a robot with reference to the acceleration due to gravity. Thus only four degrees of freedom need to be calculated, for which the MROL performance is acceptable. Figure 10 displays the map generated by the MROL algorithm for the QCAT data set. Colouring by height shows how the architectural structure may be separated from clutter and transient objects such as humans in the environment which tend to be lower heights. The map is generated by the MROL algorithm matching each scan to the previous map and adding that scan into the map as opposed to the results in Figure 9 which involves matching each scan to the previous scan. In this way localisation accuracy is improved because there is higher overlap between the scan and the map. Listed in Table 1 are various statistics for the MROL alignment of two typical 3D scans. The first column lists the voxel size for both the map and scan. The modal fraction used for that resolution is also present. The numbers of voxels in the map and scan respectively are shown along with the number of poses before and after rejection by the modal fraction. Finally the time taken analysing each resolution indicates the distribution of computational effort amongst the different resolutions. The execution time at each resolution depends on the number of poses that need to be tested and the number of scan voxels. The time does not depend on the map voxel count as the hash structure means that look ups take the same amount of time irrespective of the number of map voxels. In this case if the number of scan voxels exceeds 1000 then a sample of around 1000 is selected. This is an optimisation that is not necessary for correct operation of the algorithm but enhances its operation speed with negligible impact on accuracy as shown in Figure 5.

5 Conclusion

This paper concentrates on single robot mapping with a custom 3D laser sensor (Ryde and Hu, 2008) and offers two main innovations. The first is the new way to represent 3D probabilistic map data referred to as a multi-resolution occupied voxel list. Although previous work has extended 2D occupancy grids to multi-resolution and, to some extent 3D, our work applies this idea to occupied voxel lists which themselves have never been formally described. The space quantisation for converting point clouds to multi-resolution occupied voxel lists is different in that it applies a half cell offset before quantising at the next resolution.

The second innovation is the description of a method similar to a multi-resolution particle filter for the rapid alignment of two occupied voxel lists. The main difference here is that the sampling is uniform and exhaustive, and consequently the subsequent behaviour is deterministic which improves repeatability, reliability and testability. These two innovations have enabled a single mobile robot to move through a relatively disordered environment and generate an internal 3D representation of the surroundings accurate to better than 0.02m.

The algorithm for aligning 3D occupied voxel lists is presented along with a theoretical treatment that gives an estimate of the significance of particular overlap values. This analysis is vital for determining whether scans should be added to the map as well as in detecting loop closures. The conclusion of this theoretical analysis is tested experimentally for a range of occupied voxel list resolutions. It is found that the agreement with experiment improves as the occupied voxel list voxel size increases. It is postulated that the reason for this is the increasing validity of the assumption of voxel independence with growing voxel size.

This paper also addresses the justification for equating occupancy with the integer number of times the voxel has been observed to contain a scan point. This is done by assuming voxel independence starting from the Bayesian update and applying the Markov assumption. The resulting occupancy probability update equation is re-expressed in terms of log odds probability and from this the validity of an additive voxel update is apparent. Under these conditions the occupied voxel list integer values must be considered as proportional to the log odds of the occupancy probability.

Experiments testing the validity of this approach to 3D mapping are conducted in a variety of indoor environments, some of which are extremely cluttered and others featureless. The idea of extracting cross-sections from the 3D map data is presented with an illustration of the reduction in environmental complexity as compared to that for a floor based robot. It is shown that maps of indoor areas above a certain height are good representations of the architectural structure of the surroundings. It is proposed that such maps are more

suitable for localisation over the longer term since they exclude most moving objects.

Finally measures for establishing the mapping performance of an algorithm were presented. The entropy of occupied voxel lists is derived. The second measure presented is the measure of the voxel count of the global map at a given resolution. When the accessible environment is thoroughly covered this is a quantitative measure of map quality with lower global map voxel counts indicating better performance. Recasting the SLAM problem in this way as one of compression allows the comparison of different mapping algorithms and methods.

Future work is likely to entail further speeding up the MROL based algorithm by implementing the rate limiting steps in C or in a parallel manner on a graphics processing unit. The maintenance of the map over time also needs to be considered especially for dynamic environments. The comparison with ICP indicates a possible direction of a hybrid MROL-ICP approach in which the coarse alignment is done with MROL and the fine adjustment with ICP. Finally in order to produce fully automatic mapping path planning over MROL a solution to the next best view problem would need to be implemented to enable autonomous exploration.

References

Burgard W, Fox D, Hennig D, Schmidt T (1996) Estimating the absolute position of a mobile robot using position probability grids. In: AAAI/IAAI, Vol. 2, pp 896–901

Burgard W, Derr A, Fox D, Cremers A (1998) Integrating global position estimation and position tracking for mobilerobots: the Dynamic Markov Localization approach. In: Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems

Burgard W, Cremers AB, Fox D, Hähnel D, Lakemeyer G, Schulz D, Steiner W, Thrun S (1999) Experiences with an interactive museum tour-guide robot. Artificial Intelligence 114(1–2):3–55

Burgard W, Moors M, Stachniss C, Schneider F (2005) Coordinated multi-robot exploration. IEEE Transactions on Robotics 21:376–386

Choset H, Nagatani K (2001) Topological simultaneous localization and mapping (SLAM): toward exact localization without explicit localization. IEEE Transactions on Robotics and Automation 17(2):125–137

Cole D, Newman P (2006) Using laser range data for 3D SLAM in outdoor environments. In: Proceedings of the IEEE International Conference on Robotics and Automation (ICRA), pp 1556–1563

Dellaert F, Burgard W, Fox D, Thrun S (1999) Using the condensation algorithm for robust, vision-based mobile robot localization. In: Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition

Elfes A (1987) Sonar-based real-world mapping and navigation. IEEE Journal of Robotics and Automation 3(3):249–265

Elfes A (1991) Dynamic control of robot perception using stochastic spatial models. In: Schmidt G (ed) Information Processing in Mobile Robots, invited Paper presented at the International Workshop on Information Processing in Mobile Robots, TechnischeUniversitat Munchen, Germany, March 6 - 8, 1991.

Fox D (1998) Markov localization: A probabilistic framework for mobile robot localization and navigation. PhD thesis, University of Bonn

Fox D (2003) Adapting the sample size in particle filters through KLD-sampling. International Journal of Robotics Research 22(12):985–1003

Fox D, Burgard W, Dellaert F, Thrun S (1999) Monte carlo localization: Efficient position estimation for mobile robots. In: AAAI/IAAI, pp 343–349

González J, Blanco J, Galindo C, Ortiz-de Galisteo A, Fernández-Madrigal J, Moreno F, Martínez J (2009) Mobile robot localization based on ultra-wideband ranging: A particle filter approach. Robotics and Autonomous Systems 57(5):496–507, DOI http://dx.doi.org/10.1016/j.robot.2008.10.022

González-Baños H, Latombe J (2002) Navigation strategies for exploring indoor environments. The International Journal of Robotics Research 21(10-11):829–848

Gustafsson F, Gunnarsson F, Bergman N, Forssell U, Jansson J, Karlsson R, Nordlund P (2002) Particle filters for positioning, navigation and tracking. IEEE Transactions on Signal Processing

Hager G, Burschka D (2000) Laser-based position tracking and map generation. In: Proceedings of the IEEE International Conference on Robotics and Automation (ICRA), pp 149–155

Hähnel D, Schulz D, Burgard W (2002) Map building with mobile robots in populated environments. In: Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)

Howard A, Wolf D, Sukhatme G (2004) Towards 3D mapping in large urban environments. In: Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Sendai, Japan, vol 1, pp 419–424

Hu H, Gu D (2000) Landmark-based navigation of industrial mobile robots. Industrial Robot: An International Journal 27(6):458–467

Jensfelt P, Austin D, Wijk O, Andersson M (2000) Feature based condensation for mobile robot localization. In: Proceedings of the IEEE International Conference on Robotics and Automation (ICRA), pp 2531–2537

Jeong W, Lee K (2005) CV-SLAM: a new ceiling vision-based slam technique. In: Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), pp 3195–3200

Konolige K, Gutmann S (2000) Incremental mapping of large cyclic environments. International Symposium on Computer Intelligence in Robotics and Automation (CIRA) pp 318–325

Kuipers B, Beeson P (2002) Bootstrap learning for place recognition. In: Proceedings of the AAAI Conference on Artificial Intelligence, Edmonton, Canada, pp 174–180

Kümmerle R, Triebel R, Pfaff P, Burgard W (2008) Monte carlo localization in outdoor terrains using multilevel surface maps. J Field Robotics 25(6-7):346–359

Lee D (1995) The map-building and exploration strategies of a simple, sonar-equipped, mobile robot; an experimental, quantative evaluation. PhD thesis, University College London

Lenser S, Veloso M (2000) Sensor resetting localization for poorly modelled mobile robots. In: Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)

Lim J, Cho D (1992) Physically based sensor modelling for a sonar map in a specular environment. In: Proceedings of the IEEE International Conference on Robotics and Automation (ICRA), Nice, France, pp 1714–1718

Lingemann K, Surmann H, Nüchter A, Hertzberg J (2004) Indoor and outdoor localization for fast mobile robots. In: Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Sendai, Japan, vol 3, pp 2185–2190

Montemerlo M, Thrun S (2004) Large-scale robotic 3D mapping of urban structures. In: Proceedings of the International Symposium on Experimental Robotics (ISER), Singapore

Moravec H (1989) Certainty grids for sensor fusion in mobile robots. Sensor Devices and Systems for Robotics pp 253–276

Nüchter A, Lingemann K, Hertzberg J, Surmann H (2005) Heuristic-based laser scan matching for outdoor 6D SLAM. In: Advances in Artificial Intelligence. 28th annual German Conference on AI, Koblenz, Germany, pp 304–319

Olson C (1999) Subpixel localization and uncertainty estimation using occupancy grids. In: Proceedings of the IEEE International Conference on Robotics and Automation (ICRA), Detroit, USA, pp 1987–1992

Olson C (2000) Probabilistic self-localization for mobile robots. IEEE Transactions on Robotics and Automation 16(1):55–66

Pfaff P, Triebel R, Burgard W (2007) An efficient extension to elevation maps for outdoor terrain mapping and loop closing. The International Journal of Robotics Research 26(2):217–230

Rocha R (2005) Building volumetric maps with cooperative mobile robots and useful information sharing: a distributed control approach based on entropy. PhD thesis, Faculty of Engineering of University of Porto, Portugal

Rocha R, Dias J, Carvalho A (2005) Cooperative multi-robot systems: a study of vision-based 3-D mapping using information theory. Robotics and Autonomous Systems 53:282–311

Rocha R, Ferra F, Dias J (2008) Multi-robot complete exploration using hill climbing and topological recovery. In: Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), pp 1884–1889

Ryde J, Hu H (2007) Mobile robot 3D perception and mapping with multi-resolution occupancy lists. In: Proceedings of IEEE International Conference on Mechatronics and Automation (ICMA 2007), Harbin, Heilongjiang, China

Ryde J, Hu H (2008) 3D laser range scanner with hemispherical field of view for robot navigation. In: Proceedings of IEEE/ASME Advanced Intelligent Mechatronics International Conference on Mechatronics and Automation (AIM 2008), Xi'an, China

Stachniss C, Burgard W (2003a) Exploring unknown environments with mobile robots using coverage maps. In: Proceedings of the International Conference on Artificial Intelligence (IJCAI), Acapulco, Mexico, pp 1127–1132

Stachniss C, Burgard W (2003b) Mapping and exploration with mobile robots using coverage maps. In: Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Las Vegas, USA, vol 1, pp 476–481

Strand M, Erb F, Dillmann R (2007) Range image registration using an octree based matching strategy. In: Proceedings of IEEE International Conference on Mechatronics and Automation (ICMA 2007), Harbin, Heilongjiang, China, pp 1622–1627

Thrun S (2001) A probabilistic online mapping algorithm for teams of mobile robots. International Journal of Robotics Research 20(5):335–363

Thrun S, Montemerlo M (2006) The GraphSLAM algorithm with applications to large-scale mapping of urban structures. The International Journal of Robotics Research 25(5/6):403–430

Thrun S, Burgard W, Fox D (2005) Probabilistic Robotics. MIT Press

Triebel R, Pfaff P, Burgard W (2006) Multi-level surface maps for outdoor terrain mapping and loop closing. In: Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Beijing, China, pp 2276–2282

Vandorpe J, van Brussel H, Xu H (1996) Exact dynamic map building for a mobile robot using geometrical prim-

itives produced by a 2D range finder. In: Proceedings of the IEEE International Conference on Robotics and Automation (ICRA), Minneapolis, USA, pp 901–908

Victorino A, Rives P, Borrelly J (2003) Safe navigation for indoor mobile robots. The International Journal of Robotics Research 22(12):1005–1039

Wang L, Yong L, Jr MA (2002) Mobile robot localisation for indoor environment. SIMTech Technical Report

Wang Z, Huang S, Dissanayake G (2007) D-SLAM: A decoupled solution to simultaneous localization and mapping. The International Journal of Robotics Research 26(2):187–204

Weingarten J, Gruener G, Siegwart R (2004) A state-of-the-art 3D sensor for robot navigation. In: Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), vol 3, pp 2155–2160

Wetzler C, Weiss G, von Puttkamer E (1994) Keeping track of position and orientation of moving indoor systems by correlation of range-finder scans. In: Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), pp 595–601

Wulf O, Arras K, Christensen H, Wagner B (2004) 2D mapping of cluttered indoor environments by means of 3D perception. In: IEEE/RAS Int. Conf. on Robotics and Automation (ICRA), New Orleans, USA, pp 4204–4209

Yguel M, Aycard O, Laugier C (2005) Wavelet occupancy grids: a method for compact map building. In: Proceedings of the International Conference on Field and Service Robotics (FSR), Port Douglas, QLD, Australia, pp 219–230