Robust Distinctive Place Recognition for Topological Maps

T. Bailey, E. M. Nebot, J. K. Rosenblatt and H. F. Durrant-Whyte

Australian Centre for Field Robotics (ACFR)

Department of Mechanical and Mechatronic Engineering

The University of Sydney NSW 2006

e-mail: tbailey/nebot/julio/hugh @mech.eng.usyd.edu.au

Abstract

Topological maps provide a compact and flexible method for mobile robot navigation without the requirement of high precision localisation or pre-planned trajectories. They represent an environment as a graph where each node of the graph is a distinctive place and each edge describes the path between two distinctive places. A distinctive place is a location in the environment which is distinguishable from other places on the basis of patterns observable in sensory data.

The main difficulty with topological maps is that reliable distinctive place recognition has been hard to attain except in very simple, structured environments. Even in these environments, failure to recognise a place and false place recognition have been major problems.

This paper discusses a method for recognising distinctive places, defined here as a set of features observed from a particular vehicle pose (position and orientation). Distinctive place recognition is achieved by extracting features from sensory data and matching their relative geometry to the relative geometry of the features stored in the topological map. If the observed features match those of a distinctive place, then the pose of the robot relative to the distinctive place pose is determined.

Experimental results obtained with a 2D scanning laser on a mobile robot platform demonstrate that this method is robust to dynamic objects, occlusions and varied viewpoints.

1 Introduction

In mobile robot navigation, there exist two main techniques for enabling robot localisation within an environment. The first is to maintain a very accurate global estimate of the robot's pose. This has been implemented using various methods such as evidence-grids [11]; feature tracking [4]; and Iterative Closest Points [8]. The problem with all of these methods is that they are quite fragile to errors in the location estimate and tend to fail catastrophically if the estimated location drifts too far from the true location. This is usually due to a failure to perform correct data association (i.e., to associate newly observed features with previously observed features).

The second technique is to localise via a topological map. Topological maps are a graph-based description of an environment comprising of nodes and connecting edges or paths. Each node of the map is a *distinctive place* [7] in the environment and the connecting paths between nodes are sets of behaviours that will enable the vehicle to travel from one distinctive place to the next. These maps have the advantages of compactly representing the environment in an intuitive format and being quite robust to inaccuracies in the robot's pose estimate. Topological maps tend to fail, however, if they are unable to recognise nearby distinctive places [13] or if they match a node to the wrong place. This may be because of dynamic objects in the environment or perhaps simply that the robot is observing the place from a different viewpoint.

Most recent research has restricted distinctive place definition to a set of models of very simple indoor places. Kuipers and Byun [7] define a distinctive place type by a set of elementary rules such as "equal distance to near objects". The robot moves into the vicinity of the place and then moves via a hill-climbing algorithm to the point where the highest distinctiveness measure is attained from its sonar sensors. This method was tested only in a static, simulated environment. Aycard *et al.* [1] use second order Hidden Markov Models to learn distinctive place types such as open doors, corridors and T-intersections. The robot must move past a place to recognise it from the data sequence of its sonar sensors. Sequence matching, however, was susceptible to dynamic obstacles and changes in the robot's viewpoint. Kortenkamp and Weymouth [6] also rely on simple models of place types. They specify that a place must be a *gateway* between two areas (e.g., a door or T-intersection). Their sonar-based place recognition is augmented by a vision sensor, which matches visual cues with an image taken previously at the place location. Owen and Nehmzow [9] avoid using distinctive place models and, instead, define a place by a vector map of the landmarks seen by their sonar sensors. They had problems, however, when seeing a place from different viewpoints and try to overcome this by using a compass to ensure the sensors are always facing compass north.

The method described in this paper attempts to enable reliable place recognition in an arbitrarily unstructured or dynamic environment using features extracted from a 2D laser scan. It is designed to recognise a distinctive place in the presence of substantial noise, occlusion or viewpoint variation provided at least a few of the original node features remain observable. If a node match is made then the relative pose of the vehicle to the node is obtained.

The inspiration for this method comes from attempts to solve the *lost-in-space* problem for star-trackers [10]. Bright stars are used as point features and are grouped into sets of three (triads). The attitude of a star-tracker in space can then be estimated by matching triads extracted from an image of the sky with triads in a map.

For the mobile robot, features are extracted from each laser scan. Nodes in the topological map are described by triads of extracted features (see Figure 1) and any subsequent scan can be matched to the node. Because matching is based only on relative geometry between triads of features, it is robust to both false features and occlusions. Currently, an exhaustive triad matching technique is used which is rather inefficient and can only cope with about 30 features in real time.

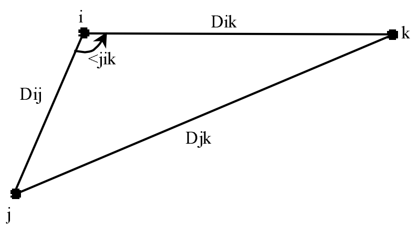

Figure 1: A triad of point features.

Graph theoretic methods may provide a more efficient means to match laser scans than the triad matching method. Converting the matching problem into a Maximal Common Subgraph (MCS) problem [2] is proposed but, at the time of writing, is untested.

This paper is organised as follows. Section 2 describes the vehicles and sensors used to collect experimental data. Section 3 describes the features extraction methods used to generate static features from each laser scan. The next section presents the triad-matching algorithm, which is the core of this paper, and section 5 proposes an outline for using graph isomorphism as an alternative matching scheme. Section 6 shows the results of the experimental data and section 7 gives some plans for future research to improve this method of distinctive place recognition. The final section makes concluding remarks on the reliability of matching and place recognition as demonstrated by the experimental data.

2 Test Vehicles and Sensors

The data used in this paper was logged from an indoor scanning laser mounted on a three-wheeled mobile robot - SydNav. SydNav, shown in Figure 2, is driven and steered from the front wheel only. The two rear wheels are fixed facing forwards and can rotate freely. The front wheel has very accurate encoder data for steering (96000 counts per revolution) and drive (106 667 counts per revolution). There are, however, quite substantial biases in the dead reckoning estimate due to fairly coarse approximations of the centre location of the three wheels, the front wheel radius and the vehicle wheelbase.

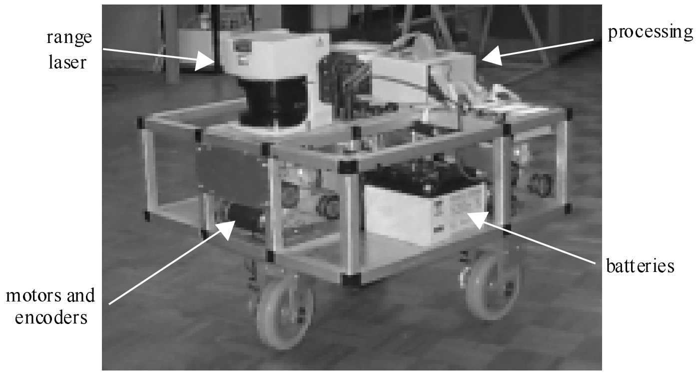

Figure 2: The SydNav mobile robot.

The scanning laser produces a 2D scan over 180° with a resolution of 0.5°. It has a maximum range of about 30m. A complete scan from the laser can be obtained at a rate of 2Hz down a serial link at 19200 baud.

Some additional data was logged in an outdoor environment using an outdoor scanning laser mounted on a quayside cargo-handling vehicle called a straddle-carrier. The reduced resolution and longer range of this sensor made feature extraction rather more difficult than for the indoor laser and so testing with this data remains incomplete at this time.

3 Feature Extraction

Feature extraction is performed on each individual laser scan. The assumption here is that within a single scan the relative geometry between features is fixed (i.e., the scan is an instantaneous snapshot of the environment). This is reasonable as the scan sweep takes about 40ms and the indoor robot moves at less than 0.5m/s.

Presently, the feature extraction algorithms are very basic but they appear to be sufficient to produce reasonably stable features from indoor laser data. The first step in feature extraction is to cluster the range data. This is performed sequentially from the first range measurement. Each range measurement, R_i, is compared with the next adjacent range measurement, R_{i+1}, as shown below:

$$\Delta R_{max} = C1 + C2 * \min\{R_i, R_{i+1}\}$$
$$\text{if } \Delta R < \Delta R_{max} \rightarrow \text{add to cluster}$$
$$\text{else} \rightarrow \text{start new cluster}$$

The constants C1 and C2 are tunable to the laser's noise and resolution characteristics. They were set to 0.07m and 0.04m/m respectively for the experimentation in this paper.

After clustering, there are three types of features that are extracted: foreground edges, foreground points and lines. Foreground edges and points are obtained by checking the end points on the two edges of each cluster. These are compared with the edge points of the two adjacent clusters. If the cluster is reasonably large and either edge point has a shorter range than the edge on the adjacent cluster, then that point is classified as an edge. If, however, the cluster is fairly small and both edge points have shorter ranges than the appropriate edges of the adjacent clusters, then that cluster is a point (see Figure 3).

Lines were extracted using a typical least-squares based line-fitting algorithm such as in [5]. Line features are not currently being used in the matching process. This serves to make matching less reliable as there are many stable line features within the indoor environment and they are generally more reliable than point features. It is hoped to incorporate line features into the matching algorithm in the future.

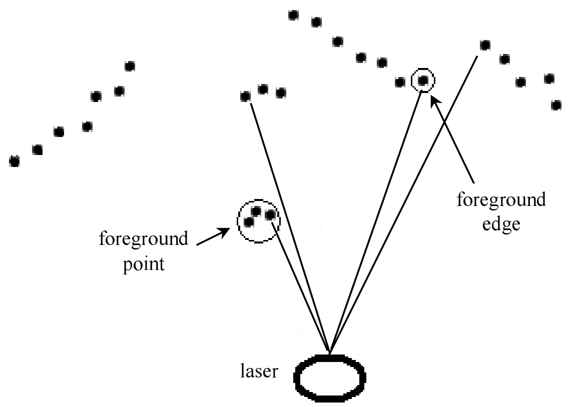

Figure 3: Edge and point extraction from a laser scan.

4 Triad Matching Method

The triad matching method described in this paper is based on an idea used to determine the attitude of a star-tracker from an image of the starscape. Quine [10] approaches this problem by extracting brightest stars from a star map. For each of these stars, the two closest neighbouring stars of reasonable brightness are used to form a triad. In this way, a list of star triads (and their relative geometry) is created. Any subsequent image of the sky will have valid triads that can be matched against the map triad list and, hence, provide the attitude of the star-tracker in space.

The star-tracker method cannot be used directly as a distinctive place recognition algorithm for several reasons. It assumes perception of many (~2000) accurate static point features with very few occlusions or false features (e.g., planets, satellites). With these assumptions, it only generates a single triad for each brightest-star feature. There is no cross-triangulation.

Features extracted from the 2D-laser scan are subject to substantial noise, occlusion and false-features. Also, the number of features extracted is typically quite low (~20). If the features were formed into isolated triads then the probability of subsequent matches would be small as, for each false feature, two possibly good features would be lost. To ensure optimal matching, the maximum number of feature triad combinations must be formed. That is, there would be a list of C_3^N triad combinations formed from N features.

The method presently used to match two scans involves an exhaustive search to match each triad from one scan to each triad from the other. This is a very inefficient algorithm but it guarantees to find a match if at least three matching features exist. It was found to be able to perform real-time matching with around 30 features using a C program on a fast Pentium II PC. The algorithm is as follows:

> First generate the topological map nodes from data logged in the environment in question. These nodes are described by a triad list generated from the scan taken from the node pose. Scans from subsequent runs through the environment are matched to the map scans to obtain the relative pose of the vehicle with respect to the node pose.
>
> Triad list generation:
>
> 1. Generate a list of distances, Dij, from each feature, i, to every other feature, j, in the scan.
> 2. Create a list of triads consisting of all the combinations of three features i, j, k and their associated distances Dij, Dik, Djk.
> 3. For each triad, sort the feature indices i, j, k so that Dij, Dik, Djk are in ascending order.
> 4. Calculate the angle <jik (opposite angle to longest side Djk).
> 5. Sort the triad list according to the minimum distance Dij.
>
> Triad matching:
>
> 1. Create a triad list for the new scan.
> 2. Create an m by n matrix of zeros, M, where m is the number features in the new scan and n is the number of features in the map scan.
> 3. Check sequentially through the triad list of the new scan. For each triad, select the subset of map scan triads that have a valid minimum distance match.
> 4. Select those triads from the map scan subset whose angle and remaining two distances match. (The angle indicates that triads with matching sides are not mirror image matches.)
> 5. For each triad match there is a mapping: new scan {i, j, k} to map scan {m, n, o}. Increment the matrix M at indices (i, m), (j, n) and (k, o).
>
> Relative pose calculation:
>
> 1. From the matrix M, obtain the three strongest mappings (highest count of matches).
> 2. Calculate the relative change in pose required to map the three new scan features onto the three map scan features.
> 3. Transform the new scan features so that the three points lie on their appropriate map scan points.
> 4. Perform a Nearest-Neighbour search for all the new scan features in relation to the map scan features. A feature match exists if a new scan feature has a map scan nearest-neighbour closer than D_{max}.
> 5. If the change in pose found in step 2 reasonably matches the predicted pose of the vehicle relative to the map pose, and the number of nearest-neighbour matches is greater than N (where N>=3) then we have a place recognition.

5 Maximal Common Subgraph

A promising but, as yet, untested alternative to the brute-force triad matching approach is a graph theoretic method to find the Maximal Common Subgraph (MCS) [2]. The idea is that each feature from a scan forms a graph node and the distances or angles between features are the graph edges:

Nodes: Points and lines.
Edges: Point-point → distance
Point-line → perpendicular distance
Line-line → acute angle

A fully connected graph is generated for both the map scan and the new scan and, by solving the MCS between these two graphs, the maximum subgraph that is isomorphic to both graphs is obtained. It is hoped that this subgraph will directly indicate the mapping of features that are geometrically consistent between the two scans.

6 Results

Experimental data was logged by SydNav in an indoor corridor environment (see Figure 4). An accurate global pose of the vehicle was calculated off-line using a localisation and map building algorithm described in [3]. This accurate pose information is for display purposes only and was not used in the matching or node recognition process.

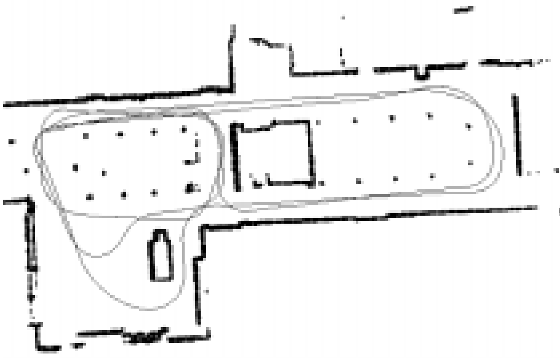

Figure 4: Laser generated map of environment and the path traveled by the robot.

Six distinctive places were chosen at arbitrary locations from the logged data as shown by the triangles indicating robot pose in Figure 5. These places are described by the single scan taken at that pose. Matching was then performed sequentially for each scan from the experimental run against each of the map scans.

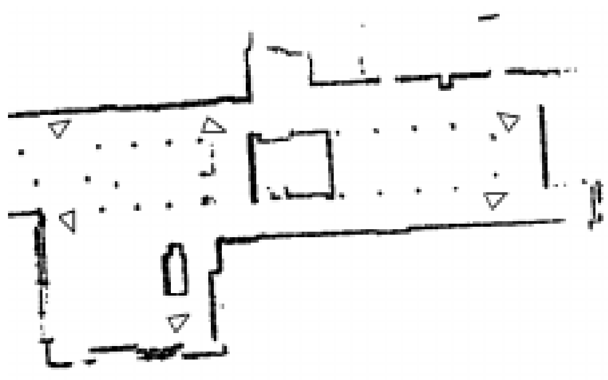

Figure 5: Six distinctive places – the point of the triangle represents the front-centre of the vehicle.

A photo of the environment as seen from the upper-right-hand node is shown in Figure 6. The pose of the vehicle at this distinctive place and its defining laser scan are shown in Figure 7 (note the railing poles, the two doorways and the support column in both pictures). The positions from which the upper-right-hand node was visible and correctly identified are indicated by the dark dots and the positions from which it was falsely identified are indicated by the light dots. At most of the locations where false matches were made, the distinctive place was not even visible. Usually these false matches were based on matches between three or four features only, with the even spacing of the railing poles as the main offenders.

Figure 6: Corridor environment in the region of the upper-right-hand distinctive place.

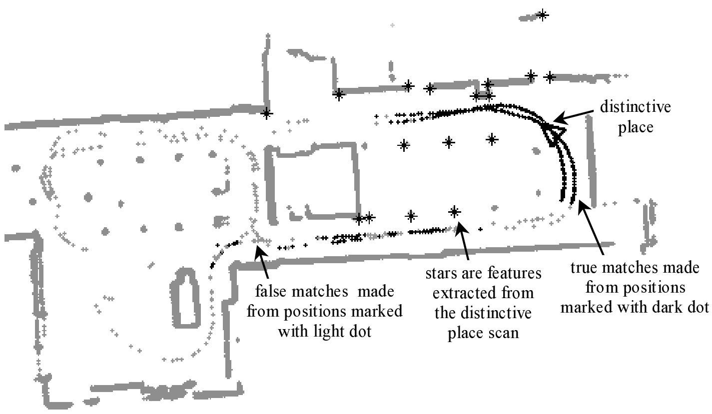

Figure 7: Definition of upper-right-hand node and positions from which matches to this place were made.

Figure 8 shows an example of a correct match to the upper-right-hand distinctive place based on five matching features. The node pose and the new pose of the vehicle are shown in their true global positions, as are their respective scan features. The agreement between the relative pose generated by the matching process and the true relative pose can be seen by the way the scan features lie in close vicinity to their appropriate mapped feature. The match in Figure 9, however, is a false match based on six matching features. The features in the new scan were mapped to the node features 1.8m to the left and this resulted in a 1.8m discrepancy between the relative pose generated by the matching process and the true relative pose.

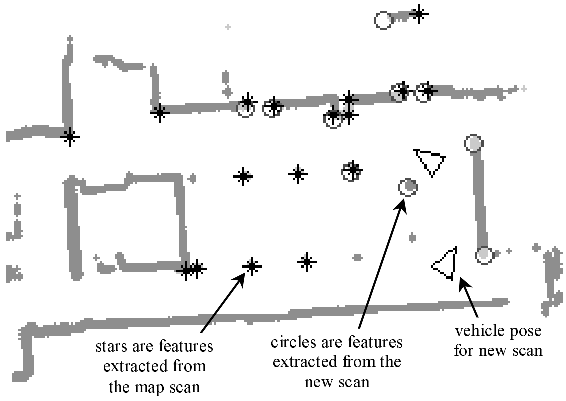

Figure 8: Correct match to the distinctive place.

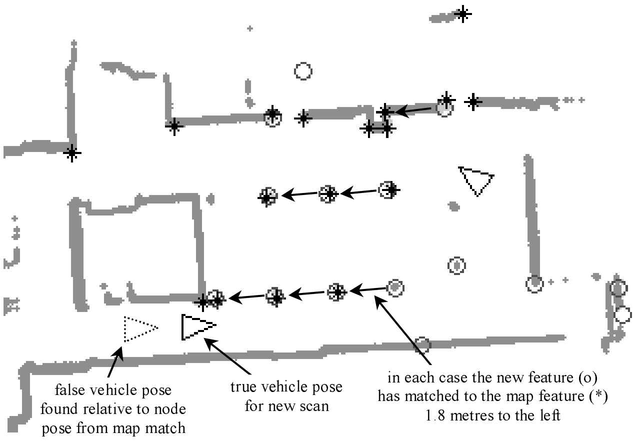

Figure 9: False match to the distinctive place.

The magnitude of the discrepancy between the map matched relative pose and the predicted relative pose is the basis for determining whether to accept a match as true or to reject it. In the experiment described in this paper, the predicted relative pose was based on raw encoder-based odometry. True matches were found by gating the distance discrepancy at 0.4m and the rotational discrepancy at 0.1 radians. These values allow correct matches even in the presence of significant accumulated error in the encoder data (although, in practice, the error accumulation between nodes would be small if distances between nodes were small).

From the entire run there were 2884 matches made to the six distinctive places. The consistency of the matching procedure can be seen by the large concentration of distance errors within the 0.4m gate (particularly at 0.05m) in Figure 10 compared to the reasonably flat distribution outside the 0.4m gate in Figure 11.

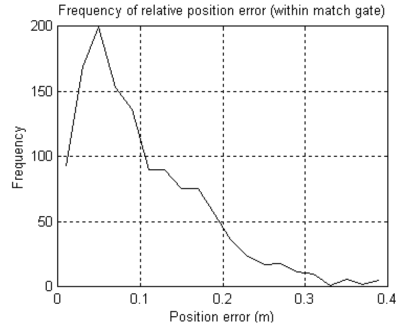

Figure 10: Frequency of distance discrepancies less than the distance discrepancy gate.

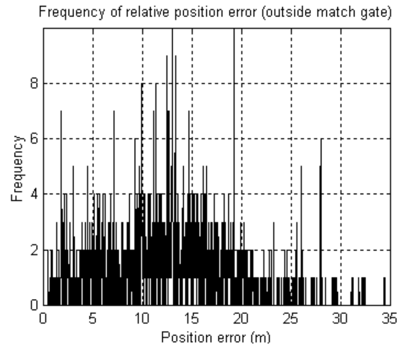

Figure 11: Frequency of distance discrepancies greater than the distance discrepancy gate.

False matching decreases rapidly with the number of features matched in a scan (see Figure 12). No false matches were found if more than six features were matched between the two scans. It is hoped that mismatching will be reduced even further with the addition of line information.

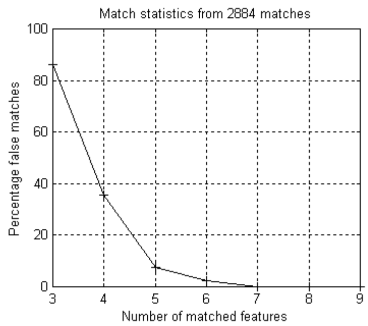

Figure 12: Percentage of false matches versus the number of features matched.

The use of encoders as the comparison for the map matched relative pose is only one method for gating out false matches. An alternative method would be to define many distinctive places in fairly close proximity such that multiple nodes could be identified in a single scan. In this way matches could be used to validate each other – requiring no encoder data at all.

7 Future Work

The following avenues are to be examined in relation to this work. Feature extraction must be improved to obtain stable features from the outdoor laser data (and better stability for the indoor laser data). A stochastic feature covariance measure would also be of value.

The graph matching process requires a substantial speed-up to make it computationally tractable with larger numbers of features. It is hoped that the MCS method will be the answer here. The MCS method would also allow straightforward integration of line features into the matching process.

The stability of features over several scans can be checked by performing matching between sequential scans. This can be used to give a weighting measure for the number of times a feature has been sighted and successfully matched.

Finally, it is hoped to develop robust automatic distinctive place generation and dynamic maintenance of these node scans. Distinctive places should be defined at intervals when scan data of high confidence is available. The confidence measure would be a combination of the number of features in the scan, the weighting measure for each feature (based on sequential scanning) and the confidence associated with the geometric configuration of the features (similar to the position estimation precision available for different satellite configurations with GPS [12]). Dynamic node maintenance would involve tentative additions and subtractions to the node features whenever the place is revisited. These additions and subtractions would cater for the existence of temporary static objects in the environment.

8 Conclusion

Although the experimental data was obtained in a static environment, its usage in the place recognition algorithm was non-ideal. As the aim of this paper was to demonstrate the robustness of the matching process, no effort was made to improve the following:

- By using just point features, a large portion of the available information (i.e., line features) was unused in the matching process.
- The distinctive place nodes were chosen arbitrarily rather than at locations where there was a lot of stable feature information.
- The nodes were defined by a single isolated scan without any checking made for the stability of the features extracted from it.

The results from place recognition showed that most matches were either well within the distance and rotation validation gates or had large (easily detectable) discrepancies. The majority of the false matches were matches based on just three features. Dramatic improvement was obtained by only accepting matches involving five or more features.

The distinctive place recognition method described in this paper appears to be reliable and its robustness can only improve with additional line information and intelligent map generation.

References

[1] Aycard O., Charpillet F., Fohr D., Mari J. F., "Place Learning and Recognition using Hidden Markov Models", *Proceedings of the 1997 IEEE International Conference on Intelligent Robots and Systems*, pp 1741-1746, 1997.

[2] Chen C. K., Yun D. Y. Y., "Unifying Graph Matching Problems with a Practical Solution", *International Conference on Systems, Signals, Control, Computers*, Sept. 1998.

[3] Dissanayake M. W. M. G., Newman P., Durrant-Whyte H. F., Clarke S., Csorba M., "An Experimental and Theoretical Investigation into Simultaneous Localisation and Map Building", *6th International Symposium on Experimental Robotics*, pp. 171-180, March 1999.

[4] Durrant-Whyte H. F., "An Autonomous Guided Vehicle for Cargo Handling Applications", *International Journal of Robotics Research*, pp. 407-441, 1996.

[5] Kahn P., Kitchen L., Riseman E. M., "A Fast Line Finder For Vision-Guided Robot Navigation", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 12, no. 11, pp. 1098-1102, November 1990.

[6] Kortenkamp D., Weymouth T., "Topological Mapping for Mobile Robots using a Combination of Sonar and Vision Sensing", *Proceedings of the 12th National Conference on Artificial Intelligence*, pp 979-984, July 1994.

[7] Kuipers B. J. and Byun Y. T., "A Robot Exploration and Mapping Strategy Based on a Semantic Hierarchy of Spatial Representations", *Journal of Robotics and Autonomous Systems*, pp. 47-63, 1991.

[8] Madhavan R., Dissanayake M. W. M. G., Durrant-Whyte H. F., "Autonomous Underground Navigation of an LHD using a Combined ICP-EKF Approach", *Proceedings of the 1998 IEEE International Conference on Robotics and Automation*, vol. 4, May 1998.

[9] Owen C., Nehmzow U., "Landmark-Based Navigation for a Mobile Robot", *Proceedings Simulation of Adaptive Behaviour*, MIT Press, 1998.

[10] Quine B. M. and Durrant-Whyte H. F., "Rapid Star Pattern Identification", *Proceedings SPIE Acquisition, Tracking, and Pointing X*, Vol.2739, pp. 351-360, Orlando, U.S.A, 1996.

[11] Schultz A. C., Adams W., "Continuous Localisation Using Evidence Grids", *Proceedings of the 1998 IEEE International Conference on Robotics and Automation*, vol. 4, May 1998.

[12] Spilker J. J., "Satellite Constellation and Geometric Dilution of Precision", *Progress in Astronautics and Aeronautics*, vol. 163, pp. 177-208, 1996.

[13] Thrun S., "Learning Metric-Topological Maps for Indoor Mobile Robot Navigation", *Artificial Intelligence*, vol. 99, no. 1, pp 21-71, February 1998.