Localisation In Large-Scale Environments

Tim Bailey and Eduardo Nebot

Australian Centre for Field Robotics
Department of Mechanical and Mechatronic Engineering
The University of Sydney, NSW 2006, Australia

Abstract

This paper describes a localisation framework that combines the accuracy of feature maps with the scalability of topological maps. The map is structured as a graph of nodes where each node defines a local region feature map. This breaks the localisation process into a combination of regional feature tracking and node-to-node context switching. As part of the practical implementation of the localisation system, we introduce a batch data association method that uses the simultaneous observation of multiple features to determine data associations in a manner decoupled from the vehicle pose estimate. We also present an observation-based dead reckoning procedure that estimates vehicle motion in place of odometry and does not require a kinematic vehicle model. Experimental results demonstrate that this approach is capable of localising in large-scale outdoor environments. We perform tests in an inner-city park and a suburban street using a scanning range laser as the sole information source. The diverse nature of these two environments indicates that these techniques have broad application.

Key words: Outdoor navigation, Autonomous systems, Data association, Topological feature maps, Odometry-free dead reckoning

1 Introduction

A mobile robot's ability to determine its location within an environment is a basic competency for autonomous navigation. Genuine autonomy further dictates that the robot be able to localise from its natural surroundings using

Email addresses: tbailey@acfr.usyd.edu.au (Tim Bailey), nebot@acfr.usyd.edu.au (Eduardo Nebot).

onboard sensors without requiring any artificial modification of the environment. Some major issues today regarding localisation from natural surroundings concern generality (different types of environment), scalability (application to large environments) and the ability to enhance and extend a map of an environment while localising from it [16,21,34]. Therefore, we desire a paradigm that will operate successfully in any environment of arbitrary size and uses an extendible map structure.

This paper presents several localisation experiments using a scanning range laser sensor. A particular characteristic of the laser, the ability to obtain batches of near simultaneous observations with high precision, is exploited to implement novel data association and dead reckoning algorithms. These algorithms are useful in their own right but are used here to demonstrate a topological feature map structure which enables localisation in large general environments. The main purpose of this map structure is to facilitate implementation of large-scale Simultaneous Localisation and Mapping (SLAM) and particularly to address the revisitation problem for large cyclic environments. Although the topological feature map is only presented in terms of localisation in this paper, its application to SLAM is the subject of current research.

The localisation approaches most commonly used today include evidence grids, feature maps, and topological maps. Each have their particular advantages and each, to date, have important issues concerning their ability to operate in certain environments and to scale appropriately to large, dynamic applications.

Evidence grids [12,29] represent a region as a matrix of cells. Each cell describes a small rectangular area in the environment by a number which indicates the probability that the area is occupied. A robot's position is maintained by accumulating a short-term perception map of the local environment and periodically registering it with the global map. The short-term map fuses sensor data into a uniform occupancy representation and is a good means for filtering ambiguous sensor data like sonar. The registration process is used to offset accumulated pose errors.

Evidence grid maps have the advantage of being able to easily fuse data from different sensors and to explicitly model both obstacles and free-space. They present some problems, however, concerning the granularity, scalability and extensibility of the map. These problems are mainly due to a map's fixed grid size which defines the limit of localisation accuracy and incurs memory and computation overhead proportional to the number of grid cells. One method designed to address this issue uses quadtrees or octrees to provide variable-sized cells [24,8] thus concentrating memory usage to occupied areas and enabling variable granularity. Nevertheless, evidence grids have not yet been implemented in large outdoor environments or in places where the maximum map area cannot be specified *a priori.*

Feature maps [20,32] consist of the global locations of a set of features and, sometimes, their geometric properties. Localisation is then formulated as a multiple target tracking problem. The robot pose (i.e., 2D position and heading) is usually defined in state-space by a state vector $[x, y, \phi]^T$ and covariance matrix with the pose estimate maintained by a recursive filter, usually an extended Kalman filter (EKF). As the robot moves, its state is predicted using dead reckoning, typically odometry with a kinematic vehicle model, and the covariance of the state estimate expands. Periodically, externally sensed information of the local environment is obtained and features are extracted. If a sensed feature is uniquely associated to a map feature, the relative location of the feature to the vehicle can be used to improve the state estimate and decrease its covariance. The state-space approach is efficient and produces an optimal global pose estimate but tends to be brittle in the presence of modelling and data association errors.

Modelling error problems are largely related to the implicit assumption of the EKF that estimate errors have a zero-mean Gaussian distribution. In reality, however, estimate errors can be due largely to systematic errors in the state and observation models, non-model dynamics and non-linearities. These problems may introduce significant biases into the filter and lead to either sub-optimal or even inconsistent filter operation. For example, inaccuracies in the robot's odometric motion model (e.g., in the wheel-base or wheel radius estimate) will result in a biased state prediction. Furthermore, an odometric model using wheel encoders cannot observe wheel slip and this injects another bias into the state prediction. These errors are usually compensated for by over-conservative tuning of the EKF so that the covariance estimate is large enough to encompass both Gaussian and systematic errors. The drawback of such tuning is sub-optimal filter performance and an increased likelihood of data association failure.[1] An alternative to odometry-based dead reckoning is given in this paper which requires no vehicle model and is largely free of systematic errors making it ideal for a Kalman filter prediction phase.

Data association within the Kalman filter framework is usually performed as follows. Features are extracted from a set of sensed data. The locations of map features relative to the current robot pose are predicted and compared with the sensed features. If a one-to-one mapping exists between the sensed and map features, the information is used to update the robot pose estimate. If, however, ambiguous mappings exist, these observations must either be rejected or applied using a more sophisticated association technique (such as multi-hypothesis [26]). The gating threshold to validate a particular association is a

[1] Recent work using Monte Carlo methods [10] may allay some of these difficulties. Monte Carlo filters represent uncertainty distributions via discrete samples and can more accurately describe non-Gaussian distributions. They can therefore implicitly represent multi-modal distributions incurred by ambiguous data-associations.

function of both the observation and the vehicle pose uncertainties, and the dependence on vehicle pose uncertainty is the key reason for data association failure. If the Kalman filter is tuned too tightly and becomes inconsistent, the threshold may be too small to allow any matches. This may lead to prolonged periods of dead reckoning and, with a biased prediction model, may result in false matches. Alternatively, if the filter is tuned too conservatively, multiple feature associations are likely and false matches could occur. A false feature match often results in the divergence of the localisation algorithm. This paper presents a more robust alternative data association procedure [5] which uses the relative spatial coordinates of features to decouple feature matching from the robot pose estimate.

A final issue regarding state-space methods concerns the extension to map-building while localising. This concept was originally introduced in [31] wherein every observed feature is added to the state vector. This means the state vector increases linearly with the number of features and the covariance matrix increases to the square of the number of features. Thus it becomes necessary to derive a consistent map-management method to make SLAM scaleable to arbitrary sized environments. Current attempts to address these problems include using multiple maps [21] and exploiting the special characteristics of the matrices involved in SLAM [15], but there still remain problems with very large environments.

Topological maps are a graph-based description of the environment where each node is a Distinctive Place (DP) [19], and the connecting edges between nodes contain procedural information that will enable the vehicle to travel from one node to the next. A DP is a location in the environment that is distinguishable from other places on the basis of patterns observable in sensory data. These patterns are usually model-based representations of common indoor structures such as doorways and corridor intersections [2] although some work has been done to identify places by images seen from that location [36]. Edges connecting DPs are typically either the set of behaviours required to travel between them, such as wall following, or local metric information of the relative pose of one DP to the other [13,30].

A major advantage of topological maps is that they divide the world up into a connected set of regions such that the robot is required only to successfully navigate from one region to the next. Thus, topological maps provide an inherently scalable localisation structure which relies merely on robust transition between local regions. The primary weakness of topological map representations has been reliable DP recognition. Most work to date has been in indoor environments and assumes the presence of common indoor structures but, even in these areas, recognition can be sensitive to occlusions and viewpoint variations. Failing to recognise a DP or mistaking the wrong part of the environment as a DP can result in localisation failure. Another limitation of

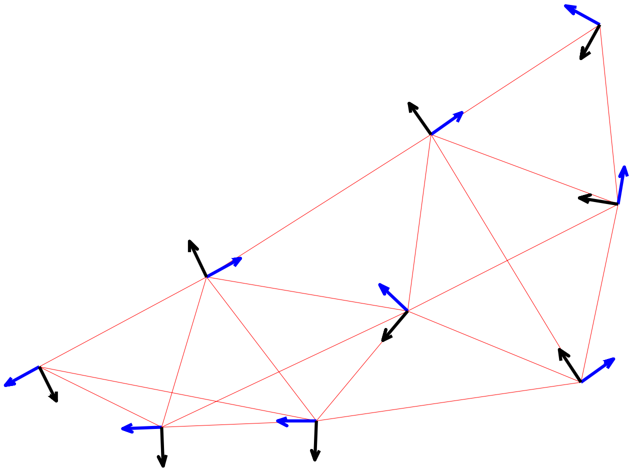


Fig. 1. Topological feature map structure. The map is defined as a graph data structure where each node is a local feature map and each edge represents the relative pose between two coordinate frames.

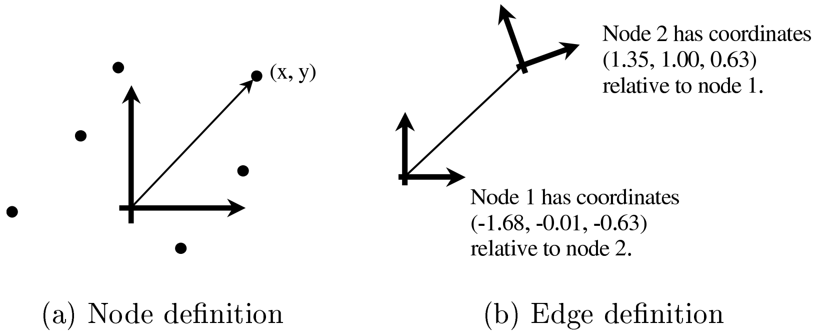
current topological implementations is that the robot can usually only recognise a DP once it has reached the DP location, so that the robot can only offset accumulated dead reckoning errors upon arrival, and is unable to track its pose relative to the DP.

1.1 A Topologically Structured Feature Map

This paper proposes a topologically structured feature map that combines the optimal tracking capabilities of feature maps with the scalability of a topological map. The definition of this map structure is as follows:

- The map consists of a topological graph of nodes and edges as shown in Figure 1. Each node defines a unique physical location $[x, y, \phi]^T$ in the environment as the origin of a local coordinate frame $[0, 0, 0]^T$.
- Each node is represented by a set of nearby features. The locations of these features are stored in terms of the local coordinate system as shown in Figure 2(a). We define the area surrounding the node containing representative features as the *node region*. The extent and shape of the node region is arbitrary.
- The connecting edges between nodes specify the pose of the coordinate origin of one node in terms of the local coordinate system of the other (see Figure 2(b)).

A robot's location is obtained in the coordinate system of the node nearest and most visible to it. More precisely, in our implementation, it is in the coordinate system of either the nearest node or the second nearest depending on



(a) Node definition (b) Edge definition

Fig. 2. Topological feature map definition. The nodes (a) are defined by surrounding features and the graph edges (b) specify adjacent nodes in terms of their relative pose.

which offers the greater number of features in the robot's sensory field-of-view. The robot pose is determined relative to the subset of node features that are observed by its sensors. When a neighbouring node in the map becomes more visible than the current node (i.e., becomes one of the two closest nodes and has a greater number of features-in-view) the robot simply switches coordinate systems.

In the region of a node's feature set a robot can track its pose relative to the node with bounded growth in its estimate error. If node regions overlap, then it is possible to maintain tracking continuously but, if the spacing between nodes is greater than the node regions, there will be periods where the robot must rely on dead reckoning alone, during which error growth is unbounded. Entering a node region (and returning to tracking state) after prolonged dead reckoning requires data association that is capable of coping with substantial vehicle pose covariance. Some methods for pose decoupled data association are introduced later in this paper.

As an *a priori* map, the topological feature map offers no real advantage over ordinary feature maps. However, this structure possesses the ability to be efficiently extended making it a feasible map representation for large-scale SLAM where normal feature maps would become computationally intractable. The map may be augmented in two ways. First, the state of the features surrounding a particular node can be updated using standard SLAM algorithms leading to a locally optimal map within each node region. Second, the map graph can be extended with new nodes to provide a set of loosely coupled node regions which define a large-scale map without loss of consistency. Applying SLAM to this framework is proposed as a future extension to this work [3].

Traditional data association methods suffer from their reliance on knowledge of the vehicle pose estimate. If the quality of the pose estimate deteriorates, these methods can become quite brittle, particularly in cluttered environments. For sensors capable of obtaining *batch observations*, however, it is possible to constrain data association in a manner decoupled from the pose estimate. Batch observations are a set of observations (i.e., viewed features) sensed simultaneously or within such a short time-span that motion compensation can offset any geometric distortion. These observations enable data association to be performed on the basis of the relative geometry between feature locations (e.g., see [25] for a good example).

Additional constraint on data association can be applied using techniques, borrowed from the topological map community, for improving distinctive place recognition. Topological place recognition methods often attempt to match sensed data to a database of place definitions via the correspondence of data signatures (e.g., [33,1,36]). The likelihood of a unique matching is determined by the uniqueness of the signature and this can be increased by, for example, using disparate image characteristics such as dominant edge orientation, edge density, gradients, and texturedness [1]. Similarly, place distinctiveness can be improved by combining multiple sensing modalities such as sonar and vision [18]. We have taken these methods on board, at least in principle, by defining our features, not simply in terms of location, but including information regarding the feature type (e.g., points, lines, objects) and characteristics (e.g., size, shape). The current implementation applies this notion in a limited sense by defining two feature types, points and edges, and one feature characteristic, point radius. Therefore, by increasing the individuality of the features in a batch observation, we are more likely to obtain a correct set of data associations.

In this paper we present three data association algorithms. The first algorithm is particularly useful as it generates a maximum combinatorial likelihood of associations with no prior estimate of the relative pose between the two sets of data. This allows the relative pose between the data sets to be determined without tracking. A correct mapping between the data sets is guaranteed provided the set of features common to both data sets outnumbers the set of features that generate some coincidental false mapping through geometric symmetry. The likelihood of false mappings through coincidental symmetry decreases greatly with the number of associations found [4] but there can be no direct guarantee that the association set is correct.

Data association with further constraint imposed by tracking the relative pose between data sets (i.e., having prior relative pose information) is considerably more reliable. The two subsequent algorithms in this paper generate a set of associations given a prior relative pose distribution between the two data sets. Therefore, of the three algorithms, the first (non-tracking) algorithm can provide an initial guess of the relative pose and one of the tracking algorithms can confirm and track this estimate.

1.3 Odometry Free Dead Reckoning

There exist applications, particularly in all-terrain outdoor environments, where odometry-based dead reckoning is extremely unreliable. While encoder data may not be particularly noisy, systematic errors (i.e., from faults in the vehicle model) and bias errors due to slip can be substantial enough to promote filter divergence.

Tracking the relative pose of sequential batch observations provides an alternative dead reckoning source that, while perhaps more noisy than odometry, is largely free of systematic error. Sequential tracking does not require a vehicle model and will operate successfully while ever there are observable features in the sensor's field of view. This paper presents experimental results using laser which show remarkably low drift rates in two different outdoor environments over distances of several kilometres. This is shown to be a significant improvement when compared to the dead reckoning information of standard encoder-based odometry.

1.4 Format of This Paper

The paper is organised as follows. The next section describes the experimental equipment used and the environments in which these methods were tested. Section 3 details feature extraction from raw sensed data. Section 4 describes the data association algorithms and section 5 presents dead reckoning information obtained by mapping sequential laser scans. Section 6 defines the manual construction of the topological feature maps and gives the experimental localisation results. Section 7 discusses some future directions of this research and the final section makes concluding remarks regarding the experimental results.

2 Experimental Arrangement

Sensor data from a 2D scanning range laser (SICK PLS) was used for experimental demonstration of the algorithms described in this paper. The laser

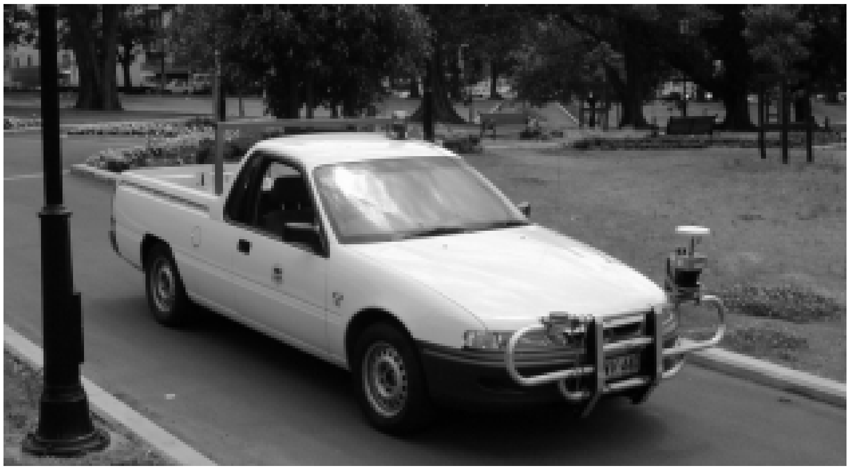


Fig. 3. Utility vehicle with laser sensor in park environment.

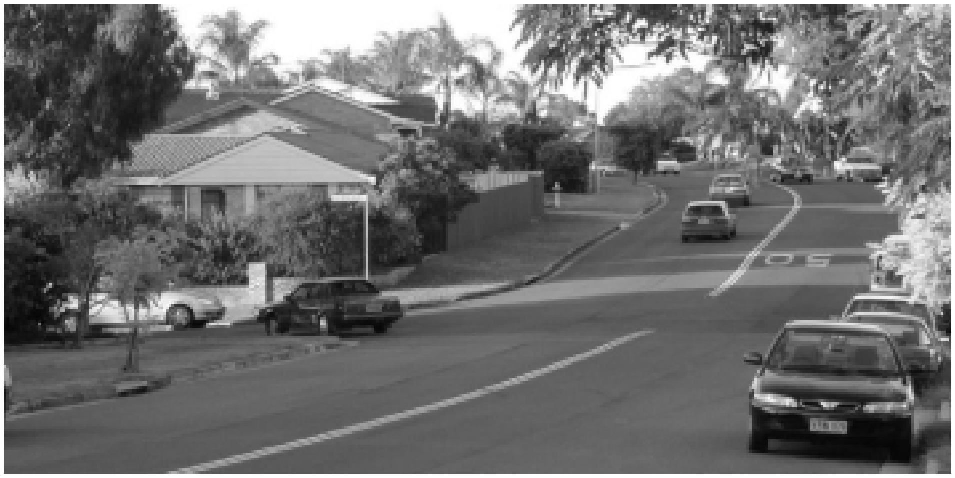


Fig. 4. Suburban street environment.

returns a 180° planar sweep of range measurements in 0.5° intervals (i.e., 361 range values in anti-clockwise order) with a range resolution of about $\pm 5cm$. Data was logged by attaching the laser to the front bumper of a utility vehicle (see Figure 3) and driving it through an outdoor environment. The vehicle was equipped with encoders for measuring wheel rotation and steering angle and a differential GPS unit capable of sub-metre accuracy. This information was not used by our localisation algorithms but provided the ground truth for map construction and results validation.

This paper describes the results from two outdoor environments. The first was an inner city park shown in Figure 3 with a test run of 3200 metres and the second a suburban street shown in Figure 4 with a test run of 4350 metres.

The park environment had a large number of trees that provided good point features although there were portions of open-space where the laser was unable to detect features of any kind. During the test run, dynamic obstacles were present in the form of people walking through the field-of-view and cars driving along the major road adjacent to the park. An additional source of false features and occlusion was due to ground undulation where occasionally sweeps of ground would dominate a scan.

The street environment involved driving through a suburban area in the presence of traffic. This route was characterised by fences, shrubs, sign posts and parked cars. It was a difficult area to extract good quality point features and there were a lot of noisy features. The terrain was quite hilly and this also decreased feature stability. Thus the street environment presented a more challenging test in terms of feature extraction and data association.

3 Feature Extraction

Feature extraction is the process of segmenting raw sensed data and classifying these portions according to their applicability to a set of representation models. Using models has several advantages over simply using raw sensor data directly (e.g., scan matching [22]):

- The information-space for describing the view is reduced for more efficient computation.
- It is possible to select small, discrete, foreground objects that are highly visible and are unlikely to all be affected by occlusion at the same time.
- The sensed data that does not fit a model is rejected. This ensures that, to some degree, the information used is only that which is likely to be reliable. For example, the laser sometimes scans arcs of ground due to ground undulations. This may confuse matching algorithms based on raw data but, as this data does not fit one of our feature models, the feature-based approach simply rejects it.
- The state of a feature described by a parametric model may be iteratively improved via Kalman filter.

One disadvantage, however, is that all useful information which cannot be classified to a predefined model is lost along with the bad. Also, it is possible that a set of data may be classified by a feature model that does not represent the raw information well, resulting in an unreliable feature (e.g., a set of data defining a smooth curve will not be well represented if classified as a set of line segments). Prior to classification, a raw laser scan is segmented into clusters. This is done by splitting the data at discontinuities in range using the algorithm given below. In our experiments, the constants for determining ΔR_{max} were $C = 0.07m$ and $P = 0.04$.

```
$\Delta R = abs(R_i - R_{i-1})$
$\Delta R_{max} = C + P * min\{R_i, R_{i-1}\}$
if $\Delta R < \Delta R_{max}$
   add $R_i$ to cluster
else
   $R_i$ starts new cluster
```

The feature types used in this paper are foreground points and edges as shown in Figure 5. A cluster is classified as a point if it is in foreground (i.e. has shorter range than clusters adjacent to it) and is reasonably small (i.e., has a small distribution of points). Point features are modelled as a circle with radius

$$r = D\frac{\sin\theta}{1 - \sin\theta}$$

Laser data often gives spurious range values at the edges of objects but usually the angular measurement presents a more reliable information in such circumstances. Thus a circle is defined by D, the minimum range measurement in the cluster, and 2θ, the subtended angle of the cluster (see Figure 6). A circular point model is not a good representation for non-circular objects but, given that the maximum allowable radius was small, it is quite accurate for even non-ideal object shapes.

A foreground edge is defined when any cluster, that has not been classified as a point, has an edge point that is in front of the edge point of the adjacent cluster.The edge model is generally a quite stable feature although it is unreliable for detecting the edges of smooth curves.

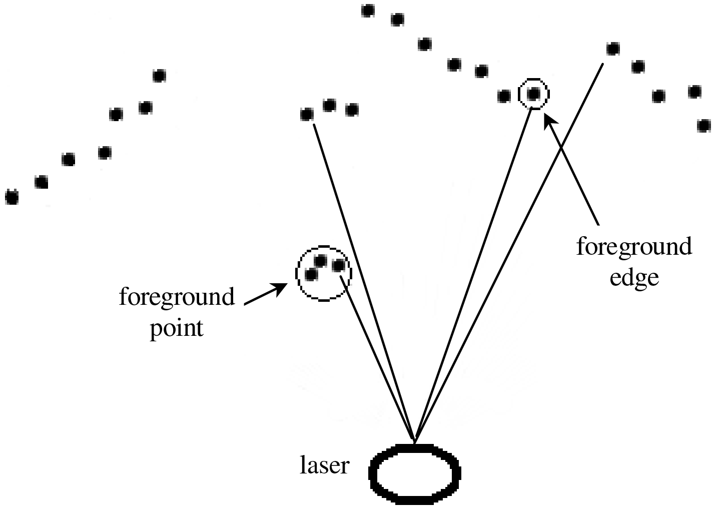


Fig. 5. Feature extraction. Raw laser data is first clustered based on discontinuities between adjacent range measurements and then classified according to point and edge feature models.

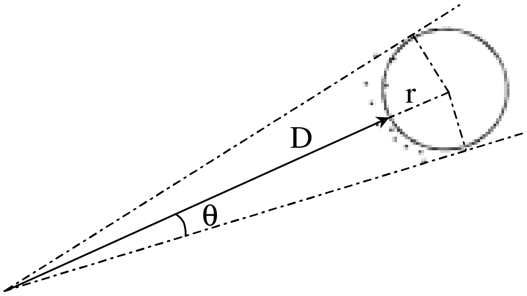


Fig. 6. Point feature circle model. The parameters of the foreground point are specified by the subtended angle 2θ and the minimum range measurement D.

The feature extraction procedures shown here are simplistic but demonstrate the concept of having feature descriptions beyond just point locations. The distinctiveness of a set of features directly affects data association reliability and can be improved by increasing the diversity and precision of feature types and their characteristics. Further distinctiveness may be achieved by using more sophisticated feature models (e.g., Gaussian sum distributions [23]) or by incorporating multiple sensing modalities (e.g., vision and sonar [18]).

4 Data Association

A large number of data-association schemes exist that can be roughly divided into three subgroups. First there are those which use the entire data-set as a feature and attempt to maximise a correlation between two sets of raw data. This approach is usually implemented based on the impression of similarity of two data sets (e.g., principal component matching [9] and image signatures [13]) or on the maximum overlap of individual raw data points (e.g., scan-matching [22]). The latter of these two implementations is more robust to view-point changes, occlusion and dynamic objects. When combined with probabilistic sampling techniques, such as particle filters [14], to overcome local minima, this method could be an attractive alternative to extracted feature techniques.

The second data-association scheme involves extracting features and matching them piece-wise to map features using a knowledge of the vehicle pose. This method has been termed the Nearest-Neighbour approach [11] and is employed in most current SLAM implementations. It suffers from the fact that feature association is dependent on the vehicle pose error and can fail gradually through lack of true associations or suddenly through a false association. A significant issue with SLAM is the re-registration of map features after completing a large loop and revisiting a previous section of the map. At the point of revisitation, the pose covariance is large and nearest neighbour data association is very unreliable.

The third method requires the extraction of a batch of features simultaneously from a set of raw data and performs data association based on the combined association likelihoods between two feature sets. One approach, presented by Uhlmann [35], combines the individual association probabilities in a Joint Assignment Matrix and generates a joint association probability by considering the one-to-one mapping constraint. However, this method does not utilise the information provided by the relative spatial geometry within each feature set. Neira and Tardós [25] present a branch-and-bound algorithm that *does* exploit the relative spatial constraints. This method searches for the maximum set of associations that satisfies a Joint Compatibility measure combining both individual feature assignment likelihoods and their batch relative spatial properties. As a tracking data association algorithm, this approach is optimal but it is not suitable for global localisation (i.e., data association with no *a priori* pose information) since it is still dependent on vehicle pose information.

Sometimes it is useful to obtain a most likely set of associations without a prior vehicle pose estimate. Bailey *et al.* [4] present a method for performing batch data association with no prior pose information that uses the relative geometry between point-feature triplets. A more rigorous method is presented in [5] which generalises the non-tracking data association concept to arbitrary feature types using a graph-based algorithm. This graph-theoretic data association approach has been used for many years in the areas of 2D and 3D object recognition [7,17]. However, to our knowledge, this technique has not previously been applied to target tracking data association or the robot localisation problem.

A rather different paradigm for performing localisation without prior pose information is demonstrated by Dellaert *et al.* [10]. This method is a Monte Carlo algorithm that samples pose values in the robot domain and weights them according to an observation-based likelihood function. The algorithm does not use features but correlates the observation data set (to the map) in its raw form. It is worth making some remarks concerning the Monte Carlo global localisation approach compared to graph-theoretic data association. They both, in a limited sense, permit global localisation but each are restricted in different ways. The Monte Carlo method does not strictly perform global localisation but actually describes a location distribution over the domain covered by samples. The extent of this domain is limited to the area that can be sufficiently covered by the number of samples available. The graph-based data association method, on the other hand, is not limited by the size of the region but by the number of features in a batch. As such it can enable localisation for much greater domain sizes (particularly if a batch of features is processed in sub-batches).

A drawback of explicit feature association, however, is the inability to directly describe a location distribution. For example, a situation that would appear as a multi-modal distribution for the Monte Carlo approach would manifest as multiple alternative sets of data associations for the graph-based method—where each association set defines a different vehicle pose [3]. One way to counter this ambiguity is to incorporate vehicle pose information using a tracking data association algorithm (such as Neira and Tardós [25]). In this way, each ambiguous location can be tracked until confirmed or rejected. Two new tracking association algorithms are presented in this paper. The first is an extension to the graph-based non-tracking method and the second is a Monte Carlo nearest-neighbour algorithm. These algorithms are useful as flexible alternatives to existing tracking batch data association techniques; particularly for applications where non-stochastic state estimation is used (i.e., where covariance and correlation information are not available).

4.1 Non-Tracking Data Association: Maximum Common Subgraph

The main aim of this approach is to decouple data association from the vehicle pose estimate (see Figure 7). As mentioned previously, this requires that the sensor used is capable of observing a set of features either simultaneously, or with sufficiently small temporal displacement that motion compensation can offset distortion of the scene. Essentially, the sensor must be able to provide multiple features with accurate inter-feature geometry.

This algorithm relies on the fact that a set of static features will possess invariant inter-feature relationships (e.g., distance between two points or angle between two lines). It is worth noting that these inter-feature relationships are arbitrary and need not be spatial. They may be any invariant property that will apply some constraint between two features. The feature information is characterised as a graph where the feature types and their various properties are stored as graph nodes, and the graph edges represent the inter-feature relationships. Thus, data association between two feature sets can be expressed as the graph-theoretic problem of finding the Maximum Common Subgraph (MCS) between two graphs. This results in the maximum subset of features where all the inter-feature constraints are mutually satisfied. A complete description of the MCS algorithm for non-tracking data association is provided in [5].

4.2 Tracking Data Association

The basic MCS algorithm is typically advantageous for robot pose initialisation or lost state recovery where there is no prior pose estimate. The data

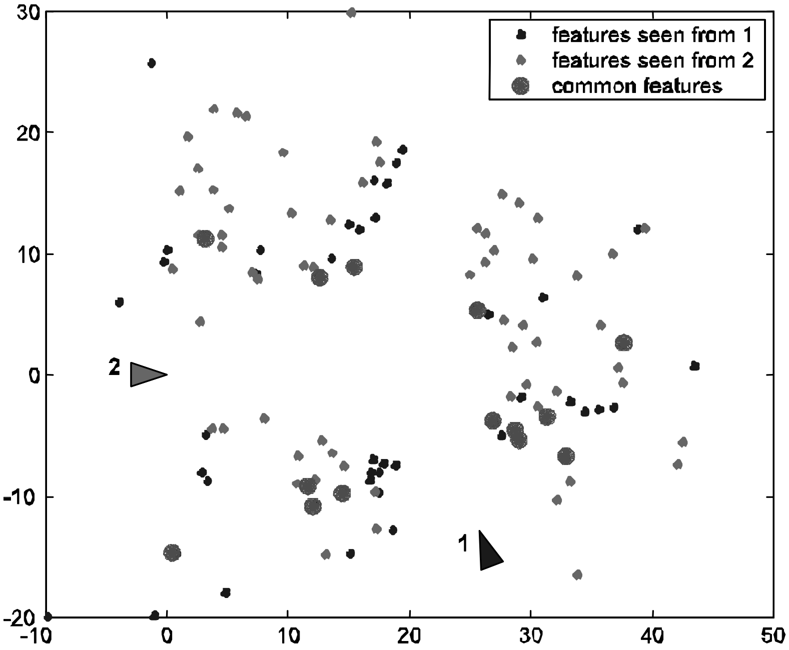


Fig. 7. Non-tracking batch data association. Two feature sets (extracted from two real laser scans) are associated with no prior relative pose information. Both sets contain a large proportion of false features (clutter) but the common features are associated successfully. The relative pose between the two observer locations is then obtained as shown.

association set it returns, however, may be a false mapping between feature subsets with similar geometric properties. This is common in man-made environments where the layout of objects is often highly structured. Protection against environmental symmetry failure can be obtained by combining the inter-feature constraints of MCS with standard vehicle pose constraints.

There are cases where even the combined constraints are not sufficient to ensure correct mapping. In these situations, additional strategies such as multi-hypothesis tracking may be required.

4.2.1 Pose constrained MCS

Given *a priori* information concerning the location of the vehicle (or the relative pose between the two feature sets), batch data association can be further restricted by prohibiting associations between distant feature pairs. Consider the following example where the map features are exactly known (in global space) and an observed feature is also exactly known (relative to the vehicle). The location of the vehicle in the global coordinate system is known to exist within a bounded uncertainty region (see Figure 8). That is, the nominal vehicle location $[x_v, y_v, \phi_v]^T$ is given and the true location is within $[\pm a, \pm b, \pm c]^T$ of this location. On the basis of this information, if the nominal location of

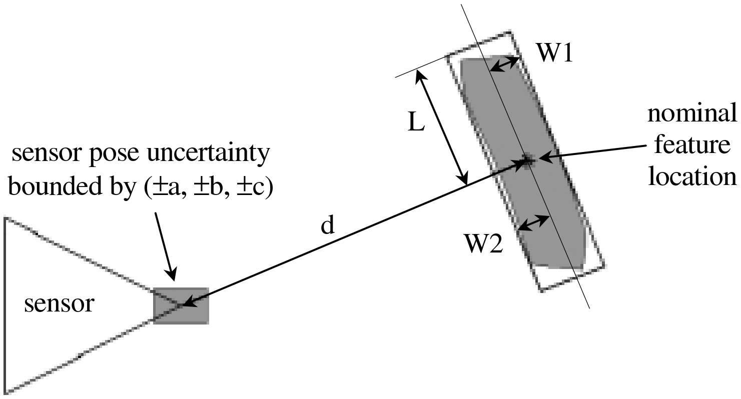


Fig. 8. Conservative bounding box. Given a bounded pose uncertainty, the uncertainty of a feature location is depicted by the shaded area. This region is bounded by a rectangular box with the dimensions shown.

the observed feature in global coordinates is (x, y), the true location of the observed feature may be anywhere in the region bounded by the shaded area.

A conservative bounding box that encompasses the observed feature uncertainty region may be calculated from the following parameters.

$$\begin{aligned} d &= \sqrt{x^2 + y^2} \\ W1 &= \sqrt{a^2 + b^2} \\ W2 &= W1 + d(1 - \cos c) \\ L &= W1 + d \tan c \end{aligned}$$

Therefore, only map features that fall within this box region are considered as possible associations to the observed feature. These pose-based constraints can be incorporated into the MCS algorithm to reinforce the constraints imposed by the inter-feature relationships. In context of the maximum clique implementation of the MCS algorithm [5], these additional constraints restrict the generation of nodes in the correspondence graph (i.e., the graph representing possible associations between the two feature sets).

The above example describes pose constrained MCS for a situation where only bounded uncertainty information is available. This approach may be useful if the estimation process being used does not offer more explicit information concerning the vehicle state. An expanded version of the graph-theoretic data association technique can be found in [3] which integrates the pose constrained MCS algorithm completely with the EKF estimation process—thus enabling the use of correlation information between vehicle and feature states to obtain an optimal association set.

4.2.2 Monte Carlo tracking

A simple alternative tracking algorithm involves selecting a random sample of poses from the predicted pose distribution and, for each sample, performing nearest-neighbour data association between the two feature sets. The best data association set is found simply from the sample with the largest number of valid nearest neighbour data associations. For a closely tracked vehicle with small pose uncertainty, this method can be implemented with low computational cost and works reliably when the number of sampled poses sufficiently covers the distribution space.

This Monte Carlo approach should not be confused with the Monte Carlo localisation algorithm of Dellaert *et al.* [10]. This method is a search for a maximum set of feature associations only and does not produce a sample-based pose distribution approximation. The maximum set of associations found is sent to a separate estimation process which calculates the pose estimate. In fact, rather than performing Monte Carlo pose sampling, the same result could be obtained by taking a few pose samples within the region and implementing a hill-climbing search that maximises the number of valid associations.

5 Dead Reckoning

Odometry has long been the mainstay of mobile robot dead reckoning but there are situations where odometric information can be very unreliable. This is particularly the case with non-holonomic vehicles in all-terrain outdoor environments.

The first concern arises with a vehicle's kinematic motion model since errors in the kinematic model result in systematic biases in the odometric motion estimate. A typical Ackerman steered vehicle requires accurately calibrated wheel alignment and steering angle, and a precise estimate of the wheel radius and wheel-base. Wheel radius is particularly difficult to estimate as it can change over time with wear or load changes.

The second problem is the non-kinematic dynamics of the vehicle motion due to slip. The amount of slip is dependent on the speed and steering angle of the vehicle and is particularly an issue with outdoor vehicles where different surfaces will produce varying degrees of slip. Also, some kinematic systems are more prone to slip than others. For example, articulated skid-steered vehicles, such as a mining LHD vehicle [27], have very large amounts of slip induced when turning.

An alternative to odometric dead reckoning is sensor based dead reckoning where observations are compared sequentially to obtain a relative motion estimate. This has the advantage of being free from vehicle models so that estimates do not suffer from model dependent errors. A second advantage is that the dead reckoning system becomes a self-contained module and can be transferred from one vehicle to another without change. Some disadvantages of sensor based dead reckoning are its reliance on regions with observable static features and the possibility of tracking failures through bad data associations or tracking moving objects. Tracking failure through lack of features is easily detectable but often incorrect data associations and mistaking slow moving objects for static ones are not. They can fail subtly and give motion estimates that are reasonable and consistent but wrong.

Clearly a desirable compromise would be the redundant use of odometry and sensor based dead reckoning so that a fused estimate could be obtained during normal operation and discrepancies could signal fault situations. Determining the source of fault (i.e., tracking failure or slip) is a more difficult issue and outside the scope of this paper. Frequency domain fault detection [28] prescribes the need for a third sensor (e.g., inertial platform) to statistically determine the fault source.

A common example of sensor-based dead reckoning is optical flow which estimates motion without the requirement of a vehicle model and so is easily installed as self-contained retrofit module. Optical flow [6] computes an estimate of 3D motion by generating a vector field of image intensity flow between sequential image frames. Its main weaknesses are its high computational cost (often requiring specialised hardware) and its unreliability in general environments. Several assumptions must be (approximately) met for reliable quantitative results. These include locally uniform illumination, Lambertian surface reflectance, and pure translation parallel to the image plane. It is also necessary to have a high frame rate relative to the rate of motion to aid correlation between sequential frames.

The optical flow approach is a high frequency motion estimator needing high bandwidth data for correct operation. The laser-based dead reckoning method proposed here, however, is relatively low frequency and does not require small motions between scans to ensure proper feature correspondence and so forth. The basic requisite of the algorithm is a connectedness between scans where, given two scans, there exist features common to both scans. By integrating the change in pose of the feature sets over sequential scans, a motion estimate is obtained.

Consider two sequential laser scans where the feature associations between the two sets have been correctly determined using batch data association. Laser-based dead reckoning involves using these associations to find the relative pose between the two scans and hence the change-in-pose estimate of the vehicle. One simple method for calculating the relative pose between the two sets of point-location features is given in Appendix C of [22]. However, this method does not consider the covariance of each feature and does not produce an estimate of the relative pose covariance.

A technique for computing the relative pose and its covariance is to implement a form of piece-wise SLAM such that, for each sequential pair of laser scans, the following steps are taken. First, an augmented SLAM vector $\hat{\mathbf{x}}_a = [\hat{\mathbf{x}}_v, \hat{\mathbf{x}}_m]^T$ is initialised where $\hat{\mathbf{x}}_v = [\hat{x}_v, \hat{y}_v, \hat{\phi}_v]^T$ is the vehicle pose estimate and $\hat{\mathbf{x}}_m = [\hat{x}_1, \hat{y}_1, \ldots, \hat{x}_n, \hat{y}_n]^T$ are the point features from the first scan. The location of the vehicle relative to the first scan set is defined as $\hat{\mathbf{x}}_v = [0, 0, 0]^T$ with covariance $\mathbf{P}_{vv} = \mathbf{0}$. The features in $\hat{\mathbf{x}}_m$ are not correlated to each other so the feature location covariance $\mathbf{P}_{mm}$ is block-diagonal (i.e., the only cross-correlations are between x-y values of each feature). Also, the feature estimates are not correlated to the vehicle pose so that the SLAM covariance is

$$\mathbf{P}_{aa} = \begin{bmatrix} \mathbf{P}_{vv} & \mathbf{0} \\ \mathbf{0} & \mathbf{P}_{mm} \end{bmatrix}$$

The next step is to propagate the vehicle location to approximately the relative pose of the second scan. This value may be obtained using the method of Lu and Milios [22]. This estimate, though, is assumed to be completely uncertain and the diagonals of $\mathbf{P}_{vv}$ are set to practically infinity (10^8, say). (Note that the approximate pose estimate is only required to minimise linearisation errors in the EKF update.)

Finally, each feature in the second data set is observed using standard SLAM update equations [16]. The resulting SLAM estimate $\hat{\mathbf{x}}_a$ includes the estimate of the vehicle pose $\hat{\mathbf{x}}_v$ and its covariance which can be extracted directly from the augmented SLAM vector. It is important to note here that the change-in-pose estimate is correlated to the scan information (i.e., the cross-correlations $\mathbf{P}_{vm} \neq \mathbf{0}$) and, as each scan is used twice for sequential scan dead reckoning, the change-in-pose estimates $\hat{\mathbf{x}}_v$ are pair-wise correlated. If dead reckoning estimates are accumulated over several scans, the change-in-pose covariance must be adjusted to avoid over-optimistic uncertainty accumulation. This may be dealt with conservatively by simply expanding each change-in-pose covariance $\mathbf{P}_{vv}$ by a factor of two (ensuring against the worst case where exactly the same information is used twice). Therefore, the change-in-pose estimate is given by $\hat{\mathbf{x}}_\delta = \hat{\mathbf{x}}_v$ and $\mathbf{P}_\delta = 2\mathbf{P}_{vv}$.

The piece-wise SLAM algorithm described above is applied separately for each sequential pair of laser scans to obtain each change-in-pose estimate $\hat{\mathbf{x}}_\delta$. However, in some circumstances it may be desirable to combine a sequence of change-in-pose vectors to determine an accumulated dead reckoning estimate. This vector addition requires propagation of the covariance estimate. That is, the addition of a new change-in-pose $\hat{\mathbf{x}}_\delta$ to the previous (dead reckoning) pose estimate $\hat{\mathbf{x}}_k$ is given by the following prediction equation (as shown in Figure 9).

$$\begin{bmatrix} \hat{x}_{k+1} \\ \hat{y}_{k+1} \\ \hat{\phi}_{k+1} \end{bmatrix} = \mathbf{f}\left(\hat{\mathbf{x}}_k, \hat{\mathbf{x}}_\delta\right) = \begin{bmatrix} \hat{x}_k + \hat{x}_\delta \cos\hat{\phi}_k - \hat{y}_\delta \sin\hat{\phi}_k \\ \hat{y}_k + \hat{x}_\delta \sin\hat{\phi}_k + \hat{y}_\delta \cos\hat{\phi}_k \\ \hat{\phi}_k + \hat{\phi}_\delta \end{bmatrix}$$

and the covariance $\mathbf{P}_{k+1}$ is therefore given by

$$\mathbf{P}_{k+1} = \nabla\mathbf{f}_{\mathbf{x}_k}\mathbf{P}_k\nabla\mathbf{f}_{\mathbf{x}_k}^T + \nabla\mathbf{f}_{\mathbf{x}_\delta}\mathbf{P}_\delta\nabla\mathbf{f}_{\mathbf{x}_\delta}^T$$

where the Jacobians $\nabla\mathbf{f}_{\mathbf{x}_k}$ and $\nabla\mathbf{f}_{\mathbf{x}_\delta}$ are as follows.

$$\nabla\mathbf{f}_{\mathbf{x}_k} = \frac{\partial\mathbf{f}}{\partial\mathbf{x}_k}\Big|_{\left(\hat{\mathbf{x}}_k, \hat{\mathbf{x}}_\delta\right)} = \begin{bmatrix} 1 & 0 & -\hat{x}_\delta \sin\hat{\phi}_k - \hat{y}_\delta \cos\hat{\phi}_k \\ 0 & 1 & \hat{x}_\delta \cos\hat{\phi}_k - \hat{y}_\delta \sin\hat{\phi}_k \\ 0 & 0 & 1 \end{bmatrix}$$

$$\nabla\mathbf{f}_{\mathbf{x}_\delta} = \frac{\partial\mathbf{f}}{\partial\mathbf{x}_\delta}\Big|_{\left(\hat{\mathbf{x}}_k, \hat{\mathbf{x}}_\delta\right)} = \begin{bmatrix} \cos\hat{\phi}_k & -\sin\hat{\phi}_k & 0 \\ \sin\hat{\phi}_k & \cos\hat{\phi}_k & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

Besides the need to expand each change-in-pose uncertainty $\mathbf{P}_{vv}$ by two, the dead reckoning estimate uncertainty tends to be much greater than would be expected from the sensor accuracy alone. This is because the uncertainty for each feature measurement must be magnified to incorporate various unmodelled factors in the real environment. One unmodelled component is the non-ideal shape of the physical objects in the world that do not exactly fit the point or edge feature models. For example, trees in the area are not perfect vertical cylinders but have variable non-circular trunk cross-sections. Another non-ideal factor in outdoor environments is the rough 3D ground surface which is modelled in these experiments as flat 2D terrain. The projection onto a 2D plane distorts the sensor measurements and it is necessary to expand the measurement uncertainty to cater for a presumed worst-case level of distortion.

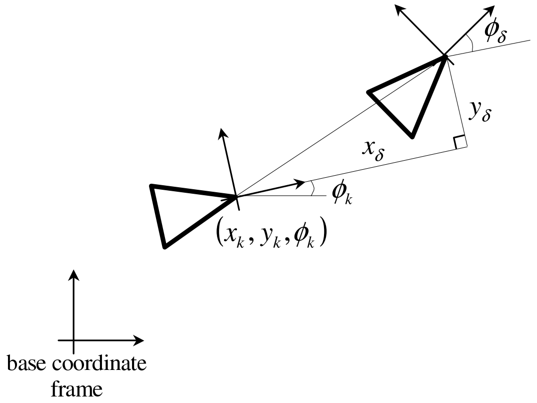


Fig. 9. Addition of change-in-pose vectors $\mathbf{x}_\delta$ to obtain an accumulated dead reckoning estimate.

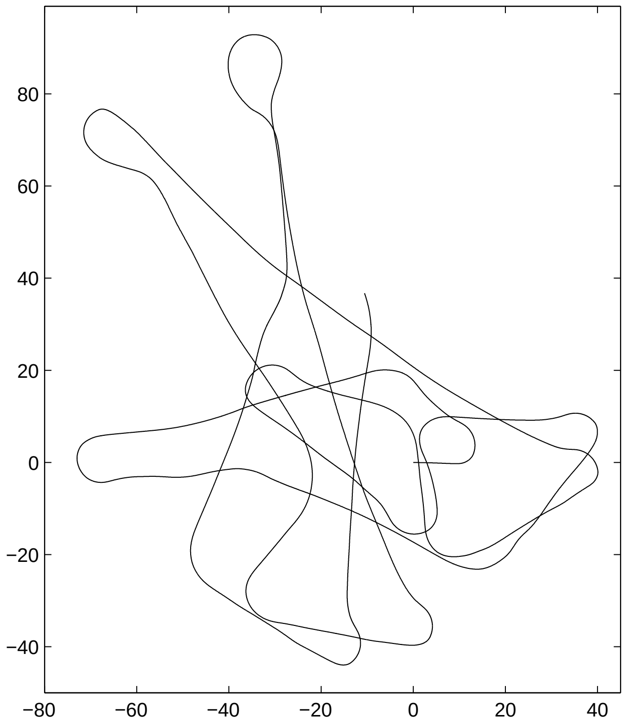


Fig. 10. Encoder dead reckoning. Unmodelled properties of the vehicle motion, such as wheel slip and non-linear parameters, contribute to rapid deterioration in the odometric pose estimate.

A comparison of odometry and laser-based dead reckoning is now presented. This data set was logged in the park environment given in Figure 3. Figure 10 shows the estimated vehicle trajectory derived from wheel and steering encoder odometry. The considerable pose error, when compared with the SLAM [16] ground truth given in Figure 11, is mostly due to non-linearities and wheel slip at high steer angles. The laser-based dead reckoning results are shown in Figure 12. It can be clearly seen that this method is unaffected by these factors and displays remarkable accuracy over a significant period of time.

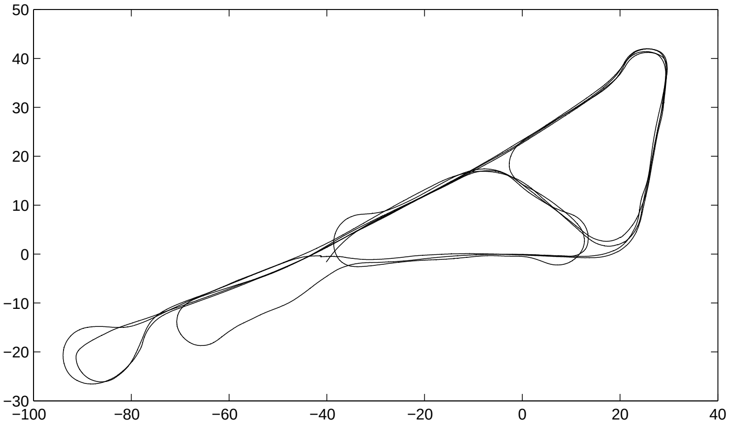


Fig. 11. SLAM-based ground truth.

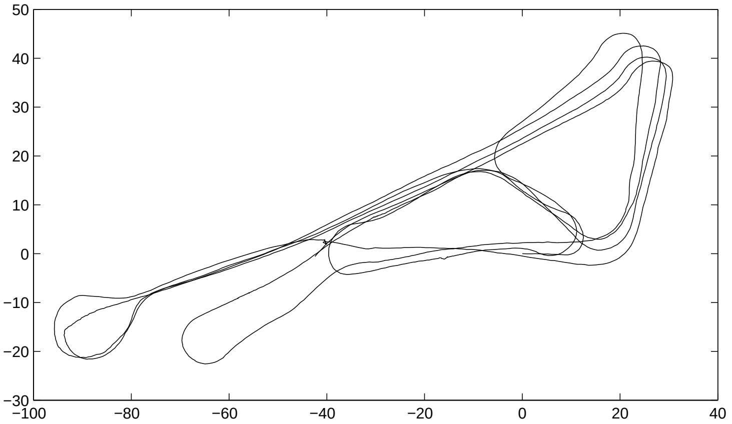


Fig. 12. Laser dead reckoning. The matching of sequential laser scans produces a pose estimate that drifts considerably more slowly than odometry and does not require a vehicle model.

5.3 Persistent Feature Filter

Sequential laser scan matching serves a second purpose in filtering out temporary and dynamic features while tracking static features. Within a single scan there are generally numerous false, noisy, or non-static features which, among other things, may be due to inconsistent clustering of the data set, undulation of the ground, or people and other objects moving through the scene. A simple form of persistent feature filter was implemented for these experiments where those features common to both the current scan and its predecessor are kept and the rest rejected, thus removing a substantial source of unreliable information (often between 50% and 90% of features in a scan).

A more sophisticated version of the persistent feature filter may incorporate a record of the number of scans over which a feature has been tracked and the distance the sensor has moved during this time. This would indicate the reliability and visibility of a feature. In particularly dynamic environments, it may also be desirable to track static features that are temporarily occluded or missed for a few scans. This would involve propagating a placeholder feature relative to the sensor motion and checking whether it reappears within a short distance.

6 Localisation Via Topological Feature Maps

Localisation was tested using experimental data from the two environments described previously. The topological feature maps were constructed manually from logged laser data and a global "ground truth" of the vehicle pose. The ground truth for the park environment was generated from the laser and encoder information using an optimal SLAM algorithm [16], while the ground truth for the suburban street area was provided by a differential GPS unit. Neither of these alternative location measures gave an ideal true path but they were sufficient to indicate the reliability of the localisation method and demonstrate the bounded nature of its error growth.

6.1 Implementation Details

Given that each experimental data set consisted of a timestamped laser scan set and a timestamped true path, the topological feature maps were generated manually as follows:

- Node locations were selected at points on the ground truth path to cover the area in question with an approximately even distribution. The street environment nodes were selected to be approximately equi-spaced over the region travelled (e.g., every 50m).
- The laser data time-stamp nearest to each node location time-stamp was found. Linear interpolation was used to align the node location with the laser scan location.
- For each node, the features extracted from the appropriate laser scan were stored as the node definition. This meant that each node was defined by features as seen within a 180° field-of-view in the direction of the positive x-axis.
- Nearby nodes were connected and their relative poses stored.

The vehicle pose within the map was initialised by a rough guess limited, in this implementation, by a search of only the nearest node. That is, a non-tracking batch data association was performed on the node known to be nearest the vehicle true location to initialise the pose estimate.

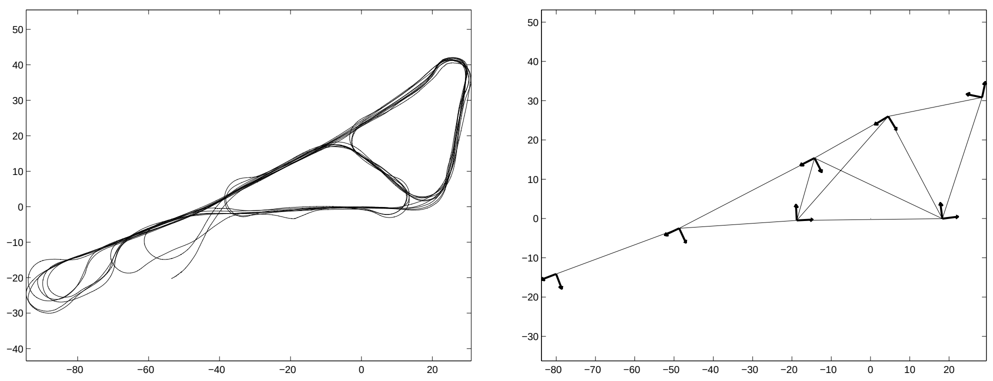
Tracking was performed such that the pose estimate of the vehicle relative to the current node was determined entirely by the latest laser scan—ignoring all prior pose information. In fact, prior pose information was merely used to indicate the current node context and to detect false mappings (via their over-large change-in-pose value). In these experiments, no incorrect mappings occurred even though only non-tracking batch data association was used to register observed data to the map nodes. The particular implementation of the tracking scheme was as follows:

- The vehicle motion (dead reckoning) estimate was obtained using the Monte Carlo tracking data association method on sequential scans and accumulating the change-in-pose estimates.
- An estimate (tagged as *reliable*) was maintained of the vehicle pose and covariance relative to the "most visible" node. This estimate had two modes, tracking and prediction.
- If the reliable estimate was in tracking mode and the number of features associated between the current observation set and the node feature set was greater than a threshold, then the pose estimate was updated directly with this observation information. If, on the other hand, insufficient associations were made, the estimate would switch to prediction mode.
- While in prediction mode, the reliable estimate was based on accumulated dead reckoning information only and so suffered growing uncertainty. Since prediction mode essentially stated that the reliable estimate had lost track of the current node, a separate estimate (tagged *tentative*) was attempted by performing a non-tracking batch data association to the predicted most visible node. If the mapping was successful (i.e., had sufficient number of associations) then the tentative track was maintained over a number of scans for several metres. If this track remained consistent over the specified period, the reliable estimate switched to this result and reverted to tracking mode.
- After each new scan, and subsequent motion estimate, the pose of the vehicle relative to the nodes adjacent to the current node was calculated along with the visibility of the features of these nodes. If an adjacent node had more features in the vehicle's field-of-view than the current node, a context switch occurred from the current node to the more visible one.

6.2 Localisation Results

6.2.1 Park trial

The SLAM ground truth of the vehicle path for the park test is shown in Figure 13 with its corresponding manually constructed map. This trial lasted for 16 minutes with the vehicle travelling about 3.2 km. Over this period, the



(a) True path

(b) Topological feature map

Fig. 13. Park environment trial run. Seven nodes were selected from the SLAM true path data set and each node was defined by the single corresponding laser scan taken at its location.

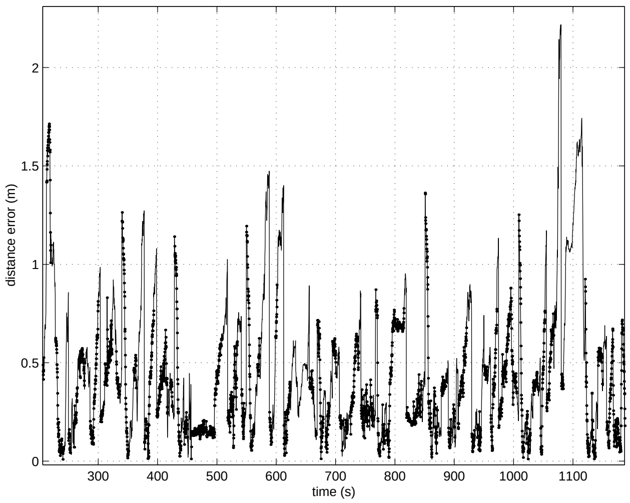
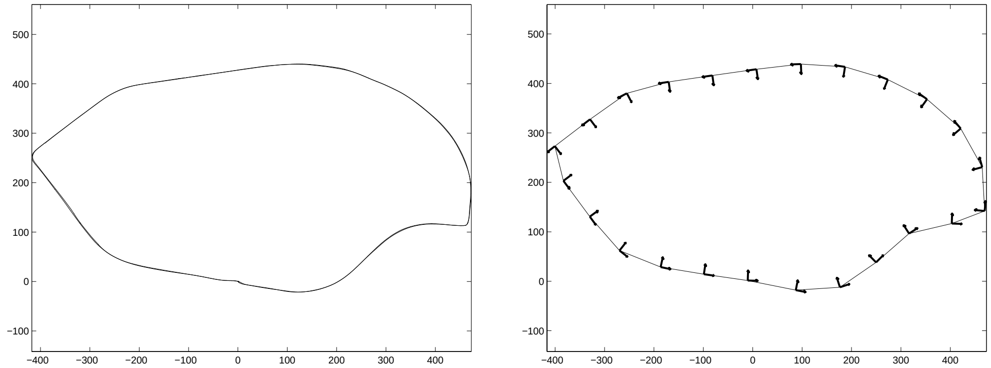


Fig. 14. Estimated error in the park environment.

localisation algorithm was in tracking mode 60% of the time. Of the remaining 40%, 27% was purely dead reckoning and 13% was relying on dead reckoning but with a tentative fix on the current map node. Figure 14 shows the distance error in the pose estimate compared with the ground truth estimate. The estimate error is generally less than one metre and, in fact, this error is mostly due to ground truth error and map alignment problems as discussed below. The important points to note with these results is the bounded error while tracking (tracking scans indicated by dots) and the recovery of tracking after periods of dead reckoning (e.g., around the time t = 1100 sec).



(a) True path

(b) Topological feature map

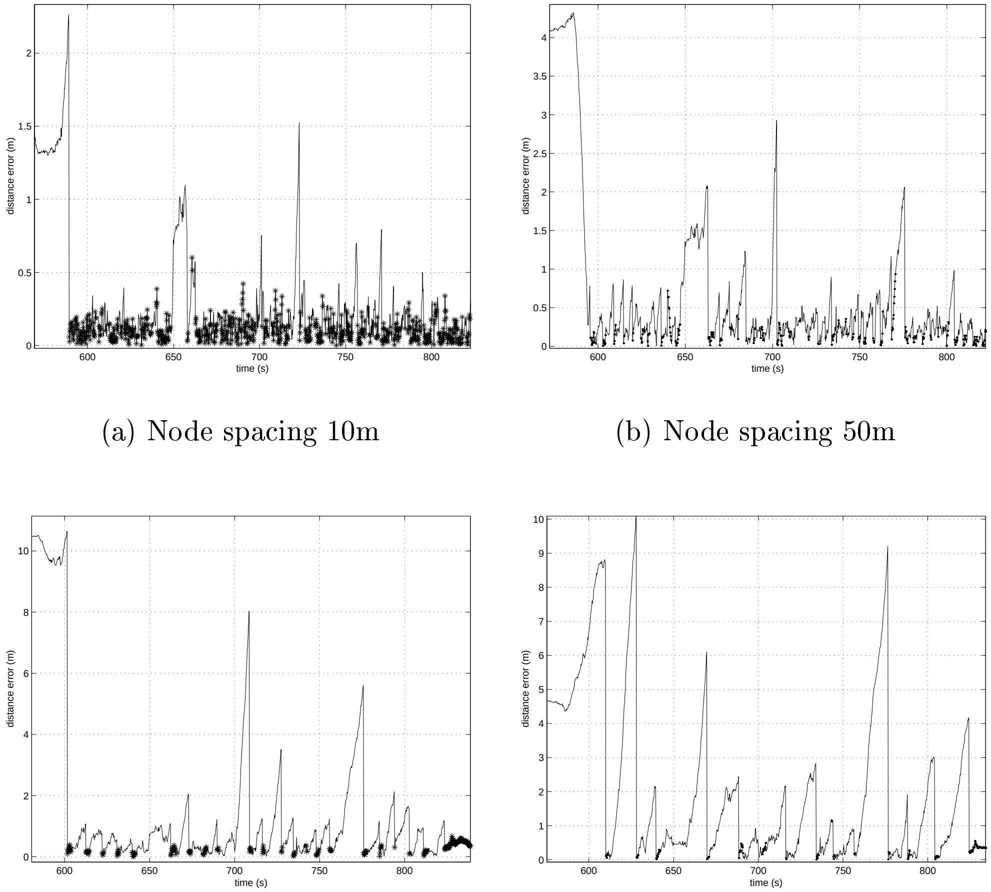
Fig. 15. Suburban street environment trial run. The nodes were placed at approximately equal spacing along the trial loop. Each node was defined by the single laser scan associated with the chosen DGPS true path location.

6.2.2 Suburban street trial

GPS ground truth of the street test run is shown in Figure 15 with a corresponding topological feature map. The trial run involved two loops of a suburban residential block taking 15 minutes to travel about 4.3 km. The results of the second loop are not shown here as the differential GPS (used for ground truth) gave unreliable readings during this loop. Nevertheless, the path tracked during this second loop was sufficiently similar to the first to presume that localisation was performed successfully.

Several maps were constructed for this route in order to test the behaviour of the algorithm with different node spacings. Spacings of 10m, 50m, 90m and 150m were tested and their results are shown in Figure 16. The black stars on the plots indicate scans where a tracking mode estimate is obtained. For each map, the vehicle travelled in tracking mode for 60%, 25%, 20%, and 12% of the time respectively and relied on dead reckoning for 30%, 68%, 75%, and 85% of the time. The differential GPS provided a more reliable ground truth (usually $\sim 10 - 20cm$) than was available for the park trial but, with the problem of map node alignment, the determination of error source remained somewhat ambiguous.

During tracking mode, the distance error was usually less than 0.5m and, when this was not the case, it is probable that the error was in the map alignment (see below) and not in the tracking algorithm. The accumulated error due to prolonged dead reckoning was proportional to the spacing of the map nodes as would be expected and, particularly apparent in the map with 150m node spacing, the localisation algorithm was able to regain track when the pose



(a) Node spacing 10m

(b) Node spacing 50m

(c) Node spacing 90m

(d) Node spacing 150m

Fig. 16. Estimated error in the street environment.

estimate error was substantial.

6.3 Results Analysis

As can be seen from the experimental results, this localisation method presents very consistent behaviour in different types of environment. Nevertheless, it is important to determine the sources of localisation errors and discuss the implications of these errors on the results.

When localising relative to a particular node, the only sources of vehicle pose errors are in the observation (scan) data and the feature set defining the node itself. These errors are quite small—simply the sensor accuracy and some non-modelled quantities such as non-ideal feature shape and terrain slope. However, to relate the relative pose estimate back to the ground truth, the relative pose must be transformed to the global coordinate system (i.e., the relative pose is converted to a global pose given the known global pose of the node). This transformation introduces a substantial error source in terms of node alignment.

The estimated error results depicted in this paper were found by obtaining the vehicle pose estimate relative to a map node and transforming this estimate by the global node pose. The resulting global vehicle pose was then compared with the global pose of the "true path" at the same timestamp. The estimated error shown in the graphs is simply the distance between the two global pose estimates.

Thus, there are essentially three sources of error in the vehicle pose error estimate. The first is the fairly small error in the actual relative pose estimate. The second is the error in the true path global estimate at the given timestamp. The third, and perhaps most significant, is the error in the global pose estimate of the node. This last error source is important largely in terms of the node's angular alignment because the effect of misalignment increases with the distance of the vehicle from the node origin. Even a small angular misalignment can cause the estimated global error to appear large if the vehicle is 50 metres from the node.

In the park environment the true path error was often in the order of one metre and so presented the main error source. On the other hand, in the street environment, the DGPS ground truth was very accurate but correctly aligning the nodes was difficult and, particularly for the wider node spacings, created the main pose error. In both situations, these errors tended to hide the actual localisation error which was the vehicle pose estimate relative to the map node.

Another minor influence on the rather pessimistic error estimate results was because the nodes were chosen at arbitrary locations where good feature information may not have been available. A measure of the information contained in each scan feature set may have enabled the selection of better quality node candidates.

Finally, despite node misalignment, the suburban street tests demonstrate that the node spacing directly affects the localisation accuracy. If the node spacings are larger than the node regions, such that the feature sets do not overlap, there will be regions where the vehicle must navigate using dead reckoning only. Conversely, if the node regions do overlap, the vehicle will be able to maintain track of its pose with, perhaps, intermittent periods of dead reckoning due to occlusion or dynamic objects. This issue is important because, although the non-tracking batch feature mapping method increases data association robustness, it cannot guarantee correctness. Constraining association with pose information is required in most environments to prevent divergence and, therefore, it becomes necessary for the vehicle pose uncertainty to be minimal (i.e., to maintain tracking mode as much as possible). Further confidence checks for correct data association may be required in environments with a lot of symmetry in which case a multi-hypothesis implementation may be needed.

7 Future Research

One of the most challenging problems of the current implementation is the feature extraction process. By only selecting point and edge features (or other simple geometric primitives such as lines or corner points) a large portion of the available information is discarded. This makes tracking in unstructured environments like the suburban street more difficult than it should be and reduces data association reliability by ignoring distinguishing features within a scan. The primary difficulty with improving feature extraction is defining general parametric feature representations that can be augmented with subsequent information.

Extending the current implementation to incorporate simultaneous map building is presently being investigated [3]. This involves spawning new nodes and augmenting the feature sets representing old nodes. Updating feature information within a node region could be performed using standard SLAM algorithms while the relative pose between nodes need only be loosely coupled. If the node regions were made to overlap, the result would approach the accuracy of optimal SLAM. Perhaps the most difficult problem with extending this framework into SLAM is the detection of cycles where one section of the map rejoins an old section (i.e., a graph self-intersection). Detecting cycles in a reliable and scalable manner will be the main issue in this research.

8 Conclusion

The topological feature map described in this paper was capable of reliably localising a vehicle in the two environments tested. These environments were large outdoor regions and sufficiently diverse to suggest that the method could have broad application.

The specific implementation presented here used a 2D scanning laser as the sole information source (although other sources were used to validate the results). Batch feature data association was used to track the vehicle's pose in the map and also to provide a dead reckoning estimate while ever tracking was lost. The dead reckoning result was reliable and accurate in situations where odometry suffered from slip and other non-linearities. It is likely that laser-based dead reckoning and odometry could be used in a complementary fashion where the laser detects slip and odometry operates during tracking failure (from lack of features etc). Using the laser information for dead reckoning and tracking meant that a pose estimate could be maintained without a vehicle model. This implies simple modular installation and freedom from problems concerned with kinematic modelling errors and vehicle dynamics.

The two experimental trials did not use vehicle pose information to constrain data associations but still operated successfully. It is likely that environments would exist where pose constraint information would become necessary due to environmental symmetries but these results indicate the inherent robustness of the non-tracking batch data association approach. It was also shown that the vehicle was able to relocalise after long periods of dead reckoning and with significant pose estimate error. Similarly, localisation was not sensitive to the initial conditions.

References

[1] S. Argamon-Engelson. Using image signatures for place recognition. *Pattern Recognition Letters*, 19:941–951, 1998.

[2] O. Aycard, F. Charpillet, D. Fohr, and J.F. Mari. Place learning and recognition using hidden markov models. In *IEEE International Conference on Intelligent Robots and Systems*, pages 1741–1746, 1997.

[3] T. Bailey. *Mobile Robot Localisation and Mapping in Large Outdoor Environments.* PhD thesis, University of Sydney, Australian Centre for Field Robotics, 2001.

[4] T. Bailey, E.M. Nebot, J.K. Rosenblatt, and H.F. Durrant-Whyte. Robust distinctive place recognition for topological maps. In *International Conference on Field and Service Robotics*, pages 347–352, 1999.

[5] T. Bailey, E.M. Nebot, J.K. Rosenblatt, and H.F. Durrant-Whyte. Data association for mobile robot navigation: A graph theoretic approach. In *IEEE International Conference on Robotics and Automation*, volume 3, pages 2512–2517, 2000.

[6] S.S. Beauchemin and J.L. Barron. The computation of optical flow. *ACM Computing Surveys*, 27(3):433–467, 1995.

[7] R.C. Bolles and R.A. Cain. Recognizing and locating partially visible objects: The local-feature-focus method. *International Journal of Robotics Research*, 1(3):57–82, 1982.

[8] W. Burgard, A. Derr, D. Fox, and A.B. Cremers. Integrating global position estimation and position tracking for mobile robots: The dynamic markov localization approach. In *IEEE/RSJ International Conference on Intelligent Robots and Systems*, 1998.

[9] J.L. Crowley, F. Wallner, and B. Schiele. Position estimation using principal components of range data. In *IEEE International Conference on Robotics and Automation*, pages 3121–3128, 1998.

[10] F. Dellaert, D. Fox, W. Burgard, and S. Thrun. Monte Carlo localization for mobile robots. In *IEEE International Conference on Robotics and Automation*, pages 1322–1328, 1999.

[11] M.W.M.G. Dissanayake, P. Newman, H.F. Durrant-Whyte, S. Clark, and M. Csorba. A solution to the simultaneous localization and map building (SLAM) problem. Technical report, University of Sydney, Australian Centre for Field Robotics, 1999.

[12] A. Elfes. Occupancy grids: A stochastic spatial representation for active robot perception. In *Sixth Conference on Uncertainty in AI*, 1990.

[13] S.P. Engelson and D.V. McDermott. Error correction in mobile robot map learning. In *IEEE International Conference on Robotics and Automation*, pages 2555–2560, 1992.

[14] N.J. Gordon, D.J. Salmond, and A.F.M. Smith. Novel approach to nonlinear/non-Gaussian Bayesian state estimation. *IEE Proceedings-F*, 140(2):107–113, 1993.

[15] J. Guivant and E. Nebot. Optimization of the simultaneous localization and map building algorithm for real time implementation. *IEEE Journal of Robotic and Automation*, submitted 2000.

[16] J. Guivant, E. Nebot, and H.F. Durrant-Whyte. Simultaneous localization and map bulding using natural features in outdoor environments. In *Sixth International Conference on Intelligent Autonomous Systems*, volume 1, pages 581–588, 2000.

[17] R. Horaud and T. Skordas. Stereo correspondence through feature grouping and maximal cliques. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 11(11):1168–1180, 1989.

[18] D. Kortenkamp and T. Weymouth. Topological mapping for mobile robots using a combination of sonar and vision sensing. In *Twelfth National Conference on Artificial Intelligence*, pages 979–984, 1994.

[19] B. Kuipers and Y.T. Byun. A robot exploration and mapping strategy based on a semantic hierarchy of spatial representations. *Journal of Robotics and Autonomous Systems*, 8:47–63, 1991.

[20] J.J. Leonard, H.F. Durrant-Whyte, and I.J. Cox. Dynamic map building for an autonomous robot. *International Journal of Robotics Research*, 11(4):286–298, 1992.

[21] J.J. Leonard and H.J.S. Feder. A computationally efficient method for large-scale concurrent mapping and localization. In *Ninth International Symposium on Robotics Research*, pages 316–321, October 1999.

[22] F. Lu and E. Milios. Robot pose estimation in unknown environments by matching 2D range scans. *Journal of Intelligent and Robotic Systems*, 18:249–275, 1997.

[23] S. Majumder, J. Rosenblatt, S. Scheding, and H. Durrant-Whyte. Map building and localization for underwater navigation. In *International Symposium On Experimental Robotics*, 2000.

[24] E.M. Nebot and D. Pagac. Quadtree representation and ultrasonic information for mapping an autonomous guided vehicle's environment. *International Journal of Computers and Their Applications*, 2(3):160–170, 1995.

[25] J. Neira and J.D. Tardós. Robust and feasible data association for simultaneous localization and map building. In *IEEE International Conference on Robotics and Automation, Workshop on Robot Navigation and Mapping*, 2000.

[26] D.B. Reid. An algorithm for tracking multiple targets. *IEEE Transactions on Automatic Control*, 24(6):843–854, 1979.

[27] S. Scheding, G. Dissanayake, E.M. Nebot, and H. Durrant-Whyte. An experiment in autonomous navigation of an underground mining vehicle. *IEEE Transactions on Robotics and Automation*, 15(1):85–95, 1999.

[28] S. Scheding, E. Nebot, and H. Durrant-Whyte. High-integrity navigation: A frequency-domain approach. *IEEE Transactions on Control Systems Technology*, 8(4):676–694, 2000.

[29] A.C. Schultz and W. Adams. Continuous localization using evidence grids. In *IEEE International Conference on Robotics and Automation*, pages 2833–2839, 1998.

[30] H. Shatkay and L.P. Kaelbling. Learning topological maps with weak local odometric information. In *Fifteenth International Joint Conference on Artificial Intelligence*, pages 920–927, 1997.

[31] R. Smith, M. Self, and P. Cheeseman. A stochastic map for uncertain spatial relationships. In *Fourth International Symposium on Robotics Research*, pages 467–474, 1987.

[32] A. Stevens, M. Stevens, and H.F. Durrant-Whyte. OxNav: Reliable autonomous navigation. In *IEEE International Conference on Robotics and Automation*, pages 2607–2612, 1995.

[33] H.D. Tagare, D.V. McDermott, and H. Xiao. Visual place recognition for autonomous robots. In *IEEE International Conference on Robotics and Automation*, pages 2530–2535, 1998.

[34] S. Thrun, W. Bugard, and D. Fox. A real-time algorithm for mobile robot mapping with applications to multi-robot and 3D mapping. In *International Conference on Robotics and Automation*, pages 321–328, 2000.

[35] J.K. Uhlmann. *Dynamic Map Building and Localization: New Theoretical Foundations*. PhD thesis, University of Oxford, Department of Engineering Science, 1995.

[36] I. Ulrich and I. Nourbakhsh. Appearance-based place recognition for topological localization. In *IEEE International Conference on Robotics and Automation*, pages 1023–1029, 2000.