14

Robust Autonomous Navigation and World Representation in Outdoor Environments

Favio Masson†, Juan Nieto‡, José Guivant‡, Eduardo Nebot‡
†Instituto de Investigaciones en Ingeniería Eléctrica, Universidad Nacional del Sur Argentina
‡ARC Centre of Excellence for Autonomous Systems, University of Sydney Australia

1. Introduction

Reliable localisation is an essential component of any autonomous vehicle system. The basic navigation loop is based on dead reckoning sensors that predict high frequency vehicle manoeuvres and low frequency absolute sensors that bound positioning errors. The problem of localisation given a map of the environment or estimating the map knowing the vehicle position has been addressed and solved using a number of different approaches. A related problem is when neither, the map nor the vehicle position is known. In this case the vehicle, with known kinematics, starts in an unknown location in an unknown environment and proceeds to incrementally build a navigation map of the environment while simultaneously using this map to update its location. In this problem, vehicle and map estimates are highly correlated and cannot be obtained independently of one another. This problem is usually known as Simultaneous Localisation and Map Building (SLAM).

As an incremental algorithm, the SLAM in large outdoor environments must address several particular problems: the perception of the environment and the nature of features searched or observables with the available sensors, the number of features needed to successfully localise, the type of representation used for the features, a real time management of the map and the fusion algorithm, the consistency of the SLAM process and the data association between features mapped and observations. A good insight into the SLAM problem can be found in Durrant-Whyte & Bailey (2006).

This chapter presents recent contributions in the areas of perception, representation and data fusion, focusing on solutions that address the real time problem in large outdoor environments. Topics such as DenseSLAM, Robust Navigation and non-Gaussian Observations in SLAM are summarised and illustrated with real outdoor tests.

2. Detailed environment representation

One of the main issues of the SLAM problem is how to interpret and synthesize the external sensory information into a representation of the environment that can be used by the mobile robot to operate autonomously. Traditionally, SLAM algorithms have relied on sparse

Source: Mobile Robots: Perception & Navigation, Book edited by: Sascha Kolski, ISBN 3-86611-283-1, pp. 704, February 2007, Plv/ARS, Germany

environment representations: maps built up of isolated landmarks observed in the environment (Guivant et al., 2002; Neira & Tardós, 2001). However, for autonomous navigation, a more detailed representation of the environment is necessary, and the classic feature-based representation fails to provide a robot with sufficient information. While a dense representation is desirable, it has not been possible for SLAM paradigms.
The next generation of autonomous systems will be required to operate in more complex environments. A sparse representation formed only by isolated landmarks will in general not fulfil the necessities of an autonomous vehicle, and a more detailed representation will be needed for tasks such as place recognition or path planning. Furthermore, not only is a dense representation of the environment required, but also an algorithm that is able to obtain *multi-layered maps*, where each layer represents a different property of the environment, such as occupancy, traversability, elevation, etc (Lacroix et al., 2002).

2.1 DenseSLAM

Mapping techniques that are able to handle vehicle uncertainty such as EKF-SLAM are not able to obtain dense representations due to the extremely high computational burden involved. On the other hand, mapping algorithms that are able to obtain detailed representations such as Occupancy Grids (Elfes, 1989) are known to have problems coping with vehicle pose uncertainty. The concept of DenseSLAM was introduced in (Nieto et al., 2004) as *the process of simultaneous vehicle localisation and dense map building*.
DenseSLAM is then a more ambitious problem than classic feature-based SLAM. A solution for DenseSLAM will have to deal with computational and consistency issues, arising from the dual purpose of trying to obtain a dense representation while simultaneously doing localisation.
This section presents the Hybrid Metric Maps. The Hybrid Metric Maps (HYMMs) algorithm (Guivant et al., 2004; Nieto et al., 2004) presents a novel solution for addressing the mapping problem with unknown robot pose. The HYMM is a mapping algorithm that combines feature maps with other metric sensory information. The approach permits the localisation of the robot and at the same time constructs a detailed environment representation (DenseSLAM). It is also a powerful technique to solve several practical problems (Masson et al., 2005)
Rather than incorporating all the sensory information into a global map, the algorithm maintains a features map and represents the rest of the sensed data in local maps defined relative to the feature positions. A joint state vector with the vehicle pose and the feature positions is maintained and the dense maps are stored in a separate data structure. When new observations are received, the state vector is augmented with the feature positions and the rest of the information is fused into the local regions. The main difference between feature-based SLAM and DenseSLAM is that feature-based SLAM incorporates the features into the map and neglects the rest of the information, whereas DenseSLAM has the ability to maintain all the sensory information to build a detailed environment representation.
The algorithm works as follows. When the robot starts to navigate, it will extract features from the environment that will be incorporated in the state vector. The feature map will be used to partition the global map into smaller regions, Fig. 1 illustrates this

process. The dense sensory information will be represented in these local regions. Fig. 2 shows a hypothetical dense map. The figure shows the division of the global map into smaller regions and the dense multi-layer maps obtained by DenseSLAM. Each of these layers depicts different environment properties. The global dense map consists of a set of local maps defined relative to the feature positions. Fig. 3 shows a basic flow diagram of the algorithm.

The main characteristic of DenseSLAM is the local representation used to fuse the dense information. The motivation behind the relative representation is to reduce correlations between states. Using this relative representation, the states represented in a local frame become strongly correlated and the states represented in different frames become weakly correlated. This is the key which allows the decorrelation of the dense maps with the rest of the system making the representation tractable.

Since the observations of the world are taken relative to the vehicle pose, any environment representation created will be correlated with the vehicle pose. Augmenting the state vector with all the information rendered by the sensors and maintaining the correlations is infeasible due to the computational burden involved. Therefore, DenseSLAM incorporates a set of landmarks in the state vector and the rest of the sensed data is decorrelated and stored in a separate data structure.

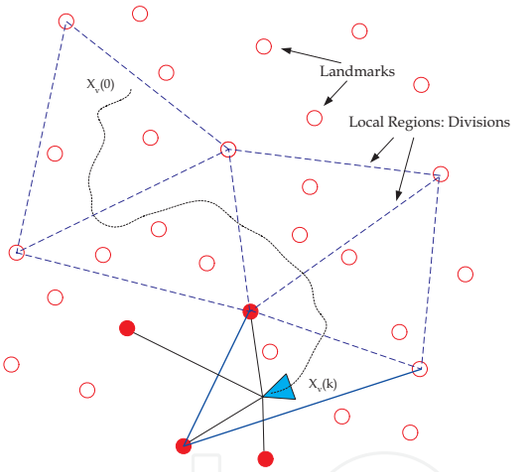


Fig. 1. Landmarks map ('o') and a particular partition of the global map in local regions. As shown, not all the landmarks are needed as vertex points in the regions definition.

The approximation made by the algorithm consists of representing the dense information in the local regions without including the correlations between the locally represented information and the rest of the system. These correlations will be zero only when there is full correlation between the local property (expressed in global coordinates) and the features that define the respective local frame (assuming the same uncertainty magnitude), so their relative positions are perfectly known. Although it can be proved that in a SLAM process the map becomes fully correlated in the limit (Gibbens et al., 2000), in practice only high correlation is achieved. However, it can be demonstrated that the assumptions made by the HYMM framework are, in practice, very good approximations for SLAM problems. The next paragraphs explain two well known properties of SLAM that justify the approximations made in the HYMMs.

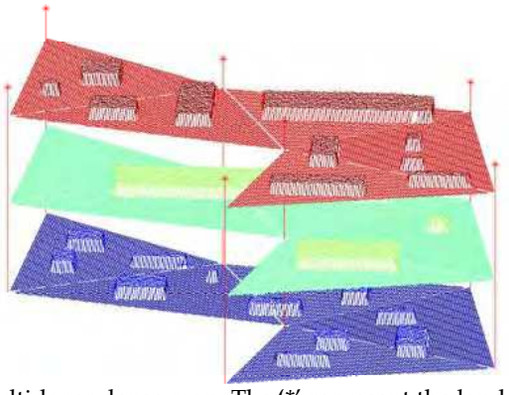


Fig. 2. Hypothetical multi-layer dense map. The '*' represent the landmark positions and the map layers depict different environment properties captured by the sensors.

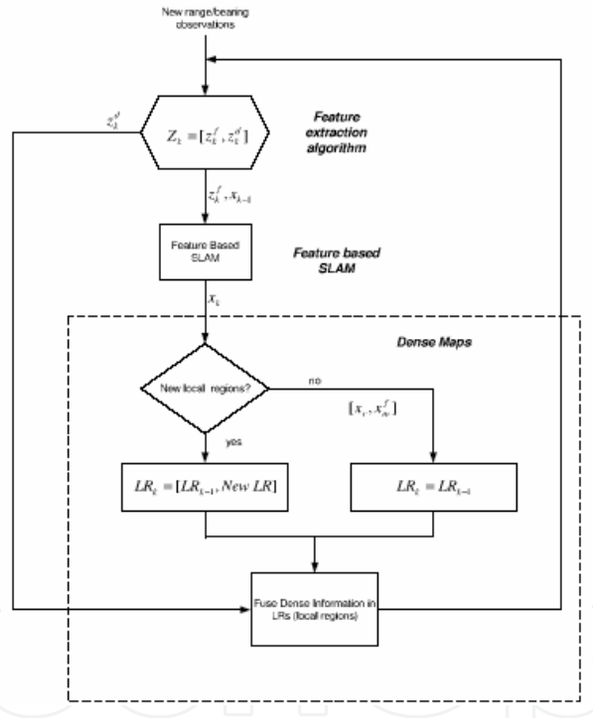


Fig. 3. HYMM algorithm flow diagram. When a sensor frame is obtained, first a feature extraction algorithm is applied and the features extracted are added to the feature-based SLAM. Then the algorithm looks for new local regions (LR) and fuses all the sensory information in the respective local frames. $\mathbf{z}_k^f$ represents the observations associated with features and $\mathbf{z}_k^d$ the rest of the observations (dense maps). $\mathbf{x}_v$ represents the vehicle position and $\mathbf{x}_m^f$ the feature map.

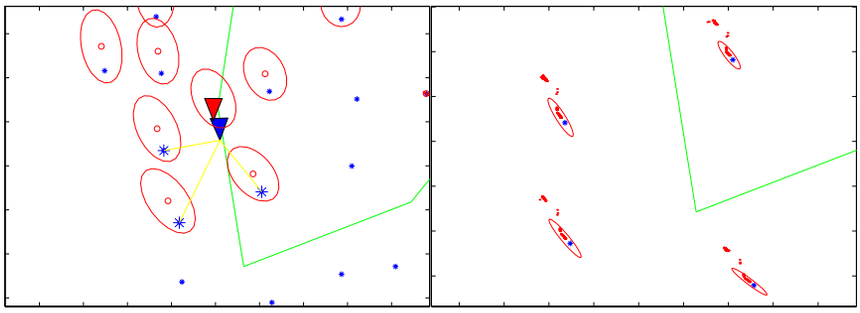
Geographically close objects have high correlation: If a set of observed objects is geographically close from the vehicle viewpoint, then the error due to the vehicle pose uncertainty will be a common

component of these estimated objects' positions. This is a typical situation in SLAM where the vehicle accumulates uncertainty in its estimated position and incorporates observations that are used to synthesize a map. Due to this fact the estimates of objects that are geographically close will present similar uncertainties (high cross-correlations). Any update of a particular object will imply a similar update of any object sufficiently close to the first one. Figure 4 shows an example of a typical SLAM map. The figure shows a landmarks map with its uncertainty bounds. It can be seen that landmarks that are geographically close have very similar uncertainty.

The relative representation stores close objects in local coordinate frames and then permits the reduction of correlation to the rest of the map (Guivant & Nebot, 2003): Assume a landmark can be represented in a local frame in the following way.

$$\mathbf{x}^L = \mathbf{h}(\mathbf{x}_v, \mathbf{z}, \mathbf{x}_b) \tag{1}$$

where $\mathbf{x}^L$ represents the relative landmark position, $\mathbf{x}_v$ the vehicle position, $\mathbf{z}$ the observations and $\mathbf{x}_b$ the position of the landmarks that define the local frame (base landmarks).



(a) (b)

Fig. 4: Map Correlation: The figures show that geographically close objects possess similar uncertainty. Figure (a) shows how the landmarks that are being observed have similar uncertainty to the robot pose. (b) shows how the estimated landmarks' means are updated after the vehicle closes the loop. The dots represent landmark position estimates over time. High correlation in geographically close objects is one of the SLAM characteristics; because the vehicle will observe close objects at similar instants it will propagate similar uncertainty to the objects.

Taking into account that the observation errors are independent of the vehicle position, the cross-correlation between the vehicle and the landmark in the local frame will be:

$$\mathbf{P}_{vL} = \mathbf{P}_{vv}\nabla\mathbf{h}_{\mathbf{x}_v}^T + \mathbf{P}_{vb}\nabla\mathbf{h}_{\mathbf{x}_b}^T \tag{2}$$

where $\mathbf{P}_{vv}$ represents the vehicle states covariance, $\mathbf{P}_{vb}$ the cross-correlation between the vehicle states and the base landmarks position estimated and $\nabla\mathbf{h}_{\mathbf{x}_i} = \frac{\partial \mathbf{h}}{\partial \mathbf{x}_i}$ is the Jacobian matrix of h with respect to the state xi.

Taking for example the one dimensional case, Equation (1) becomes:

$$x^L = h(x_v, z, x_b) = x_v + z - x_b \tag{3}$$

Applying Equation (3) to (2):

$$\mathbf{P}_{vL} = \mathbf{P}_{vv}(1) + \mathbf{P}_{vb}(-1) = \mathbf{P}_{vv} - \mathbf{P}_{vb} \tag{4}$$

Equation (4) shows that if the magnitudes of P_{vv} and the covariance of the base landmarks P_{bb} are similar, when the robot is highly correlated with the base landmarks there will be

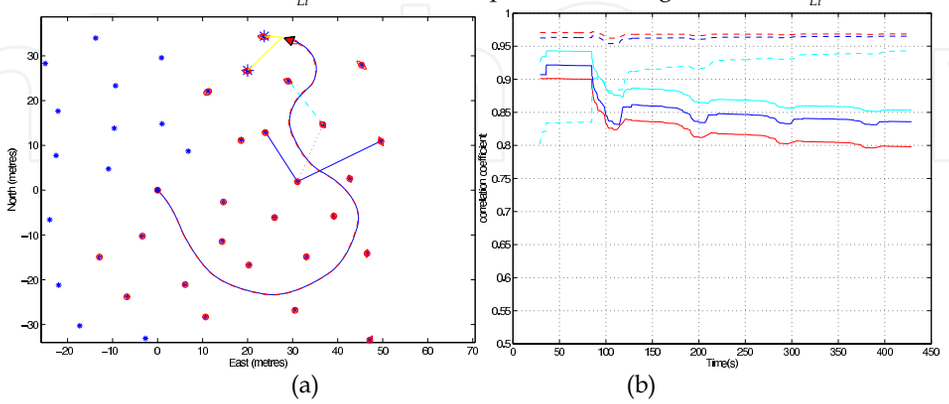
almost no correlation between the robot position and the local landmark (P_{vL}) and then no correlation between the relative landmark and the rest of the map. Since the relative and the base landmarks are geographically close, whenever the robot observes the local landmark it will be highly correlated with the base landmarks. This fact will reduce the correlation between the local landmark and the robot and therefore the correlation between the local landmark and the rest of the map will be reduced as well.

A more direct way of observing the decorrelation effect will be by evaluating the cross-correlation between a landmark in the global frame with a landmark represented in a local frame and comparing this with the cross-correlation of the same two landmarks, both represented in the global frame. In a similar manner to Equation (2), the cross-covariance matrix between the *j*-th landmark and a locally represented landmark can be evaluated in the following way:

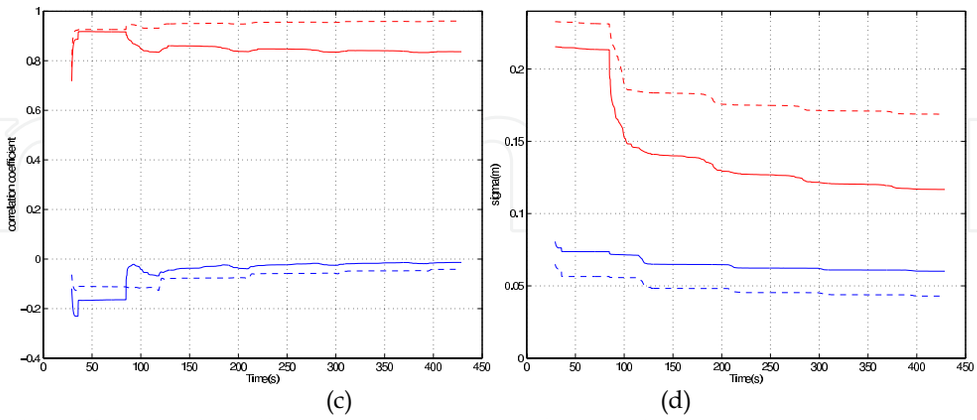
$$\mathbf{P}_{jL} = \mathbf{P}_{jb}\nabla\mathbf{h}_{\mathbf{x}_b}^{T} + \mathbf{P}_{jG}\nabla\mathbf{h}_{\mathbf{x}^G}^{T} \tag{5}$$

where the prefix *j* means the *j*-th landmark, *b* the base landmarks that define the local frame, *L* the locally represented landmark and *G* the position of the local landmark in the global frame. Then given $\mathbf{P}_{jb}$, $\mathbf{P}_{jG}$ and the transforming function from the global to the local frame **h**, it is possible to evaluate the cross-correlation between the local landmark and the *j* landmark in the map. Although the effect of decorrelation happens regardless of the particular local representation used, finding an expression to demonstrate that $\mathbf{P}_{jL} << \mathbf{P}_{jG}$ will be dependent on the local representation used. Equation (5) shows that for a particular local representation **h**, the decorrelation effect between the local object and the rest of the map will depend on the cross correlation between the rest of the map and the base landmarks and the cross-correlation between the rest of the map and the local represented object in the global frame.

Fig. 5 (a) shows the simulation environment utilised to illustrate the decorrelation effect. In the example, a local region is defined using three landmarks and another landmark is locally represented in this local frame. The *i*-th landmark is then represented in the local frame after a few observations which ensures high correlation with the base landmarks. After that, the landmark is not observed again, as it actually occurs with the dense sensory information (it is assumed that is not possible to observe exactly the same point of the environment more than once). The blue solid line in Fig. 5 (a) shows the local frame axis and the red dotted line the local landmark position $\mathbf{x}_{Li}$. The cyan dashed line joins the local represented landmark $\mathbf{x}_{Li}^{L}$ with a global landmark $\mathbf{x}_{Lj}$. The cross-correlation between $\mathbf{x}_{Lj}$ and the local *i*-th landmark will be evaluated when the *i* landmark is represented in the local frame $\mathbf{x}_{Li}^{L}$ and when it is represented in the global frame $\mathbf{x}_{Li}^{G}$.



(a) (b)



(c) (d)

Fig. 5: Decorrelation effect: (a) shows a zoom of the navigation environment where three landmarks were used to define a local region and one landmark (red dashed line) was represented in this local frame. (b) shows the correlation between the landmark represented in the global frame and the base landmarks. (c) shows the decorrelation effect; when the landmark is represented in local coordinates (blue line) the cross-correlation with other landmarks is considerably reduced in respect to the correlation between other landmarks and the landmark in global coordinates (red line). (d) shows the landmark deviation when it is represented in local (blue line) and global (red line) coordinates.

Fig. 5 (b) shows the evolution in time of the correlation coefficient of the i landmark represented in global coordinates $\mathbf{x}_{Li}^{G}$, with the landmarks used to define the local frame. The solid line depicts the cross-correlation in the east axis and the dashed line in the north axis. The different colours represent the cross-correlation with the different base landmarks. As can be seen, the landmark $\mathbf{x}_{Li}^{G}$ possesses high correlation with the base landmarks. This is due to the geographical proximity between the landmarks. Fig. 5 (c) shows the correlations between $\mathbf{x}_{Li}^{G}$ and $\mathbf{x}_{Lj}$ in red, and the correlations between the j landmark and the landmark i represented in the local frame $\mathbf{x}_{Li}^{L}$ (Equation (5)) in blue. The correlation was reduced from almost one when the landmark was represented in global coordinates, to almost zero when the landmark was represented in the local frame.

Finally Fig. 5 (d) shows the variance of the landmark i. The blue line depicts the variance when the landmark is in the local frame and the red line when it is in global. Because of the high correlation between $\mathbf{x}_{Li}^{G}$ and the base landmarks, the uncertainty in their relative position is very low, and so is the variance of $\mathbf{x}_{Li}^{L}$.

In summary the relative representation used by DenseSLAM permits the local represented information to be decorrelated with the rest of the system. This permits the incorporation of more information without increasing the computational cost.

2.2 DenseSLAM: Applications

This section shows how the detailed multi-dimensional environment description obtained by DenseSLAM can be used to improve the vehicle navigation process. Two particular applications are shown. (i) Complex landmarks can be extracted and incorporated as they

become identified using the dense representation. (ii) The dense maps can be used to estimate the variations in time of the areas explored by the robot which can be used to discriminate whether a region has potential dynamic objects.

High Level Landmarks (HLLs): One of the main problems in SLAM algorithms is the error accumulation due to non-linearities in the system. This error accumulation can be reduced if more information is added into the localisation map, since the vehicle error will remain smaller. Among the reasons to avoid including more landmarks is the computational burden required to maintain the map. However, in many situations, even when the computational cost may not be a problem, the difficulties of finding *stable and easily detectable* features cause the algorithm to use only a small number of landmarks for the localisation process, which results in a major accumulation of errors due to non-linearities.

DenseSLAM yields a rich environment representation, which gives the possibility of adding landmarks extracted from the dense maps into the landmarks map. In many situations an object cannot be detected using the measurements taken from only one vantage point. This can be due to a variety of reasons: occlusion between objects, the size of the object in relation to the sensor field of view, an inappropriate feature model, or just because the nature of the sensor makes the estimation of the landmark location impossible from only one vantage point (e.g. wide-beam sonar; Leonard J. et al. 2002, McKerrow P. 1993). Estimating partially observable features has been an important research topic in computer vision using stereo vision and bearing only information, where the initialisation of the feature position is a significant problem. The problem of partially observable features has also been studied for localisation and SLAM applications. In Leonard et al. (2002) an approach is presented that delays the decision to incorporate the observations as map landmarks. Consistent estimation is achieved by adding the past vehicle positions to the state vector and combining the observations from multiple points of view until there is enough information to validate a feature. In McKerrow (1993), intersection of constant depth range of ultrasonic sensors is used to determine the location of features from multiple vantage points.

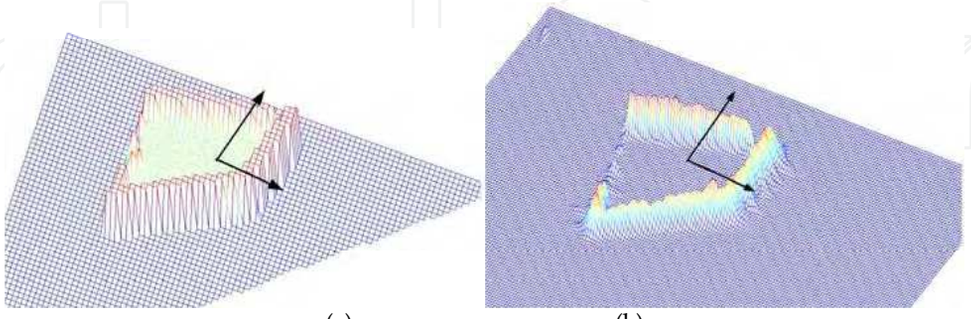
Having a comprehensive representation of the environment will enable a delayed processing to determine whether part of the map can qualify as a landmark. The rich representation obtained by DenseSLAM will enable postprocessing capabilities to continuously detect high-level landmarks using the dense map layers. The newly detected landmarks can then be added to the feature map. This approach has the potential of incorporating a large number of landmark models, some of them to be applied online at the time the observations are taken and the rest to run in the background when computer resources become available. The landmarks can then be incorporated into the features map.

High Level Landmarks representation. The only condition for the incorporation of a HLL is to represent the information in the same form as the feature map. For example, if EKF-SLAM is used, the HLLs have to be represented in state vector form.

The HLLs could be represented using geometric parameters. Experimental results of SLAM using trees as landmarks are presented in (Guivant et al., 2002). An EKF is run which estimates the trees' parameters, which consist of the centre and the diameter of the trees' trunks. In Thrun (2001) an algorithm is presented that employs expectation maximization to fit a low-complexity planar model to 3D data collected by range finders and a panoramic camera. After this model is obtained, its parameters could be added to the state vector to represent a HLL.

Fig. 6 shows an example of HLLs. In the example, the HLLs are represented as a local coordinate system and a template which is defined relative to the local axes. The templates

are formed with the information extracted from the dense maps. Scan correlation can be used to generate observations of the landmarks (see Nieto et al. 2005, for more details).



(a) (b)

Fig. 6: The figure examples of HLLs extracted from the dense maps. (a) shows a HLL represented extracted from an occupancy grid map and (b) a HLL extracted from a Sum of Gaussian dense map.

Dynamic Environments: Most of the mapping algorithms assume the world is static (Thrun, 2002). Dynamic environments require an extension of the typical representation used for static environments. That extension should allow for modelling the temporal evolution of the environment. Dynamic objects can induce serious errors in the robot localisation process. Only a few approaches that include moving objects have been presented so far. The next paragraphs review some of them.

A SLAM algorithm with generic objects (static and dynamic) is presented in (Chieh-Chih Wang, 2004). Similar to classic SLAM, the approach calculates the joint posterior over the robot and object's pose, but unlike traditional SLAM it includes also the object's motion model. The problem is shown to be computationally intractable and so a simplified version called *SLAM with Detection and Tracking of Moving Objects* (SLAM with DATMO) is presented (Chieh-Chih Wang et al., 2003). The latest algorithm decomposes the estimation process into two separate problems: (i) the SLAM problem, using static landmarks as the classic approach, and (ii) the detection and tracking of moving objects, using the robot pose estimated by the SLAM algorithm. This simplification makes updating both the SLAM and the tracking algorithm possible in real-time since they are now considered two independent filters.

In Hahnel et al. (2003) an algorithm for mapping in dynamic environments is presented. The aim of the approach is to determine which measurements correspond to dynamic objects and then filter them out for the mapping process. The approach uses the EM algorithm; the expectation step computes an estimate of which measurements might correspond to static objects. These estimates are then used in the maximization step to determine the position of the robot and the map.

An approach called *Robot Object Mapping Algorithm* (ROMA) is presented in Biswas et al. (2003). The main goal is to identify non-stationary objects and model their time varying locations. The approach assumes that objects move sufficiently slowly that they can safely be assumed static for the time it takes to build an occupancy grid map of the whole area explored by the robot. Assuming the robot is able to acquire static occupancy grid maps at different times, changes in the environment are detected using a differencing technique. The algorithm learns models of the

objects using EM. The expectation step calculates the correspondences between objects at different points in time and the maximisation step uses these correspondences to generate refined object models, represented by occupancy grid maps.

The algorithms presented in Chieh-Chih Wang et al. (2003) and Montemerlo et al. (2002) have one thing in common; they rely on pre-defined models of the specific objects they aim to track. ROMA, however, is able to learn about the shape of the objects, but the algorithm presents a number of limitations. Objects have to move slowly (it is not able to cope with fast-moving objects such as people), it is assumed the robot is able to obtain static maps at different times. The results presented include only four different objects in an environment where these objects can be perfectly segmented from a laser scan. The extension to a real environment with a larger number of objects may not be possible and will be computationally very expensive.

If navigation is the primary objective, the accurate shape of objects, or even their classification may not be important in general. What may be more useful is an algorithm able to identify observations that may be coming from objects that are not static and eliminate them from the list of observations to be used for the navigation process. Furthermore the algorithm could identify areas where it is more likely to find dynamic objects (e.g. a corridor where people walk) and then avoid their use or give a low priority to observations coming from objects in those areas.

The rich environment representation obtained by DenseSLAM allows a map layer identifying the most likely areas to possess dynamic objects to be built. As shown in Biswas et al., (2003), dynamic objects can be identified by differentiation of maps taken at different times. There are two different classes of object motions in a dynamic environment; slow motion, as for example the motion of a bin, which will be static during most of the day but will eventually be moved; and fast motion, such as people. Using DenseSLAM, and applying a straightforward differentiation, it is possible to identify regions with dynamic objects for either fast or slow motion objects.

One of the main problems with the differentiation is that maps obtained at different times will have different uncertainty. If a global map is maintained and two maps acquired at different times want to be differentiated, the uncertainty in the maps will make the matching process very difficult.

In DenseSLAM the global map is divided into smaller regions, so the whole map can be differentiated by applying differentiation between the corresponding local regions. As a consequence, the differentiation process will be prone only to local errors (which were shown to be much smaller than the global ones) eliminating detection errors due to the uncertainty between maps acquired at different moments.

A particular case where the DenseSLAM representation will not present advantages over other approaches is in decentralised multi-robot mapping. If multiple robots are used to map an area, each robot will form the map using different local regions. If the objective is to detect dynamic objects by fusing the maps built by different robots, a global map will have to be used and DenseSLAM loses advantages with respect to other approaches.

Fast motion can be captured by differentiation, in a similar way to slow motion. The main difference is that the differentiation is done over shorter periods of time and only in the region under the sensor view. As in the detection of objects with slow motion, using

DenseSLAM the differentiation is done using local regions instead of the global map. The motion detection will be included in a map layer as will the other properties captured by the sensors, then the global position of the dynamic map will be updated together with the other map properties (colour, occupancy, etc.).
It is important to note that fast motion detection can be also done with other techniques that are able to create a dense map in proximity to the vehicle position. The advantage of DenseSLAM is that the dense representation is already obtained, therefore, the detection of moving objects is a straightforward procedure that does not add computational cost.

2.3 Experimental Results

This section presents experimental results of DenseSLAM in an outdoor environment. The environment is a large area of 120 by 200 metres and the run is approximately 1 km long. The experimental platform used for the experiments is a conventional Holden UTE equipped with Sick lasers, a linear variable differential transformer sensor for the steering mechanism, back wheel velocity encoder, inertial unit and GPS.
In order to test the DenseSLAM algorithm, the GPS information was fused with the feature-based SLAM to obtain a laser image of the environment that is used as a reference to compare with the estimates by DenseSLAM. Fig. 7 shows the laser image obtained with the GPS information and the final map obtained with DenseSLAM. The dense map was obtained by fusing the raw laser observations into the local regions. The figure also shows the landmark positions.

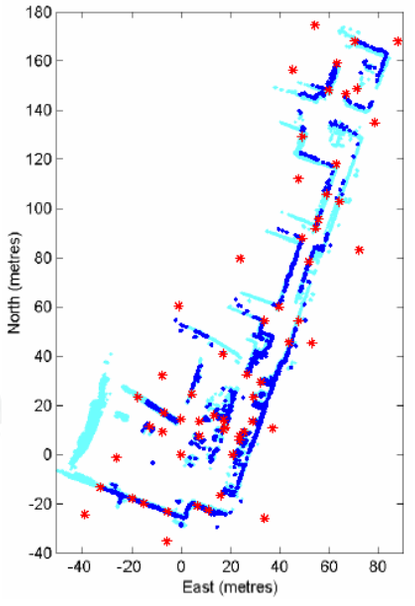


Fig. 7: Final map obtained with DenseSLAM. The light points represent the laser image obtained using GPS and SLAM. The dark points depict the dense map estimated by DenseSLAM.

Fig. 8 shows a zoom of the top part of the run. Fig. 8 (a) shows the result obtained by DenseSLAM before closing the loop. The figure also shows the laser image used as a reference. The error in the estimated map before closing the loop can be easily observed. Fig. 8 (b) shows the result after closing the first loop. Although there is still some residual error, it is clear how the estimated map has been corrected. Looking at Fig. 8 (b) it can be seen that there is some remaining uncertainty in these landmarks even after the loop is closed. This is because the vehicle does not return to the top part of the run after closing the first loop. As a result, the error in that region is not reduced as much as in the bottom part of the run. Nevertheless, an important correction in all the regions has still been made.

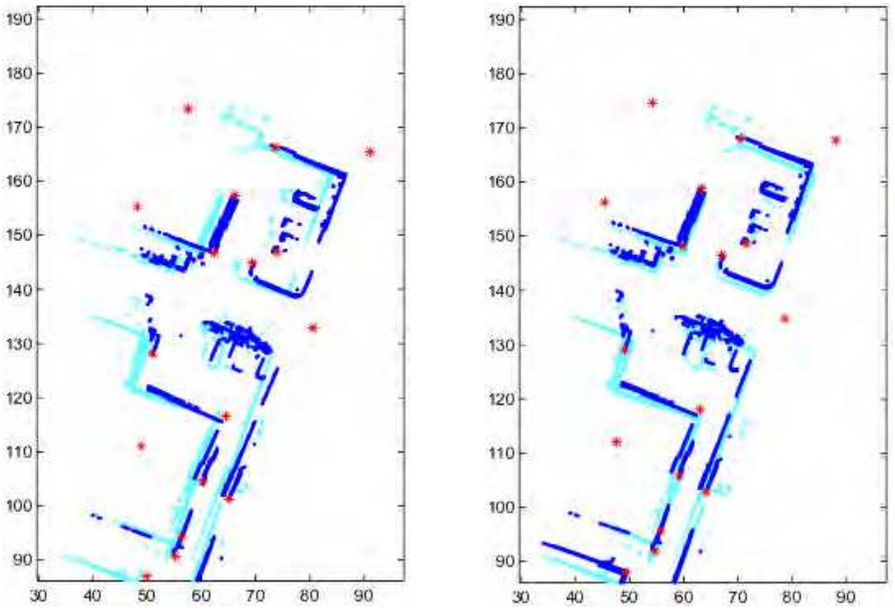


Fig. 8: The lighter points represent the laser image obtained using GPS/SLAM. The darker points represent the final map obtained with DenseSLAM. Figure (a) shows the result before closing the loop, and (b) after the loop is closed.

3. Fundamental issues when working in large areas

Most EKF implementations generate state estimations with mono-modal probability distributions and are not capable of handling multi-modal probability distributions. Multi-modal distributions are typical when closing large loops, that is, revisiting known places after a large exploration period. It is at this stage where the standard SLAM based on Kalman filters is especially fragile to incorrect association of landmarks (Neira & Tardós, 2001). Other data fusion algorithms, such as the ones that use particle filter (Montemerlo et al., 2002), can address this problem since they naturally deal with multi-hypothesis problems.

In Masson (2003) is proposed a robust data fusion algorithm, which uses a hybrid architecture. The algorithm uses Compressed EKF (CEKF in Guivant & Nebot, 2003) under normal conditions to perform SLAM. At a certain time the system may not be able to perform the association task due to large errors in vehicle pose estimation. This is an

indication that the filter cannot continue working assuming a mono-modal probability density distribution. At this time, we have the CEKF estimated mean and deviation of the states representing the vehicle pose and landmark positions. With the currently estimated map, a decorrelated map is built using a coordinate transform and a decorrelation procedure (Guivant & Nebot, 2002). A particle filter (Gordon et al., 1993) is initialised using the available statistics and is then used to resolve the position of the vehicle as a localisation problem. Once the multi-hypothesis problem is solved, the CEKF is restarted with the states values back propagated to the time when the data association problem was detected. Then the CEKF resumes operation until a new potential data association problem is detected.

There are several important implementation issues that need to be taken into account to maximise the performance of the hybrid architecture proposed. The solutions they need to consider are the uncertainties in vehicle, map and sensor to maximise the number of particles in the most likely position of the vehicle.

The SLAM algorithm builds a map while the vehicle explores a new area. The map states will be, in most cases, highly correlated in a local area. In order to use the particle filter to solve the localisation problem, a two dimensional map probability density distribution needs to be synthesised from an originally strongly correlated n dimension map. The decorrelation procedure is implemented in two steps. The map, originally represented in global coordinates is now represented in a local frame defined by the states of two beacons that are highly correlated to all the local landmarks. The other local landmarks are then referenced to this new base. A conservative bound matrix can be obtained as a diagonal matrix with bigger diagonal components and deleting the cross-correlation terms (Guivant & Nebot, 2002).

In most practical cases the local map is very large when compared to the sensor field of view. Most of the landmarks are usually beyond the range of the sensor. It is then possible to select only the *visible beacons* from the entire map by considering the estimated uncertainties. This will significantly reduce the computation complexity for the evaluation of the likelihood for each predicted particle. The boundaries of the reduced map are fixed based on the beacons that are close to the vehicle location, the particle positions, the observation and their respective uncertainty. Only a few beacons are within the field of view of any of the particles. The other beacons are not considered to be part of the reduced map.

As the number of particles affects both the computational requirements and the convergence of the algorithm, it is necessary to select an appropriate set of particles to represent the a priori density function at time T_0, that is, the time when the data association fails. Since the particle filters work with samples of a distribution rather than its analytic expression it is possible to select the samples based on the most probable initial pose of the rover. A good initial distribution is a set of particles that is dense in at least a small sub-region that contains the true states value. The initial distribution should be based in the position and standard deviations reported by the CEKF, and in at least one observation in a sub-region that contains this true state's value. In Lenser & Veloso (2000) a localisation approach is presented that replaces particles with low probability with others based on the observations. Although this algorithm is very efficient it considers that the identity of that landmark is given (known data association). This is true in some applications such as the one addressed in this work but not common in natural outdoor environments where landmarks have similar aspects and the presence of spurious objects or new landmarks is common. Here, the data association is implicitly done by the localisation algorithm.

The multi-hypotheses considered are defined by the uncertainty of the robot pose estimation. In addition the method presented is able to deal with false observations. Spurious observations and landmarks that do not belong to the map are naturally rejected by the localiser. The technique presented considers the information from a set of observations to select particles only in the initial distribution and combined with the CEKF estimates as was mentioned previously. In fact, this localisation filter is a Monte Carlo Localisation.

The initial distribution is created from range/bearing observations of a set of landmarks. This probability distribution is dominant in a region that presents a shape similar to a set of helical cylinders in the space (x, y, φ). Each helix centre corresponds to a hypothetical landmark position with its radio defined by the range observation. The landmarks considered are only the ones that the vehicle can *see* from the location reported by the CEKF and within the range and field of view of the sensors.

Although it is recognised that some observations will not be due to landmarks, all range and bearing observations in a single scan are used to build the initial distribution. Even though a set of families of helices will introduce more particles than a single family of helices (one observation), it will be more robust in the presence of spurious observations. By considering that the range/bearing observations are perfect then the dominant region becomes a discontinuous one dimensional curve (family of helices) *C*, in the three dimensional space (x, y, φ)

$$C = \bigcup_{i=1}^{N} C_i = \bigcup_{i=1}^{N} \left\{ (x, y, \varphi) \setminus \begin{array}{c} x = x(\tau) = x_i + z_r \cdot \cos(\tau) \\ y = y(\tau) = y_i + z_r \cdot \sin(\tau) \\ \varphi = \varphi(\tau) = \tau - z_\beta - \dfrac{\pi}{2} \\ \tau \in [0, 2\pi) \end{array} \right\} \tag{6}$$

These regions can be reduced by adjusting the variation of τ according to the uncertainty in φ. Assuming the presence of noise in the observations and in the landmark positions

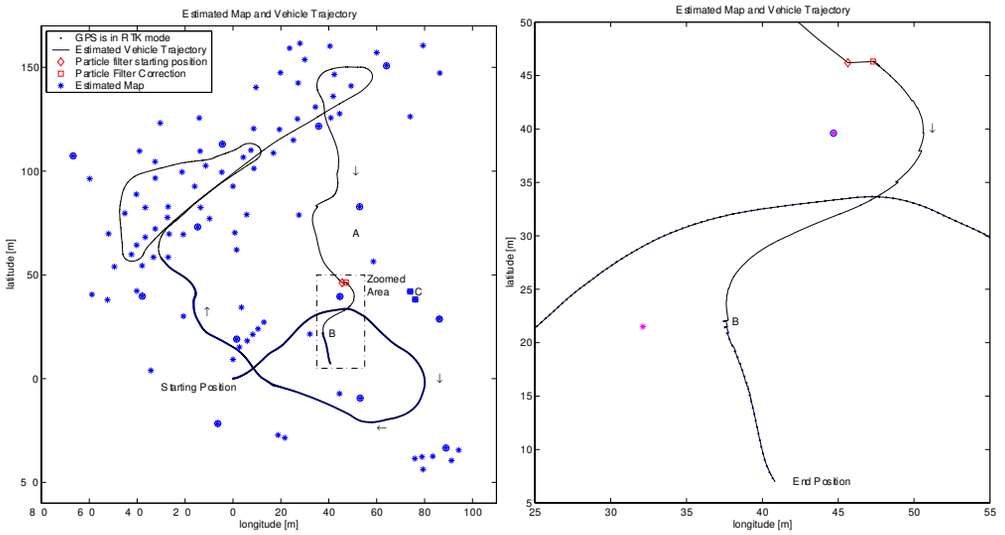
$$\begin{array}{c} z_r = z_r^* + \gamma_r, \;\; z_\beta = z_\beta^* + \gamma_\beta \\ x_i = x_i^* + \gamma_{x_i}, \;\; y_i = y_i^* + \gamma_{y_i} \end{array} \tag{7}$$

this family of helices becomes a family of cylindrical regions surrounding the helices. The helical cylinder section can be adjusted by evaluating its sensitivity to the noise sources $\gamma_{x_i}, \gamma_{y_i}, \gamma_r, \gamma_\beta$.

The same assumptions can be made for the case of using bearing only observations. Although this method can be more efficient than the standard uniform o Gaussian distribution it is still very demanding in the number of particles. A more efficient algorithm can be designed considering two observations at a time. With no data association a pair of observations will generate a family of curved cylinders to cover all possible hypotheses. This initialisation is significantly less expensive than a uniform distributed sample in a large rectangular region in the (x, y, φ) space or even a Gaussian distribution in this region. In the case of range only observations, the initialisation is very similar to the range and bearing problem. In this case the main difference is in the evaluation of the orientation (Masson et al., 2003).

Finally, two main issues need to be addressed to implement the switching strategy between the CEKF and the SIR filter. The first problem involves the detection of a potential data association failure while running the CEKF. This is implemented by monitoring the estimated error in vehicle and local map states and the results of the standard data association process. The second issue is the

reliable determination that the particle filter has resolved the multi-hypothesis problem and is ready to send the correct position to the CEKF back propagating its results. This problem is addressed by analysing the evolution of the estimated standard deviations. The filter is assumed to converge when the estimated standard deviation error becomes less than two times the noise in the propagation error model for x, y and φ. The convergence of the filter is guaranteed by the fact that the weights are bounded (Masson et al., 2003) above at any instant of time (Crisan & Doucet, 2000).
The following are results obtained using the hybrid architecture in an outdoor environment populated by trees that are used as the most relevant features to build a navigation map (Guivant et al., 2002). Full details of the vehicle and sensor model used for this experiment are available in Nebot (2002).
The CEKF filter is used to navigate when no potential data association faults are detected. When a data association failure is detected the particle filter is initialised according to the procedure presented in section 4.2 and is run until convergence is reached. At this point the filter reports the corrections to the CEKF that continues the SLAM process using EKF based methods.

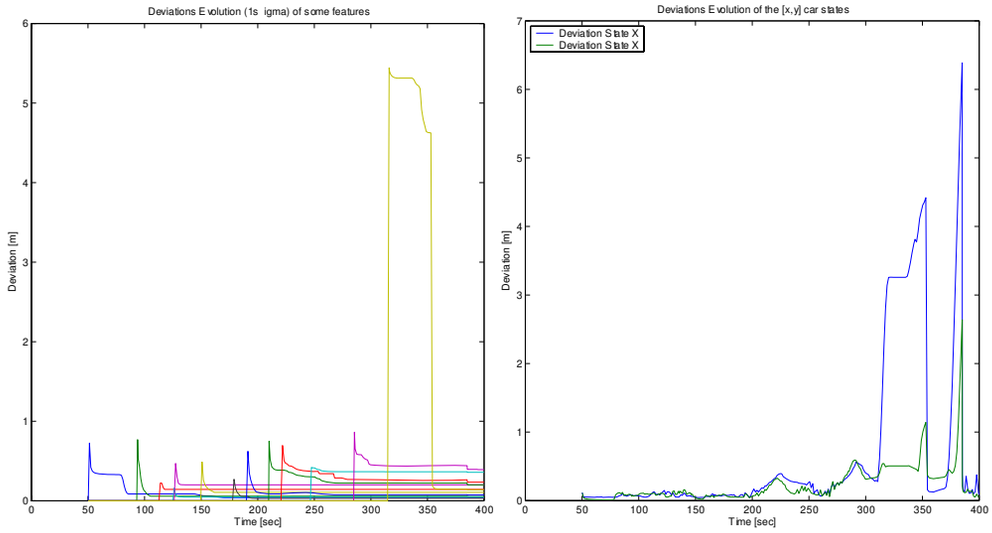


(a) (b)

Fig. 9: (a) Experimental run implementing SLAM using all the available information. (b) A zoomed area. A diamond and a square show the start and end position respectively of the particle filter correction. The dots represent the RTK GPS information.

The algorithms were tested in an environment with areas of different feature density as shown in Fig. 9. In this experiment we logged GPS, laser and dead reckoning information. The GPS used is capable of providing position information with 2 cm accuracy. This accuracy is only available in open areas and is shown in Fig. 9 with a thick line. The vehicle started at the point labelled "Starting Position" and the filter used GPS, laser and dead reckoning to perform SLAM (Guivant et al., 2002) until it reached the location at coordinates (-30,60) where GPS is no longer available. The SLAM remained operating using Laser and dead-reckoning information only. High accuracy GPS was again available close to the end of the run and will be essential to demonstrate the consistency and performance of the hybrid navigation architecture proposed.

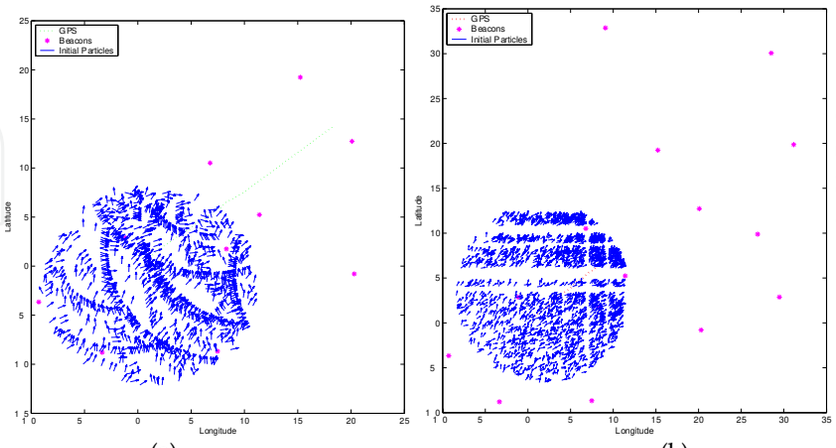
The stars and encircled stars in Fig. 9 (a) represent the natural features incorporated into the map and the selected landmarks whose deviations are shown in Fig. 10(a) respectively. A diamond and a square represent the starting and ending position resulting from the particle filter correction and are clearly shown in Fig. 9 (b). The beacons that produce the association failure are the squared stars marked as C in the figure.
Fig. 10(b) presents the vehicle position estimated error. It can be seen that the error was very small when the system was operating with GPS, *time* < *200ms*. It is then maintained below 0.5 m while in the area with high feature density. The error then started to increase before reaching point "A" since the laser cannot detect any known feature. At this time (320 sec) a new feature was incorporated but with large uncertainty as shown in Fig. 10(a). Then a known landmark was detected and since it can be associated correctly, the error in vehicle and landmark position dramatically decreased as expected.



(a) (b)

Fig. 10: Standard deviation (a) of selected beacons in the map and (b) of the car positions over time. These beacons are shown as rounded stars in Fig. 9.

A different situation is presented in Fig. 9 (b) that corresponds to the area marked as zoomed area in Fig. 9 (a). Once the laser stopped seeing the previous known landmarks the error built up again to the point where the system can no longer associate the detected landmarks to a single known landmark. The location of the vehicle at this time is represented as a diamond at coordinates (45,45) in this figure. In this case the system has to activate the Monte Carlo localiser to generate the relocalisation results shown as a square at coordinates (47,45) in the same figure.
Examples of the Monte Carlo filter initialisation are shown in Fig. 11. Fig. 11(a) shows the initialisation for the range and bearing case. The figure clearly shows the helical shape of the initial distributions. The arrows represent the position and orientation of the vehicle and the stars the beacons present in the map. The initialisation for the case of bearing only is also shown in Fig. 11(b).



(a) (b)

Fig. 11. Initialisation of the particle filter (a) using range and bearing information and (b) using bearing only information

The relocalisation result is then reported to the CEKF to continue with the SLAM process for the rest of the run. At the end of the trajectory high accuracy GPS was again available (thick line). It can be clearly seen, specially in Fig. 9 (b), that the estimated vehicle pose just before GPS became available is very close to the high accuracy GPS position reported. This demonstrates the performance and consistency of the hybrid architecture proposed.

3.1 Assimilation of non-Gaussian observations

A pure SLAM algorithm is based in measures relative to the vehicle. Nevertheless a practical application of localisation must fuse all the available sources of information that are available, included absolute information. This is a fundamental issue in navigation. Although many pure SLAM algorithms can work in large areas they could also benefit from absolute position information such as GPS. In many applications, it is not possible to obtain GPS information for long periods of time. However, at some locations this sensor will be able to report navigation data with an estimated error. It is clearly important to be able to incorporate this information to improve the localisation estimates and at the same time enable the SLAM algorithm to explore and incorporate new features while bounding the absolute pose error with the absolute information.
In order to add this information in a consistent manner some important issues need to be considered. The quality of the models and the relative navigation information used in SLAM algorithms could lead to very large innovations errors when the absolute information is fused. This occurs after long periods of navigation when only relative information is used (pure SLAM). A strong correction will make the linearisation of the models not valid generating incorrect update of covariance. The innovations may not be large but can generate strong updates in the covariance matrix. This can potentially introduce serious numerical errors. In order to prevent these problems, it is possible to treat new absolute information as L observations such that the total information introduced becomes equivalent to a single update (Guivant et al., 2002). In this case, the filter will perform L updates with the observation value and modified noise covariance. The sequential updates generate the same results as the single update but alleviate numerical problems arising from large covariance updates.

Even so, there is another potential issue that must be considered with some sensors. A typical measurement obtained from a GPS occurs when it operates in environments where there are forest and/or buildings. In open places GPS operation is usually satisfactory but is not the case in forest or *urban canyons*. The problem arises from total unavailability of satellite signals to partial occlusion and performance degradation due to multi path effects. Others sensors such as compasses present similar behaviour in static and dynamic environments where magnetic field perturbations affect the sensor operation. However there is no doubt that both sensors can provide useful information to contribute in the localisation process. In the case of range only and bearing only sensors, one measurement generates a non-Gaussian distribution and the way to deal with it is delaying the fusion collecting several measures and recording the vehicle pose (Bailey, 2002; Sola et al., 2005).

Essentially, these kinds of sensors could introduce non-Gaussian noise and some could also introduced noise correlated in time. In the case of the GPS in autonomous mode for example, the uncertainty will be introduced as a result of many factors such as satellite availability, satellites distribution, signal reflections, multi-path, atmospheric distortion, etc. It is obvious that this cannot be modelled as Gaussian, nor white. Similarly the compass usually presents biased noise due to distortion in the magnetic field, and the change depends on time and geographical position. An unknown and changing bias that varies according to the position, orientation or time represents a difficult modelling problem.

Additional to the non-Gaussian or time correlated nature of the noise, the probability distribution of the uncertainty in the observations could be unknown or only partially known. Estimators such the EKF and also any Bayesian filters cannot deal with those measurements. The improper use of them can produce inconsistent estimations. For example, if the noise is not white and this is ignored assuming that the measurements are independent, then the estimates will be over-confident. As a conservative policy these correlated measurements could be ignored to avoid inconsistent results. However in many practical applications those measurements are crucial sources of information and should be considered in a consistent way.

Consider the following situation. At time k there exists a Gaussian estimation and an available observation. This one is neither Gaussian, nor white and with partially known probability distribution, or any of these situations.

Initially it is assumed that the observation involves only one state variable and that all its probability is concentrated in an interval $a \leq x \leq b$. The shape of the probability distribution inside that interval is completely unknown and subsequent measurements are not independent, i.e. statistical dependence exists between k and $k+1$. However even under that undesirable condition it is possible to extract information from such observations. The effect of these observations will improve the full estimates state vector and will reduce the covariance matrix. In fact, a new Gaussian probability distribution is obtained. The rest of this section explains how to obtain a conservative and consistent update.

The summary of the proposed update process is the following. At time k the estimator produces a Gaussian estimate of the states $x = \{x, y, \varphi, \mathbf{m}\}$ in the form of a joint probability distribution $p_{\mathbf{x},k+1}(\mathbf{x})$, where $\{x, y, \varphi\}$ is the pose of the vehicle and $\{\mathbf{m}\}$ are the states of the landmarks' positions. A bearing observation of φ is performed and with it a marginal probability $p_{\varphi,k+1}(\varphi)$ is obtained. With the update of the marginal probability of the observed state, a total update of the joint probability $p_{\mathbf{x},k+1}(\mathbf{x})$ is obtained.

With the non-Gaussian, non-white and partially known probability observation, a new couple $(\hat{\varphi}, \sigma_\varphi)$ is estimated. This pair completely defines the marginal Gaussian density

$$p_{\varphi,k+1}(\varphi) = \frac{1}{\sqrt{2\pi}\sigma_\varphi} e^{-\frac{(\varphi-\hat{\varphi})^2}{\sigma_\varphi^2}}, \qquad \hat{\varphi} = E\{\varphi\}, \quad \sigma_\varphi = E\{(\varphi-\hat{\varphi})^2\} \tag{8}$$

The non zero cross-correlation terms in the covariance matrix means that all the states are connected. Then, with this new couple $(\hat{\varphi}, \sigma_\varphi)$ it is necessary to carry out a virtual update with the purpose of transmitting the new information acquired to the whole density $p_{\mathbf{x},k+1}(\mathbf{x})$ whose expression is

$$p_{\mathbf{x},k+1}(\mathbf{x}) = \frac{1}{\sqrt{2\pi \det(\mathbf{P})}} e^{-(\varphi-\hat{\mathbf{x}})^T \mathbf{P}(\varphi-\hat{\mathbf{x}})}, \qquad \hat{\mathbf{x}} = E\{\mathbf{x}\}, \qquad \mathbf{P} = E\{(\mathbf{x}-\hat{\mathbf{x}})(\mathbf{x}-\hat{\mathbf{x}})^T\} \tag{9}$$

As a result of this update a new joint Gaussian density is obtained, and the normal estimation process is pursued.

In general (Guivant & Masson, 2005), for an arbitrary density $p(\varphi)$ that concentrates all its energy inside the interval (a, b), a Gaussian density with expected value b is a better approximation to $p(\varphi)$ than any other Gaussian density with expected value greater than b if the better previous estimation obtained is greater than b. In particular, this is better than discarding the observation. The same happens with Gaussian densities whose expected value is smaller than a and it is independent of the form that take $p(\varphi)$ inside the interval (a, b). Consequently, the mean ξ of the new density it is selected as

$$if \quad b < c \qquad \Rightarrow \quad \xi = b \tag{10}$$

$$if \quad a > c \qquad \Rightarrow \quad \xi = a \tag{11}$$

$$if \quad a > c > b \qquad \Rightarrow \quad \xi = c \tag{12}$$

where c is the mean of the better previous estimate. The deviation of this new Gaussian must be obtained by solving the following implicit equation

$$if\ c \notin (a,b), \qquad \ln\left(\frac{\sigma_\xi}{\sigma_c}\right) + \frac{-(a-c)^2}{\sigma_c^2} + \frac{(a-b)^2}{\sigma_\xi^2} = 0 \tag{13}$$

$$if\ c \in (a,b), |a-c| > |b-c| \qquad \ln\left(\frac{\sigma_\xi}{\sigma_c}\right) + (a-c)^2\left(-\frac{1}{\sigma_c^2} + \frac{1}{\sigma_\xi^2}\right) = 0 \tag{14}$$

$$if\ c \in (a,b), |a-c| \le |b-c| \qquad \ln\left(\frac{\sigma_\xi}{\sigma_c}\right) + (b-c)^2\left(-\frac{1}{\sigma_c^2} + \frac{1}{\sigma_\xi^2}\right) = 0 \tag{15}$$

Then, unless the mean is updated, the deviation is always improved. This is an important result because it is always necessary to maintain the absolute error between the true value and the mean of the Gaussian bounded. This condition guarantees a secure condition for the EKF as estimator. If the mean value estimated is near the true value the filter will perform almost as a linear estimator. In particular, the Jacobians will be calculated properly. In several cases, the filter could behave in a consistent way. But, given great deviations, the Jacobians evaluated at the mean value will be different from the one calculated at the true value. This fact is widely known in the EKF estimation theory.

At this point the calculation was focused on the marginal density $p(\varphi)$. However the full probability density is a Gaussian multi-dimensional density. The covariance matrix is a full matrix and this shows the correlation between the states of the vehicle and the map. It was shown (Gibbens et al., 2000) that neglecting this correlations leads to non-consistent estimations.

A virtual update is a form to update the full covariance matrix. The desired update over the individual deviation σ_φ is known. With it, it is possible to obtain the complete update without violating conditions of consistency of the estimation. The updated covariance will be

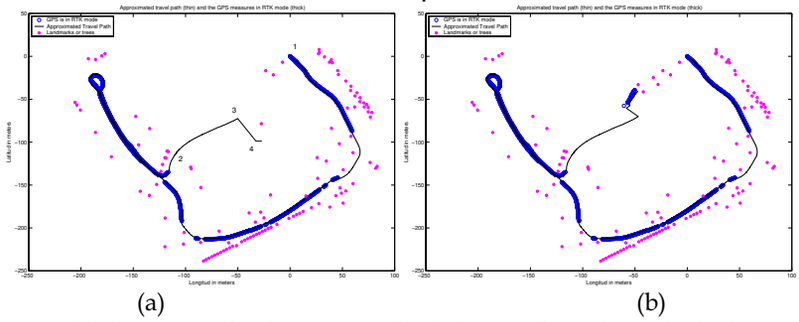
$$\mathbf{P}_{k+1|\,k+1} = \mathbf{P}_{k+1|\,k} - \Delta\mathbf{P} \tag{16}$$

There,

$$\Delta \mathbf{P} = \mathbf{P}_{k+1|k}(:, i_\varphi) \frac{\Delta \sigma_\varphi^2}{\sigma_{\varphi,k}^4} \mathbf{P}_{k+1|k}(i_\varphi, :) \tag{17}$$

where $\mathbf{P}_{k+1|\,k}(:, i_\varphi)$ is the row vector i_φ of the predicted covariance matrix, $\mathbf{P}_{k+1|\,k}(i_\varphi, :)$ is the column vector i_φ, $\Delta \sigma_\varphi^2$ is the improvement in the deviation incorporating the non Gaussian observation and $\sigma_{\varphi,k}$ is the deviation predicted in the state φ.

Fig. 12 shows the proposed approach when it is applied in a SLAM process where non-Gaussian observations come from compass measurements. Details about the vehicle model and the SLAM algorithm could be referred from (Guivant et al., 2002). In this experiment GPS, laser, compass and dead reckoning information was available. The GPS used is capable of providing position information with 2 cm of accuracy when it works in RTK mode. This quality is only available in relatively open areas and is shown in Fig. 12 by using a thick line. The vehicle started at the point labelled 1. An EKF performs SLAM by using all the available information (thin line). When the vehicle arrives at point 2, there is no GPS information and the laser and compass are intentionally disconnected until the vehicle reaches point 3. The reason for this is to allow the uncertainty to grow and clearly show the impact of the algorithm. In Fig. 12 (a), at point 4, it could be seen how the estimator goes far away from the real path that can be seen in Fig. 12 (b). In this last case, the filter uses the non-Gaussian observation of the compass to correct the mean and covariance.



(a) (b)

Fig. 12. Figure (a) shows results from a standard SLAM algorithm which does not use the available compass measurements. At point 4 the data association fails. Figure (b) shows the result from a SLAM process which does use the available compass measurements and at point 4 the data association is successful.

4. Conclusion

A solution to the SLAM problem is necessary to make a robot truly autonomous. For this reason, SLAM has been one of the main research topics in robotics, especially during the last fifteen years. While the structure of the problem is today well known, there are still many open problems, particularly when working in outdoor environments. We presented here some of the latest SLAM algorithms that address the problem of localisation and mapping in large outdoor areas.

5. References

Bailey T. (2002). Mobile Robot Localisation and Mapping in Extensive Outdoor Environments. PhD thesis, University of Sydney, Australian Centre for Field Robotics, 2002.

Durrant-Whyte, H. & Bailey T. (2006). Simultaneous localization and mapping: part I. *IEEE Robotics & Automation Magazine*, Vol. 13, No. 2, June 2006, 99 – 108, ISSN 1070-9932

Biswas R., Limketkai B., Sanner S. & Thrun S. (2003). Towards object mapping in non-stationary environments with mobile robots. *Proceedings of the 2003 IEEE International Conference on Robotics and Automation, ICRA 2003*, pp. 1014-1019, ISBN 0-7803-7736-2, Taipei, Taiwan, September 2003, IEEE.

Chieh-Chih Wang, Thorpe C. & Thrun S. (2003). Online simultaneous localization and mapping with detection and tracking of moving objects: Theory and results from a ground vehicle in crowded urban areas. *Proceedings of the 2003 IEEE International Conference on Robotics and Automation, ICRA 2003*, pp. 842-849, ISBN 0-7803-7736-2, Taipei, Taiwan, September 2003, IEEE.

Chieh-Chih Wang (2004). Simultaneous Localization, Mapping and Moving Object Tracking. *PhD thesis*, Robotics Institute, Carnegie Mellon University, 2004

Crisan D. & Doucet A. (2000). Convergence of sequential monte carlo methods. *Technical Report Cambridge University*, CUED/FINFENG /TR381, 2000.

Elfes A.(1989). Occupancy Grids: A probabilistic framework for robot perception and navigation. *Phd thesis, Department of Electrical Engineering*, Carnegie Mellon University, 1989

Gibbens P. W., Dissanayake G. M. W. M. & Durrant-Whyte H. F. (2000). A closed form solution to the single degree of freedom simultaneous localisation and map building (SLAM) problem. *Proceedings of the 39th IEEE Conference on Decision and Control*, pp. 191-196, ISBN 0-7803-6638-7, Sydney, Australia, December 2000, IEEE.

Gordon N. J., Salmond D. J. & Smith A. F. M. (1993). Novel approach to nonlinear/non-gaussian bayesian state estimation. *IEE Proceedings-F Radar and Signal Processing*, Vol. 140, No. 2, April 1993, 107–113, ISSN 0956-375X

Guivant J. & Nebot E. (2002). Improving computational and memory requeriments of simultaneous localization and map building algorithms. *Proceedings of the 2002 IEEE International Conference on Robotics and Automation, ICRA 2002*, pp. 2731-2736, ISBN 0-7803-7272-7, Washington DC, May 2002, IEEE.

Guivant J., Masson F. & Nebot E. (2002). Simultaneous localization and map building using natural features and absolute information. *Robotics and Autonomous Systems*, Vol. 40, No. 2-3, August 2002, 79–90, ISSN 0921-8890.

Guivant J. & Nebot E. (2003). Solving computational and memory requirements of feature-based simultaneous localization and mapping algorithms. *IEEE Transaction on Robotics and Automation*, Vol. 19, No. 4, August 2003, 749 - 755, ISSN 1042-296X.

Guivant J., Nieto J., Masson F. & Nebot E. (2004). Navigation and mapping in large unstructured environments. *The International Journal of Robotics Research*, Vol. 23, No. 4, April 2004, 449- 472, ISSN 0278-3649.

Guivant, J. & Masson, F. (2005). Using Absolute Non-Gaussian Non-White Observations in Gaussian SLAM. *Proceedings of the 2005 IEEE International Conference on Robotics and Automation, ICRA 2005*, pp. 336 – 341, ISBN 0-7803-8914-X/05, Barcelona, Spain, April 2005, IEEE.

Hahnel D., Triebel R., Burgard W. & Thrun S. (2003). Map building with mobile robots in dynamic environments. *Proceedings of the 2003 IEEE International Conference on Robotics and Automation, ICRA 2003*, pp. 1557-1563, ISBN 0-7803-7736-2, Taipei, Taiwan, September 2003, IEEE.

Lacroix S., Mallet A., Bonnafous D., Bauzil G., Fleury S., Herrb M., & Chatila R. (2002). Autonomous rover navigation in a unknown terrains: Functions and integrations.

The International Journal of Robotics Research, Vol. 21, No. 10- 11, October - November 2002, 917-942, ISSN 0278-3649.

Lenser S. & Veloso M. (2000). Sensor resetting localization for poorly modelled mobile robots. *Proceedings of the 2000 IEEE International Conference on Robotics and Automation, ICRA 2000*, pp. 1225–1232, ISBN 0-7803-5886-4, San Francisco, CA, April 2000, IEEE.

Leonard J. J., Rikoski R. J., Newman P. M. & Bosse M. (2002). Mapping partially observable features from multiple uncertain vantage points. *The International Journal of Robotics Research*, Vol. 21, No. 10- 11, October - November 2002, 943-975, ISSN 0278-3649.

Masson F., Guivant J. & Nebot E. (2003). Robust Navigation and Mapping Architecture for Large Environments. *Journal of Robotics Systems*, Vol. 20, No. 10, October 2003, pp. 621 – 634, ISSN 0741-2223

Masson F., Guivant J., Nieto J. & Nebot E. (2005). The hybrid metric map: a solution for precision farming. *Latin American Applied Research*, Vol. 35, No 2, April 2005, pp. 105-110, ISSN 0327-0793.

McKerrow P. J. (1993). Echolocation - from range to outline segments. *Robotics and Autonomous Systems*. Vol. 11, No. 3-4, December 1993, 205-211, ISSN 0921-8890.

Montemerlo M., Thrun S. & Whittaker W. (2002). Conditional particle filters for simultaneous mobile robot localization and people-tracking. *Proceedings of the 2002 IEEE International Conference on Robotics and Automation, ICRA 2002*, pp. 695-701, ISBN 0-7803-7272-7, Washington DC, May 2002, IEEE

Nebot E. (2002). Experimental outdoor dataset. ACFR, University of Sydney. http://www.acfr.usyd.edu.au/homepages/academic/enebot/dataset.htm

Neira J. & Tardós J.D. (2001). Data association in stochastic mapping using the joint compatibility test. *IEEE Transaction on Robotics and Automation*, Vol. 17, No. 6, December 2001, 890 - 897, ISSN 1042-296X.

Nieto J., Guivant J. & Nebot E.(2004). The hybrid metric maps (HYMMs): A novel map representation for DenseSLAM. *Proceedings of the 2004 International Conference on Robotics and Automation, ICRA 2004*, pp. 391-396, ISBN 0-7803-8232-3/04, New Orleans, LA, April 2004, IEEE.

Nieto J., Bailey T. & Nebot E. (2005). Scan-slam: Combining EKF-SLAM and scan correlation. *Proceedings of the International Conference on Field and Service Robotics*, pp. 129-140, Port Douglas, Australia, July 2005.

Solá J., Monin A., Devy M. & Lemaire T. (2005). Undelayed initialization in bearing only SLAM. *Proceedings of the 2005 IEEE/RSJ International Conference on Intelligent Robots and Systems, IROS 2005*, pp. 2751–2756, ISBN 0-7803-7736-2, Alberta, Canada, August 2005, IEEE.

Thrun S., Burgard W., Chakrabarti D., Emery R., Liu Y. & Martin C. (2001). A real-time algorithm for acquiring multi-planar volumetric models with mobile robots. *Proceedings of the 10th International Symposium of Robotics Research, ISRR'01*, pp. 21-36, ISBN 3540005501, Lorne, Australia, November 2001.

Thrun S. (2002). Robotic mapping: A survey. In *Exploring Artificial Intelligence in the New Millenium*, Lakemeyer G. & Nebel B., ,Morgan Kaufmann.

Mobile Robots: Perception & Navigation

Edited by Sascha Kolski

ISBN 3-86611-283-1

Hard cover, 704 pages

Publisher Pro Literatur Verlag, Germany / ARS, Austria

Published online 01, February, 2007

Published in print edition February, 2007

Today robots navigate autonomously in office environments as well as outdoors. They show their ability to beside mechanical and electronic barriers in building mobile platforms, perceiving the environment and deciding on how to act in a given situation are crucial problems. In this book we focused on these two areas of mobile robotics, Perception and Navigation. This book gives a wide overview over different navigation techniques describing both navigation techniques dealing with local and control aspects of navigation as well es those handling global navigation aspects of a single robot and even for a group of robots.

How to reference

In order to correctly reference this scholarly work, feel free to copy and paste the following:

Favio Masson, Juan Nieto, Jose Guivant and Eduardo Nebot (2007). Robust Autonomous Navigation and World Representation in Outdoor Environments, Mobile Robots: Perception & Navigation, Sascha Kolski (Ed.), ISBN: 3-86611-283-1, InTech, Available from: http://www.intechopen.com/books/mobile_robots_perception_navigation/robust_autonomous_navigation_and_world_representation_in_outdoor_environments

INTECH
open science | open minds

InTech Europe
University Campus STeP Ri
Slavka Krautzeka 83/A
51000 Rijeka, Croatia
Phone: +385 (51) 770 447
Fax: +385 (51) 686 166
www.intechopen.com

InTech China
Unit 405, Office Block, Hotel Equatorial Shanghai
No.65, Yan An Road (West), Shanghai, 200040, China
中国上海市延安西路65号上海国际贵都大饭店办公楼405单元
Phone: +86-21-62489820
Fax: +86-21-62489821

IntechOpen