UNIVERSIDAD DE CHILE
FACULTAD DE CIENCIAS FÍSICAS Y MATEMÁTICAS
DEPARTAMENTO DE INGENIERÍA ELÉCTRICA

THE ESTIMATION OF DETECTION STATISTICS IN SIMULTANEOUS LOCALIZATION AND MAPPING

TESIS PARA OPTAR AL GRADO DE MAGÍSTER EN CIENCIAS DE LA INGENIERÍA
MENCIÓN INGENIERÍA ELÉCTRICA
MEMORIA PARA OPTAR AL TÍTULO DE INGENIERO CIVIL ELÉCTRICO

FELIPE IGNACIO INOSTROZA FERRARI

PROFESOR GUÍA:
DR. MARTIN ADAMS

MIEMBROS DE LA COMISIÓN:
DR. MARCOS ORCHARD CONCHA
DR. MIGUEL TORRES TORRITI

SANTIAGO DE CHILE

2015

RESUMEN DE LA MEMORIA PARA OPTAR AL TÍTULO DE: Ingeniero Civil Eléctrico y grado de Magíster en Ciencias de la Ingeniría mención Ingeniería Eléctrica
POR: Felipe Ignacio Inostroza Ferrari
FECHA: 01/06/2015
PROFESOR GUIA: Martin Adams

ESTIMACIÓN DE ESTADÍSTICAS DE DETECCIÓN EN LA CONSTRUCCIÓN DE MAPAS Y LOCALIZACIÓN SIMULTÁNEA

El uso de Conjuntos Aleatorios Finitos (RFS por su sigla en inglés) tiene varias ventajas respecto de los métodos tradicionales basados en vectores. Entre ellas están el incluir las estadísticas de detección del sensor y la eliminación de las heurísticas tanto para la asociación de datos como para la inicialización y eliminación de objetos en mapa. Para obtener los beneficios de los estimadores basados en RFS en el problema de Construcción de Mapas y Localización Simultanea (SLAM por su acrónimo en inglés), las estadísticas de detección y falsa alarma del extractor de características deben ser modeladas y utilizadas en cada actualización del mapa. Esta Tesis presenta técnicas para obtener estas estadísticas en el caso de características semánticas extraídas de mediciones láser. Además se concentra en la extracción de objetos cilíndricos, como pilares, árboles y postes de luz, en ambientes exteriores. Las estadísticas de detección obtenidas son utilizadas dentro de una solución a SLAM basada en RFS, conocida como Rao-Blackwellized (RB)-probability hypothesis density (PHD)-SLAM, y el algoritmo multiple hypothesis (MH)-factored solution to SLAM (FastSLAM), solución a SLAM basada en vectores. El desempeño de cada algoritmo al usar estas estadísticas es comparado con el de utilizar estadísticas constantes. Los resultados muestran las ventajas de modelar las estadísticas de detección, particularmente en el caso del paradigma RFS. En particular, el error en las estimaciones del mapa, medido utilizando la distancia optimal subpattern assignment (OSPA) a un mapa "ground truth" generado de forma independiente, disminuye en un 13% en el caso de MH-FastSLAM y en un 13% para RB-PHD-SLAM al modelar las estadísticas de detección. A pesar de que no se tiene un "ground truth" para la trayectoria del robot, se evalúan las trayectorias visualmente, encontradose estimaciones superiores para el método propuesto. Por lo tanto, se concluye que el modelamiento de las estadísticas de detección es de gran importancia al implementar una aplicación de SLAM.

RESUMEN EN INGLÉS DE LA MEMORIA
PARA OPTAR AL TÍTULO DE: Ingeniero Civil Eléctrico y grado de Magíster en Ciencias de la Ingeniría mención Ingeniería Eléctrica
POR: Felipe Ignacio Inostroza Ferrari
FECHA: 01/06/2015
PROFESOR GUIA: Martin Adams

THE ESTIMATION OF DETECTION STATISTICS IN SIMULTANEOUS LOCALIZATION AND MAPPING

The use of Random finite sets (RFSs) in Simultaneous localization and mapping (SLAM) has many advantages over the traditional random vector based approaches. These include the incorporation of detection and clutter statistics and the circumvention of data association and map management heuristics in the estimation process. To take full advantage of RFS based estimators in feature based SLAM, the feature extractor's detection and false alarm statistics should be modelled and used in each map update. This thesis presents principled techniques to obtain these statistics for semantic features extracted from two-dimensional laser range data, and focusses on the extraction of circular cross-sectioned features, such as trees, pillars and lamp-posts, in outdoor environments. The derived detection statistics are used within an RFS based SLAM algorithm, known as Rao-Blackwellized (RB)-probability hypothesis density (PHD)-SLAM, and the state of the art multiple hypothesis (MH)-factored solution to SLAM (FastSLAM). The performance of each algorithm is compared using the derived, as well as the usually assumed constant, detection statistics. The results from real experimental data demonstrate the advantages of explicitly modelling feature detection statistics, particularly in the RFS framework. Specifically, the error in the map estimates, measured by the optimal sub-pattern assignment (OSPA) distance to an independently generated ground truth map, diminish by 13% in the case of MH-FastSLAM and by 13% for RB-PHD-SLAM. Although ground truth information for the robot trajectory is not available, the trajectory estimates are visually evaluated and found to be superior for the proposed method. Therefore, modelling detection statistics is found to be of importance when implementing SLAM applications.

ACKNOWLEDGEMENTS

First I would like to thank my mum María Teresa Ferrari and my dad Fernando Inostroza for making everything possible. I need to thank my advisor Martin Adams for his guidance and continuous support, including the funding for me to travel to the learn about the Random Set paradigm which is fundamental for this thesis and the tremendous effort in making my writing understandable both in this thesis and the papers that came out of it. I would also like to thank Keith Leung and Daniel Lühr for their help, collaboration and advice during the creation of this thesis and my lab mates Omar, Rodrigo, Paul and Isao for providing a a nice environment. Finally I would like to thank my friends, specially Roberto Müller, for providing other things to do before and during the creation of the thesis.

I would also like to acknowledge Conicyt, Fondecyt Project 1110579, the Advanced Mining Technology Centre (AMTC), Universidad de Chile and Clearpath Robotics Inc, Canada.

Gracias a todos.

Contents

List of Figures

Chapter 1

Introduction

Simultaneous localization and mapping (SLAM) is a problem in robotics in which a robot uses its available sensor measurements to estimate a map of the operating environment, while concurrently determining its pose relative to the map. The general probabilistic approach currently adopted by the mobile robotics community uses random vectors to represent the robot and map state, and solves SLAM through stochastic filtering, or batch estimation [1]. Recently, a different representation has been introduced for feature-based maps using random finite sets (RFSs) [2, 3], in which random vectors, typically representing the spatial location of individual landmarks, are placed in a set whose cardinality (feature number) is also a random variable.

There are several benefits in using a RFS-based filtering approach to estimate the map in SLAM, compared to a vector-based approach. Typically in vector-based approaches, the correspondence between measurements and landmarks is performed separately from the actual filter, and is determined using heuristics (e.g., by comparing the measurement to landmark likelihood with a preset threshold). These correspondences are required to determine which landmark estimate is updated by which measurement. In contrast, under an RFS SLAM framework, data association becomes a part of the landmark estimate update process for which Bayes theorem is applied. Essentially, the RFS approach updates landmark estimates by simultaneously associating them with every measurement, eliminating the need for heuristics. Another benefit of RFS-based filtering is that it accounts for detection statistics (feature probabilities of detection and false alarm). Finally, the RFS approach not only estimates the spatial position of landmarks, but also the number of landmarks that have entered the field of view (FoV) of the robot's sensors. This is because the cardinality of a RFS is also a random variable that is estimated. Therefore, to fully utilize the capabilities of RFS based filters, modelling of a feature detector's detection statistics is necessary.

In most RFS update stages, a feature's probability of detection is a state dependent quantity, which in the case of SLAM implies that it depends on the current robot's pose and

the position of that feature itself. However, principled methods that currently provide such statistics calculate them based upon the measurements themselves, such as the well known constant false alarm rate (CFAR) processors applied to radar data. Since the detection statistics required by RFS-SLAM are state dependent and there is no concept of feature to measurement association in RFS-SLAM, measurement based detection statistics, such as those provided by CFAR detectors, cannot be directly applied in RFS techniques. This thesis therefore addresses methods to estimate these statistics on a per estimated feature basis, taking into account feature occlusions.

This thesis proposes a method of obtaining the detection statistics of a laser data feature extractor, and its use in a RFS-SLAM implementation, known as probability hypothesis density (PHD)-SLAM; an application of one of the simplest mathematical finite set statistics (FISST) tools for estimation with RFSs, developed by Mullane, Vo, Adams, and Vo [2]. The contribution of this thesis is to demonstrate the implementation of PHD-SLAM with commonly used 2D scanning laser range finders, as well as the importance of modelling sensor detection statistics in a principled manner. A simple feature detection strategy is presented, in which the expected and variable probabilities of detection associated with laser range data are derived. Results of applying the laser based feature detector under a Rao-Blackwellized (RB)-PHD-SLAM framework [2] are presented and compared with results obtained from the same algorithm, with the usually assumed constant feature probabilities of detection within the sensor's FoV. Results from a state of the art SLAM solution, multiple hypothesis (MH)-factored solution to SLAM (FastSLAM), are also shown for comparison purposes, both with the derived detection statistics and with constant probabilities of detection.

Chapter 2

Literature Review

This chapter presents the state of the art in SLAM methods and shows some of the work that has been carried out, which includes detection statistics within state estimation framework. It also briefly presents the concepts of random finite set (RFS).

2.1 Simultaneous Localization and Mapping

The SLAM problem is one of the fundamental problems in mobile robotics, and has been considered by some to be the "Holy Grail" of mobile robotics [4]. A robot equipped with exteroceptive sensors (and optionally proprioceptive as well) is tasked with estimating a map of its environment as it transverses it, while concurrently estimating its pose within that map. During the last two decades remarkable progress has been made on this problem and this section reviews the SLAM problem and some of the most important advances.

2.1.1 SLAM Process Model Dynamics

SLAM is a state estimation problem, in which the best estimate of the robot trajectory and map feature positions is sought over time, using all sensor measurements. In general, the underlying stochastic system representing the robot's pose component of SLAM using non-linear discrete-time equations, is represented as:

$$\boldsymbol{x}_k = \boldsymbol{g}(\boldsymbol{x}_{k-1}, \boldsymbol{u}_{k-1}, \boldsymbol{\delta}_{k-1}) \tag{2.1}$$

$$\boldsymbol{z}_k^i = \boldsymbol{h}(\boldsymbol{x}_k, \boldsymbol{m}_k^j, \boldsymbol{\epsilon}_k) \ , \tag{2.2}$$

where:

- $\boldsymbol{x}_k$ represents the robot pose (spatial coordinates and orientation) at time-step k,

- $\boldsymbol{g}$ is the robot motion model,
- $\boldsymbol{u}_k$ is the the odometry measurement at time-step k,
- $\boldsymbol{\delta}_k$ is the process noise at time-step k,
- $\boldsymbol{z}_k^i$ is the i-th measurement vector at time-step k,
- $\boldsymbol{h}$ is the sensor-specific measurement model relating measurement and state in the spatial domain,
- $\boldsymbol{m}_k^j$ is a random vector for the position of landmark j,
- $\boldsymbol{\epsilon}_k$ is the spatial measurement noise

Traditional vector-based approaches to SLAM concatenate random state vectors that model the robot and landmarks, in a single vector that is used for the estimation process:

$$\boldsymbol{m}_k \equiv \left[\boldsymbol{m}_k^1, \boldsymbol{m}_k^2, ..., \boldsymbol{m}_k^m\right]^T \ . \tag{2.3}$$

Similarly, multiple measurements, corresponding to feature detections are also concatenated as:

$$\boldsymbol{z}_k \equiv \left[\boldsymbol{z}_k^1, \boldsymbol{z}_k^2, ..., \boldsymbol{z}_k^n\right]^T \ . \tag{2.4}$$

Furthermore, the generally complex data association problem needs to be solved so that i and j correspond to the same landmark.

Within the RFS approach, the observed landmarks up to and including time-step k, are defined as

$$\mathcal{M}_k \equiv \{\boldsymbol{m}_k^1, \boldsymbol{m}_k^2, ..., \boldsymbol{m}_k^m\} \ , \tag{2.5}$$

where the number of landmarks, $|\mathcal{M}_k| = m$, is also a random variable. In general, the landmark from which a measurement is generated is unknown. Furthermore, there is a probability of detection, P_D , associated with every landmark, implying that it may be misdetected with probability $1 - P_D$. Measurements may also be generated from sensor noise or objects of non-interest (clutter), with assumed known distributions. The set of all n measurements at time-step k is defined as

$$\mathcal{Z}_k \equiv \{\boldsymbol{z}_k^1, \boldsymbol{z}_k^2, ..., \boldsymbol{z}_k^n\} \ . \tag{2.6}$$

With these definitions a new, set-based, measurement model can be defined:

$$\mathcal{Z}_k \equiv \mathcal{H}(\boldsymbol{x}_k, \mathcal{M}_k, \boldsymbol{\epsilon}_k) \cup \mathcal{E}_k \ , \tag{2.7}$$

where $\mathcal{E}_k$ represents the clutter and $\boldsymbol{\epsilon}_k$ represents the spatial noise of the measurements. Using a probabilistic framework and a filtering approach, the probability density functions (PDFs)

$$p\left(\boldsymbol{x}_{0:k}, \mathcal{M}_k \middle| \mathcal{Z}_k, \boldsymbol{u}_{0:k}\right) \ , \tag{2.8}$$

$$p\left(\boldsymbol{x}_{0:k}, M_k \middle| Z_k, \boldsymbol{u}_{0:k}\right) \tag{2.9}$$

are sought by RFS and vector approaches respectively. In both cases the estimates are made relative to the initial robot's pose, at each time step.

2.1.2 Observability of SLAM

The first question that should be asked after formulating the SLAM problem is whether it is actually solvable. In control theory observability is defined as the ability to estimate the state of a system from its inputs and outputs. In 2001, Dissanayake, Newman, Clark, Durrant-Whyte, and Csorba [4] provided a solution to the two-dimensional linear version of the SLAM problem, with known initial pose, and proved its convergence. This implies the observability of the problem. However, the observability of the linear version of the problem cannot be applied to the real SLAM definition, which is highly non-linear. In 2006, Lee, Wijesoma, and Ibanez Guzman [5] showed that the two-dimensional non-linear version of SLAM with range bearing measurements and unknown initial robot pose is observable only if at least two landmarks have known positions. Wang and Dissanayake [6] used Fisher's linear discriminant to show that a general way to calculate the observability of different non-linear SLAM formulations. They used their method to confirm the results in [5] and show that the solution to two-dimensional SLAM, with known initial pose and a range bearing measurement model, is observable without the need for known feature positions.

It is important to note that all the results on the observability of SLAM have been obtained using the vector-based formulation of SLAM, with known feature number and data association. Therefore these results cannot be directly applied to the set-based definition of the SLAM problem.

2.1.3 Vector based SLAM

Dissanayake, Newman, Clark, Durrant-Whyte, and Csorba [4] introduced a solution based on the extended Kalman filter (EKF), EKF-SLAM, given known target number and data

association. In the case of linear measurement and robot motion models they proved that it will converge monotonically to the correct solution, for both the map estimate and robot trajectory. The complexity of this algorithm is $O(|\boldsymbol{z}|\ |\boldsymbol{m}|^2)$, where $|\boldsymbol{z}|$ and $|\boldsymbol{m}|$ are the number of measurements and number of landmarks respectively.

Thrun, Liu, Koller, Ng, Ghahramani, and Durrant-Whyte [7] introduced a solution to SLAM by modifying the dual of the EKF, called extended information filter (EIF). In EIF-SLAM instead of representing the posterior using its mean $\boldsymbol{\mu}$ and covariance Σ it is represented using the information matrix H and information vector $\boldsymbol{b}$, defined as:

$$H \equiv \Sigma^{-1} \tag{2.10}$$

$$\boldsymbol{b} \equiv \boldsymbol{\mu}^T H\ . \tag{2.11}$$

In the paper the authors showed that the solution to EIF-SLAM in this form is naturally sparse. By enforcing this sparseness the authors reduced the complexity of both the prediction and the update steps to constant time. This considers the addition of a single measurement at a time, making the complexity of the algorithm $O(|Z|)$. However, significant computation is required to recover the mean and covariance from the information matrix and vector. It was shown shown that this new sparse EIF (SEIF) is more likely to become inconsistent (i.e. overconfident) than its non sparse counterpart, but this did not seem to impact the results of the experiments.

Montemerlo, Thrun, Koller, Wegbreit, et al. [8] introduced a RB-particle filter (PF) solution to SLAM, called FastSLAM, by factoring the Bayes posterior into:

$$p\left(\boldsymbol{x}_{0:k}, M_k \middle| Z_k, \boldsymbol{u}_{0:k}\right) = p\left(\boldsymbol{x}_{0:k} \middle| Z_k, \boldsymbol{u}_{0:k}\right) p\left(M_k \middle| \boldsymbol{x}_{0:k}, Z_k, \boldsymbol{u}_{0:k}\right) \tag{2.12}$$

$$= p\left(\boldsymbol{x}_{0:k} \middle| Z_k, \boldsymbol{u}_{0:k}\right) \prod_{i=0}^{|M_k|} p\left(m_i \middle| \boldsymbol{x}_{0:k}, Z_k, \boldsymbol{u}_{0:k}\right)\ . \tag{2.13}$$

This implies that landmarks in the map are conditionally independent given the robot trajectory. Therefore, instead of using a single EKF with a state vector of dimension $|M_k|$, each particle uses a collection of $|M_k|$ fixed dimension EKFs to track individual landmarks separately. To do this the full trajectory has to be estimated, for which a particle filter is used. Finally by storing the features in a K-D Tree they managed to reduce the computational complexity of the algorithm to $O(|Z|\ log(|M|))$

Roller, Montemerlo, Thrun, and Wegbreit [9] proposed an improved version of FastSLAM, called FastSLAM 2.0, in which the measurements are included in the proposal distribution of the particle filter. This allows for a more efficient use of the particles, permitting either the use of a reduced number of particles or the potential closure of longer loops.

Nieto, Guivant, Nebot, and Thrun [10] introduced a different modification to the Fast-

SLAM algorithm by applying the Multiple Hypothesis Tracking concepts to deal with the data association problem in SLAM. Multiple Hypothesis Tracking is a deferred decision strategy in which multiple data association hypotheses are kept in anticipation that future measurements will disambiguate the data association uncertainty. In this algorithm, called MH-FastSLAM, when a particle determines that multiple hypotheses are possible it splits into several particles, one for each possible data association, including a new feature and false alarm hypotheses. After this, the regular weighting and re-sampling of particles is expected to resolve the uncertainty at a future time.

Thrun and Montemerlo [11] proposed a different strategy to solve the SLAM problem, called graph SLAM. This is also referred to in the literature as batch estimation and bundle adjustment from computer vision. Graph SLAM consists of progressively building a graph of soft constraints (referred in the paper as information constraints). In this graph each measurement $\boldsymbol{z}_k^i$ introduces a constraint between the landmark it represents and the robot's pose at time k, $\boldsymbol{x}_k$:

$$f_k(\boldsymbol{z}_k^i, \boldsymbol{x}_k) \equiv (\boldsymbol{z}_k^i - \boldsymbol{h}(\boldsymbol{x}_k, \boldsymbol{m}_k^{j(i)}))^T \boldsymbol{Q}_k^{-1} (\boldsymbol{z}_k^i - \boldsymbol{h}(\boldsymbol{x}_k, \boldsymbol{m}_k^{j(i)})) \ , \tag{2.14}$$

where $\boldsymbol{Q}_k$ is the covariance of the measurement model, which is assumed to have additive Gaussian noise. Similarly each movement of the robot introduces a constraint between two contiguous poses of the robot $\boldsymbol{x}_k$ and $\boldsymbol{x}_{k-1}$:

$$g_k(\boldsymbol{x}_k, \boldsymbol{x}_{k-1}) \equiv (\boldsymbol{x}_k - \boldsymbol{g}(\boldsymbol{u}_{k-1}, \boldsymbol{x}_{k-1}))^T \boldsymbol{R}_k^{-1} (\boldsymbol{x}_k - \boldsymbol{g}(\boldsymbol{u}_{k-1}, \boldsymbol{x}_{k-1})) \ , \tag{2.15}$$

where $\boldsymbol{R}_k$ is the covariance of the motion model, which is also assumed to have additive Gaussian noise. Using this graph the solution is obtained by minimizing the cost function (2.16).

$$\min_{\boldsymbol{x}_{0:k}} \quad \boldsymbol{x}_0^T \boldsymbol{\Omega}_0 \boldsymbol{x}_0 \ + \sum_k \sum_i f_k(\boldsymbol{z}_k^i, \boldsymbol{x}_k) + \sum_k g_k(\boldsymbol{x}_k, \boldsymbol{x}_{k-1}) \ , \tag{2.16}$$

where the $\boldsymbol{x}_0^T \boldsymbol{\Omega}_0 \boldsymbol{x}_0$ term is called an anchoring constraint, where $\boldsymbol{\Omega}_0$ should be a positive definite matrix, and it anchors the absolute coordinates of the map by locking the first pose of the trajectory to $[0, 0, 0]$. The minimization problem can be solved in many ways, but in the paper it was solved by linearising it and solving it in the information space. This way not only the mode is recovered but also its associated covariance.

Collectively all the methods previously described suffer from the problem of being vector based solutions to SLAM. This means that they have to rely on heuristics to solve both the data association and map management problems. For this reason this Thesis will focus on set based methods which rely on Bayes theorem to solve these problems, as well as the

conventional state estimation problem.

2.1.4 Random Finite Set based SLAM

Mullane, Vo, and Adams [12] introduced the concept of Random Finite Sets into the SLAM problem. By recognizing that the SLAM state is more naturally represented by a set, instead of a vector, they where able to include the data association and map management problems into the Bayesian estimation paradigm. Previous solutions to the SLAM problem, such as the ones described in the previous section, resolved these problems using heuristic approaches. Chapter 3 will provide the details of RFS-based SLAM.

2.2 Random Finite Sets

Traditional Bayesian state estimation has previously been focused on vector states. In this paradigm the problem is presented as having a prior distribution for the state vector $\boldsymbol{x}$:

$$\boldsymbol{x} \sim p(\boldsymbol{x}) \ . \tag{2.17}$$

And the user is provided with a measurement vector $\boldsymbol{z}$ and its corresponding measurement model:

$$\boldsymbol{z} \sim g(\boldsymbol{z}|\boldsymbol{x}) \ . \tag{2.18}$$

Using both of these and Bayes Theorem, the posterior is obtained:

$$p(\boldsymbol{x}|\boldsymbol{z}) = \frac{p(\boldsymbol{x})g(\boldsymbol{z}|\boldsymbol{x})}{p(\boldsymbol{z})} \ . \tag{2.19}$$

To apply this theorem when either the length of the state vector or the data association are unknown, heuristics with varying degrees of rigour are usually applied. In fact, Bayes theorem can only be strictly applied when the dimension of the measurement and state vectors, and the data association are known. When this is not the case, map management heuristics are required to associate the state vector elements with the corresponding elements in the measurement vector, so that Bayes theorem can be applied to the model with assumed known data association.

Mahler [13] proposed a new paradigm for Bayesian estimation in these cases, by replacing the state vector $\boldsymbol{x}$ with a RFS $\mathcal{X}$. In this context it is natural to think of the size of the set as a random variable. Also, by having no particular order, sets are well suited to handle the data association in a rigorous way. To solve this problem Mahler [13] introduced new tools to allow the propagation of random sets through Bayes theorem, called Finite Set

Statistics. By using these concepts a variety of new filters have been introduced based on various assumptions. These include the PHD, the cardinalized PHD (CPHD) [13] and the cardinality balanced multi-target multi-Bernoulli (CB-MemBer) [14] filters.

2.2.1 Probability Hypothesis Density filter

In this section, a simplified version of the PHD filter will be introduced. Further details will be presented in Chapter 3. In the PHD filter the state is assumed to be a Poisson RFS, this means that the elements are independent and identically distributed, and that the cardinality of the set follows a Poisson distribution:

$$|\mathcal{X}| = r \sim \frac{\lambda^r}{r!} e^{-\lambda} \ , \tag{2.20}$$

where r is the cardinality of the set $\mathcal{X}$ and λ corresponds to both the mean and variance of the distribution. The distribution of the set can then be modelled by its first moment $v_k(\boldsymbol{m})$, called its intensity or PHD. This intensity function is similar to a probability distribution but, instead of integrating to unity, it integrates to the mean of the number of randomly varying elements within the set.

Using this intensity the PHD filter's prediction equation propagates the intensity through time:

$$\begin{aligned} v_k^-(\boldsymbol{m}) = &\int p_{s,k}(\boldsymbol{\chi}) f_{k|k-1}(\boldsymbol{m}|\boldsymbol{\chi}) v_{k-1}^+(\boldsymbol{\chi}) d\boldsymbol{\chi} \\ &+ \int \beta_{k|k-1}(\boldsymbol{m}|\boldsymbol{\chi}) v_{k-1}^+(\boldsymbol{\chi}) d\boldsymbol{\chi} + v_k^b(\boldsymbol{m}) \ , \end{aligned} \tag{2.21}$$

where:

- $v_{k-1}^+(\boldsymbol{m})$ is the intensity of features at $\boldsymbol{m}$ at time $k-1$ given all measurements up to time $k-1$,
- $v_k^-(\boldsymbol{m})$ is the intensity of features at $\boldsymbol{m}$ at time k given all measurements up to time $k-1$,
- $p_{s,k}(\boldsymbol{\chi})$ is the probability of a feature at state $\boldsymbol{\chi}$ to survive to time k,
- $f_{k|k-1}(\boldsymbol{m}|\boldsymbol{\chi})$ is the state transition probability from state $\boldsymbol{\chi}$ to state $\boldsymbol{m}$,
- $\beta_{k|k-1}(\boldsymbol{m}|\boldsymbol{\chi})$ is the probability that a feature at state $\boldsymbol{\chi}$ will spawn another feature at $\boldsymbol{m}$,
- $v_k^b(\boldsymbol{m})$ is the intensity of the set of new features that might appear at $\boldsymbol{m}$.

In SLAM, however, features are usually considered to be static, permanent, and incapable of spawning new features, so the prediction equation simplifies to:

$$v_k^-(\boldsymbol{m}) = v_{k-1}^+(\boldsymbol{m}) + v_k^b(\boldsymbol{m}) \ . \tag{2.22}$$

The measurement set, $\mathcal{Z}$, is distributed according to a multi-target measurement model, in which each element, $\boldsymbol{m}$, in the state set $\mathcal{X}$ generates a measurement with probability $P_D(\boldsymbol{m})$ and generates no measurement with probability $(1 - P_D(\boldsymbol{m}))$. Clutter measurements are assumed to have a known Poisson distribution. Therefore, the measurement set $\mathcal{Z}$ is the union of the detection and the clutter sets. Using these assumptions, the PHD update equation

$$v_k^+(\boldsymbol{m}) = v_k^-(\boldsymbol{m}) \left[1 - P_D(\boldsymbol{m}) + \sum_i^{|\mathcal{Z}_k|} \frac{P_D \boldsymbol{h}(\boldsymbol{z}_i|\boldsymbol{m})}{\kappa(\boldsymbol{z}_i) + \int P_D \boldsymbol{h}(\boldsymbol{z}_i|\boldsymbol{m}) v_k^-(\boldsymbol{m}) d\boldsymbol{m}} \right] \tag{2.23}$$

has been derived [13]. In Equation (2.23), $\kappa(\boldsymbol{z}_i)$ is the intensity of the false alarm set at the position where measurement $\boldsymbol{z}_i$ was obtained. In this update equation, $v_k^-(\boldsymbol{m})(1 - P_D(\boldsymbol{m}))$ represents the possibility that features where misdetected while the second term represents detections. To implement this update equation the intensity can be represented either using sequential Monte-Carlo methods or a Gaussian mixture (GM)[13]. Sequential Monte Carlo methods have the advantage of being able to deal with highly non-linear measurement models while the GM-based implementation has a considerably reduced computational complexity, but is only able to deal with mildly non-linear measurement models (by using the Extended Kalman Filter to update means and covariances of each Gaussian Component). Because of the potentially large size of the SLAM state, which can range from hundreds to millions of features, only the GM-based solution will be used in this Thesis.

2.3 Detection Statistics for Feature Extractors

Within the autonomous robotic navigation literature, feature detection statistics are largely ignored, and the uncertainty is considered to lie solely in the spatial domain; this is typically modelled as range and bearing uncertainties [1, 15]. This implies that the probabilities of detection of features that have been associated are assumed to be unity, and the probabilities of false alarm of the associated measurements are assumed to be zero. The probabilities of detection of unassociated features are considered to be zero, while the probabilities of false alarms of unassociated measurements are assumed to be unity. In turn, it is then considered the task of the external map management and association heuristics to "deal" with false alarms and missed detections, before map estimation takes place. As can be seen from

Equation (2.23) the RFS-based PHD filter method includes the probability of detection P_D and false alarm distribution $\kappa(\boldsymbol{z}_i|\boldsymbol{x}_k)$ directly into the filter's update step, making the feature detector's detection statistics an intrinsic part of the state estimation process and its solution.

Within the tracking community, detection statistics are considered to be of prime importance, however object detection probabilities are usually naively considered to be constant, despite the fact that the varying relative positions of objects and the sensor, and any occlusions typically have a large effect on that object's detection probability [16]. Little attention is given to the shape of a sensor's FoV and the possibility of partial or total object occlusion, and their quantified effects on the expected detection statistics. Therefore the modelling of detection statistics, specifically for features derived from laser range finder data, is one of the main contributions of this Thesis.

In [17], the requirement for a feature detector is removed by modelling laser range data, in which multiple measurements can be produced, by single "extended" features. The number of such extended feature measurements is modelled as a Poisson RFS. They model the probability of detection, including occlusions, using the estimated features. Although this can be a good solution in tracking problems, in SLAM it is common for them to be unmodelled objects that can occlude the sensor. This is why the model proposed in this Thesis uses laser measurements, which are affected by modelled and unmodelled objects alike.

In [18], a modified PHD filter was introduced, that uses RADAR measurement amplitude information together with its accompanying range value. The distribution of this amplitude was modelled using one of the Swerling Models [19], which provide probabilistic models of received power fluctuations when RADAR to target viewing aspect changes. In the chosen model there is no dependency on the sensor position and the dependency on the environment is modelled by a single constant parameter d, which corresponds to the signal to noise ratio of the RADAR sensor:

$$p_{FA}(a) = a\ exp\left(\frac{-a^2}{2}\right), a \geq 0 \tag{2.24}$$

$$p_D(a) = \frac{a}{2(1+d)} exp\left(\frac{-a^2}{2(1+d)}\right), a \geq 0\ , \tag{2.25}$$

where a is the received amplitude of the radar signal, $p_{FA}(a)$ and $p_D(a)$ are the distributions of a for false alarms and targets of interest respectively. The inclusion of amplitude information was accomplished in three different ways, namely by having prior knowledge of the value of d, by estimating d along with the feature locations, and by having prior knowledge of the distribution of d and marginalizing it out of the PHD update equation. In every case the

modified PHD update equation is:

$$v_k^+(\boldsymbol{m}) = v_k^-(\boldsymbol{m})(1 - P_D(d)) + v_k^-(\boldsymbol{m}) \sum_i^{|\mathcal{Z}_k|} \frac{P_D(d) g_a^\tau(a|d) \boldsymbol{h}(\boldsymbol{z}_i|\boldsymbol{m})}{\kappa(\boldsymbol{z}_i|\boldsymbol{x}_k) g_{FA}^\tau(a) + \int P_D(d) g_a^\tau(a|d) \boldsymbol{h}(\boldsymbol{z}_i|\boldsymbol{m}) v_k^-(\boldsymbol{m}) d\boldsymbol{m}}, \tag{2.26}$$

where $g_{FA}^\tau(a)$ and $g_a^\tau(a|d)$ are the clutter and measurement likelihoods of the amplitude a to occur given that a detection threshold τ was exceeded:

$$g_{FA}^\tau(a) = \begin{cases} \frac{p_{FA}(a)}{\int_{a>\tau} p_{FA}(a) da} & a \geq \tau \\ 0 & a < \tau \end{cases} \tag{2.27}$$

$$g_a^\tau(a|d) = \begin{cases} \frac{p_D(a)}{\int_{a>\tau} p_D(a) da} & a \geq \tau \\ 0 & a < \tau \end{cases}. \tag{2.28}$$

The difference with the standard PHD update from Equation (2.23) is the inclusion of these likelihoods, which will augment the weight of measurements which are more likely to be true detections rather than clutter.

Using the Cramer Rao lower bound[1] [20] the authors showed that estimating d is impractical in the case of this particular sensor model. They also presented a reasonable prior distribution for d and showed that, by marginalizing d using this distribution, the filter performance in terms of the optimal sub-pattern assignment (OSPA) metric could be improved compared to experiments which ignored the amplitude information. This d or its prior distribution was also used to determine the probability of detection used in the PHD filter.

Although this filter includes the information that is typically provided by radar detectors, the requirement for the probability of detection of the filter is not removed. However, this statistic is usually not provided. In this Thesis a method to obtain the probability of detection in the case of LIDAR-based features will be presented. The incorporation of feature descriptor information within a feature detector, in a similar manner as in [18], will also be implemented within a PHD SLAM framework and compared to the standard PHD filter implementation, which ignores this information. Further analysis is required in order to determine the distribution that would be analogous to Clark et al. [18]'s $g_a^\tau(a|d)$ in the feature detector used in this Thesis. This analysis will be performed in Chapter 5.

[1]The Cramer Rao lower bound is a theoretical limit for the covariance of any unbiased estimator. It is equal to the inverse of the Fisher information matrix. An unbiased estimator that has a covariance equal to the bound is said to be efficient. [20]

Chapter 3

Random Finite Set SLAM

In this chapter, RFS-SLAM will be introduced along with the inclusion of the detection statistics of the feature detector.

3.1 Rao-Blackwellized (RB)-Probability Hypothesis Density (PHD) SLAM

The SLAM posterior PDF (Equation (2.8)) can be factored into the form [2, 8]

$$p\left(\boldsymbol{x}_{0:k}\middle|\mathcal{Z}_k,\boldsymbol{u}_{0:k}\right)p\left(\mathcal{M}_k\middle|\boldsymbol{x}_{0:k},\mathcal{Z}_k,\boldsymbol{u}_{0:k}\right)\ , \tag{3.1}$$

such that the first term in (3.1) is a conditional PDF on the robot trajectory and can be sampled using particles. The second term in (3.1) is the density of the map conditioned on the robot trajectory. In the RFS-based approach, the map RFS is also assumed to follow a multi-object Poisson distribution. This implies that features are independently and identically distributed, while the number of features is Poisson distributed.

$$|\mathcal{M}_k| = m \sim \frac{\lambda^m e^{-\lambda}}{m!} \tag{3.2}$$

$$p\left(\mathcal{M}_k = \{\boldsymbol{m}_k^1, \boldsymbol{m}_k^2, ..., \boldsymbol{m}_k^m\}\middle|\ |\mathcal{M}_k| = m\right) = m! \prod_{i=1}^{m} p_{\boldsymbol{m}}(\boldsymbol{m}_i) \tag{3.3}$$

In (3.3), $p_{\boldsymbol{m}}(\cdot)$ is the spatial distribution for the features in the map. Note that the $m!$ term comes from the fact that a set includes all possible permutations of its elements. This allows the PDF of the map RFS to be approximated by a time varying intensity function, v_k:

$$v_k = v_k(\boldsymbol{m}) = \lambda p_{\boldsymbol{m}}(\boldsymbol{m}) \tag{3.4}$$

The PDF is then approximated as follows:

$$p(\mathcal{M}_k = \{\boldsymbol{m}_1, \boldsymbol{m}_2 \ ... \ \boldsymbol{m}_m\}) = \frac{\prod_{i=1}^{m} v_k(\boldsymbol{m}_i)}{exp(\int v_k(\boldsymbol{m}) d\boldsymbol{m})} \tag{3.5}$$

In contrast to vector-based RB-PF approaches, which use the EKF to update the Gaussians for individual landmarks, a PHD filter is used to update the map intensity function [2]. A brief overview of the main steps in the RB-PHD filter now follows, highlighting the importance of detection statistics.

3.1.1 Particle Propagation

At time-step k, the particles representing the prior distribution,

$$\boldsymbol{x}_{k-1}^{i} \sim p\left(\boldsymbol{x}_{0:k-1} \middle| \mathcal{Z}_{1:k-1}, \boldsymbol{u}_{0:k-1}\right) \tag{3.6}$$

are propagated forward in time by sampling the motion noise, $\boldsymbol{\delta}_k^i$, and using the motion model (2.1):

$$\boldsymbol{x}_k^i = \boldsymbol{g}(\boldsymbol{x}_{k-1}^i, \boldsymbol{u}_{k-1}, \boldsymbol{\delta}_{k-1}^i) \sim p\left(\boldsymbol{x}_{0:k} \middle| \mathcal{Z}_{1:k-1}, \boldsymbol{u}_{0:k-1}\right) \tag{3.7}$$

This step is common to vector-based Rao-Blackwellized solutions to SLAM.

3.1.2 Prediction

For each particle, its map intensity from the previous update, v_{k-1}, is augmented with an arbitrarily small "birth" intensity v_k^b, according to the PHD filter predictor equation:

$$v_k^-(\boldsymbol{m}) = v_{k-1}^+(\boldsymbol{m}) + v_k^b(\boldsymbol{m}) \tag{3.8}$$

This "birth" intensity $v_k^b(\boldsymbol{m})$ represents the number of new features that might appear at $\boldsymbol{m}$ and is usually heuristically determined. This intensity is required to model the appearance of new features.

3.1.3 Map Update

The map intensity for each particle is updated with the latest measurements according to the PHD filter correction equation:

$$v_k^+(\boldsymbol{m}) = v_k^-(\boldsymbol{m}) \left[1 - P_D(\boldsymbol{m}) + \sum_i^{|\mathcal{Z}_k|} \frac{P_D(\boldsymbol{m}) \boldsymbol{h}(\boldsymbol{z}_i|\boldsymbol{m})}{\kappa(\boldsymbol{z}_i|\boldsymbol{x}_k) + \int P_D(\boldsymbol{m}) \boldsymbol{h}(\boldsymbol{z}_i|\boldsymbol{m}) v_k^-(\boldsymbol{m}) d\boldsymbol{m}} \right] \tag{3.9}$$

$$= v_k^-(\boldsymbol{m})(1 - P_D(\boldsymbol{m})) + v_k^-(\boldsymbol{m}) \sum_i^{|\mathcal{Z}_k|} \frac{P_D(\boldsymbol{m}) \boldsymbol{h}(\boldsymbol{z}_i|\boldsymbol{m})}{\kappa(\boldsymbol{z}_i|\boldsymbol{x}_k) + \int P_D(\boldsymbol{m}) \boldsymbol{h}(\boldsymbol{z}_i|\boldsymbol{m}) v_k^-(\boldsymbol{m}) d\boldsymbol{m}}$$

In (3.9), the first term is a copy of $v_k^-(\boldsymbol{m})$ scaled down by the factor $(1 - P_D(\boldsymbol{m}))$ to account for the possibility of missed detections. In the second term, note that instead of determining data association based on heuristics, the PHD filter determines how much a measurement should influence each landmark estimate.

3.1.4 Importance Weighting and Re-sampling:

The weighting and re-sampling of particles is the method used to update the robot trajectory PDF after propagation (also known as the proposal distribution). This is given by

$$p\left(\boldsymbol{x}_{0:k} \middle| \mathcal{Z}_{1:k-1}, \boldsymbol{u}_{0:k-1}\right). \tag{3.10}$$

This has to to be updated to become a new PDF representing the robot trajectory after measurement updates (or the target distribution),

$$p\left(\boldsymbol{x}_{0:k} \middle| \mathcal{Z}_{1:k}, \boldsymbol{u}_{0:k-1}\right). \tag{3.11}$$

Bayes rule allows the weighting distribution in terms of (3.10) and (3.11) to be expressed as

$$\frac{p\left(\boldsymbol{x}_{0:k} \middle| \mathcal{Z}_{1:k-1}, \boldsymbol{u}_{0:k-1}\right)}{p\left(\boldsymbol{x}_{0:k} \middle| \mathcal{Z}_{1:k}, \boldsymbol{u}_{0:k-1}\right)} = \eta p\left(\mathcal{Z}_k \middle| \boldsymbol{x}_{0:k}, \mathcal{Z}_{1:k-1}\right), \tag{3.12}$$

in which η is a normalizing constant. Since (3.10) and (3.11) are sampled using particles, the weighting distribution, defined as w_k, is also sampled such that a weight is calculated for

each particle. To solve (3.12), we use Bayes Theorem to express it as:

$$\begin{aligned} w_k &\equiv p\left(\mathcal{Z}_k \middle| \boldsymbol{x}_{0:k}, \mathcal{Z}_{1:k-1}\right) \\ &= p\left(\mathcal{Z}_k \middle| \mathcal{M}_k, \boldsymbol{x}_{0:k}\right) \frac{p\left(\mathcal{M}_k \middle| \mathcal{Z}_{1:k-1}, \boldsymbol{x}_{0:k}\right)}{p\left(\mathcal{M}_k \middle| \mathcal{Z}_{1:k}, \boldsymbol{x}_{0:k}\right)} \ . \end{aligned} \tag{3.13}$$

Equation (3.13) can be solved because the map RFS is assumed to be multi-object, Poisson distributed. Note from (3.13) that the choice of the map, $\mathcal{M}_k$, for which the expression is evaluated in its general form is theoretically arbitrary since the left-hand side of (3.13) is independent of the map. This has led to multiple solutions that adopt the empty-set strategy, the single-feature strategy and multi-feature strategy in determining the particle weight in (3.13). It was previously shown that the choice of the map can have a significant effect on the performance of the filter and that the performance of the multi-feature strategy is superior to the others [21]. This is achieved at the cost of an increased computational complexity. The multi-feature strategy is adopted in this work.

Importantly, within the above four steps, the map update and particle weighting steps require the knowledge of both the probability of detection of the feature detector and the distribution (PHD) of its false alarms. These important requirements are the subject of Section 3.4.

3.2 Gaussian Mixture implementation of RB-PHD SLAM

The first order approximation of the set based Bayes filter, the RB-PHD filter, presented in Section 3.1 is still intractable. This is because the intensity function of the map $v_k(\boldsymbol{m})$ is a general function of the entire landmark state space. A representation of it would have one degree of freedom for every possible value of the state vector (i.e., it would have to store the value of $v_k(\boldsymbol{m})$ for every possible value of $\boldsymbol{m}$), which is an infinite number. In practice however, by approximating $v_k(\boldsymbol{m})$ with a Gaussian mixture the number of variables needed to represent $v_k(\boldsymbol{m})$ becomes a finite number:

$$v_k = \sum_i \omega_k^i \mathcal{N}\left(\boldsymbol{m} \middle| \boldsymbol{\mu}_k^i, \boldsymbol{\Sigma}_k^i\right). \tag{3.14}$$

In this section the implementation details of the Gaussian mixture implementation of RB-PHD SLAM are demonstrated.

3.2.1 Prediction

In order to maintain the Gaussian mixture form of the PHD of the map. The prediction step of the filter requires the "birth" intensity to be a GM also:

$$v_k^b = \sum_i^{|\mathcal{Z}_{k-1}|} \omega_B \mathcal{N}\left(\boldsymbol{m}\middle|\boldsymbol{\mu}_k^i, \boldsymbol{\Sigma}_k^i\right) \ . \tag{3.15}$$

With this, the prediction step of the filter of Equation (3.8) becomes:

$$v_k^- = v_{k-1}^+ + \sum_i^{|\mathcal{Z}_{k-1}|} \omega_B \mathcal{N}\left(\boldsymbol{m}\middle|\boldsymbol{\mu}_k^i, \boldsymbol{\Sigma}_k^i\right) \ . \tag{3.16}$$

This means that the PHD prediction becomes the sum of the previously estimated GM and the "birth" GM.

3.2.2 Map Update

In the case of the update step, the spatial measurement model $\boldsymbol{h}(\boldsymbol{z}_i|\boldsymbol{x}_k, \boldsymbol{m})$ can be approximated using the EKF. This implies that $v_k(\boldsymbol{m})$ maintains its Gaussian form through the update step shown in Equation (3.9), which becomes:

$$v_k^+(\boldsymbol{m}) = (1 - P_D)v_k^-(\boldsymbol{m}) + \sum_i^{|\mathcal{Z}_k|} \sum_j^{N_k^-} \omega_k^{i,j} \mathcal{N}\left(\boldsymbol{m}\middle|\boldsymbol{\mu}_k^{[i,j]}, \boldsymbol{\Sigma}_k^{[i,j]}\right) \ . \tag{3.17}$$

As in Equation (3.9) the first term is a copy of v_k^- with lowered weights to account for the possibility of missed detections. The second term adds a new Gaussian for each pair comprising a new measurement and an existing Gaussian in the intensity map. Also, as in the previous definition, the PHD filter determines how much a measurement should influence each landmark estimate. This is carried out by the weighting factor calculation:

$$\omega_k^{i,j} = \frac{P_D \omega_k^j q\left(\boldsymbol{z}_k^i|\boldsymbol{\mu}_k^{[j]}, \boldsymbol{\Sigma}_k^{[j]}\right)}{\kappa + \sum_{l=1}^{N_k^-} P_D \omega_k^l q\left(\boldsymbol{z}_k^i|\boldsymbol{\mu}_k^{[l]}, \boldsymbol{\Sigma}_k^{[l]}\right)} \ , \tag{3.18}$$

where $q()$ is the measurement likelihood given a feature estimate, and κ is the clutter density, typically assumed constant over the FoV of the sensor and with a total mass ($\int \kappa \ dm$) equivalent to the expected number of false alarms at time k. The mean and covariance parameters for each new Gaussian, created from measurement i and landmark j, $\left(\boldsymbol{\mu}_k^{[i,j]}, \boldsymbol{\Sigma}_k^{[i,j]}\right)$, are determined using the EKF update step (note that other variants of the Kalman filter (KF) would also be possible to use).

3.2.3 Merging and Pruning of the Map

As can be seen from Equation (3.17), the number of Gaussians in $v_k(\boldsymbol{m})$ increases by a factor of $|\mathcal{Z}_k| + 1$ with each update. If left unchecked this would imply an exponential growth in Gaussian number. To account for this, Gaussians with small weights can be eliminated from the intensity function, while Gaussians that are close to each other can be merged [3, 12] This approximation is critical in limiting the computational requirement of the RB-PHD filter [22].

3.3 Limitations

As mentioned before, the PHD filter is a first order approximation of the Bayes filter for Random Finite Sets. Therefore it should have similar problems as its vector counterpart, the alpha-beta filter when compared to the Kalman filter. Some of these limitations are [13]:

- Feature number estimates tend to be very unstable in the presence of false alarms and, especially missed detections. In other words the PHD filter is forgetful and is more responsive to new measurements than to estimated priors.
- There is a great loss of information by representing the full map distribution with only its first moment. This can only be overcome by having a large enough signal to noise ratio; i.e., having high P_D and low clutter rate κ.

To overcome some of these difficulties a modification of the PHD filter, called the CPHD was devised [13]. This filter keeps track of both the first moment of the distribution (the PHD) and an estimate for the distribution of the total number of features, which can be arbitrary instead of Poisson. This stabilizes the total number of features at the cost of an increased computational complexity.

3.4 Detection statistics in RFS SLAM

In this chapter, the PHD filter was shown using a constant probability of detection P_D and false alarm density κ. In general, both can depend on the state of the robot and the state of the environment:

$$P_D = P_D(\boldsymbol{m}|\boldsymbol{x}_k, \Gamma) \tag{3.19}$$

$$\kappa = \kappa(\boldsymbol{z}|\boldsymbol{x}_k, \Gamma) \tag{3.20}$$

Here, Γ represents the full state of the environment, including but not limited to, the subsection of it being estimated (i.e., the feature locations). $P_D(\boldsymbol{m}|\boldsymbol{x}_k, \Gamma)$ is the probability of a

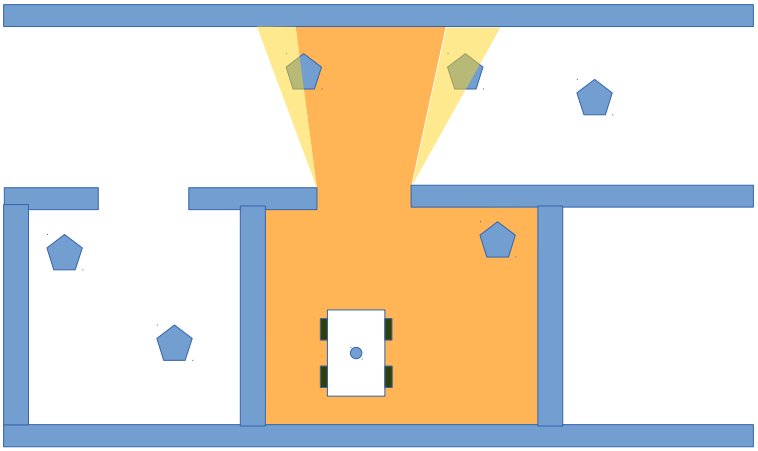

Figure 3.1: A pentagon detector that is occluded by walls, its probability of detection will not be constant. The yellow areas represent the zone where P_D is non zero and its intensity represents the value of P_D

feature at $\boldsymbol{m}$ generating a measurement. $\kappa(\boldsymbol{z}|\boldsymbol{x}_k, \Gamma)$ is the PHD for spurious measurements. For example, if the sensor being used is affected by occlusions then it may be the case that not all objects that can generate occlusions are represented in the map. In SLAM applications the probabilities of detection associated with feature extractors are usually not known and furthermore these probabilities cannot realistically be approximated by a constant value. Figure 3.1 shows a hypothetical shape detector (as an example in the figure a pentagon detector) that is occluded by walls. The value of P_D associated with each pentagon is therefore highly dependant on the state of both the robot and the environment. Even though the walls may not constitute detections of interest they still affect the probability of detection of the pentagons. The spatial distribution of false alarms is more complex to model since the source of false alarms can vary depending on the environment. Given this lack of knowledge about the false alarm distribution, approximating it by the least informative distribution; i.e., the uniform distribution, is usually accepted as a reasonable assumption. The expected number of false alarms can be estimated from real data or can be left as a parameter of the algorithm to tune. In the PHD filter formulation, the probability of detection can have any general shape. However, in the GM implementation of it, to maintain the GM form throughout the update step of the filter, shown in Equation (3.9), P_D can be assumed to vary slowly compared to the covariance of the Gaussians (and therefore be approximated by a piecewise constant function), approximated by another GM or a just a constant value.

3.4.1 Adding a buffer zone

Current implementations of the PHD filter are primarily concerned with the target tracking problem, which, although similar to the SLAM problem, does not have the same requirements.

In general it is not expected for a filter to continue tracking targets that have left the FoV of the sensor. In the SLAM paradigm, however, it is of prime importance to keep these estimates in order to continue the estimation process when the landmarks come back into the robot's FoV.

Given the PHD filter's propensity to discard old information in favour of the current measurements, as mentioned in Section 3.3, it frequently removes feature estimates that are close to the edge of the FoV and are expected to have high probability of detection but are not detected. This can occur either because there is an error in calculating P_D or because of the normal randomness of the process.

To account for this, a modification to the PHD filter is proposed. Feature estimates that are close to the edge of the FoV (i.e., in the buffer zone) are prevented from generating less weight (i.e., the sum of the weights generated by the Gaussian during the update step) than they had before the update step, while still allowing their spatial estimates to be updated. A Gaussian component is said to be in the buffer zone if its spatial uncertainty makes it possible (for the real feature it represents) to be both in and out of the sensor's FoV, this means that its probability of detection is an unknown number in the $[0, 1]$ interval. To deal with this uncertainty, a heuristic method is implemented. For every Gaussian in v_k that is in the buffer zone, the total weight that the Gaussian generated is compared to its weight before the update. If it is lower, the missed detection component is increased to compensate:

$$\omega_{k+}^{j} = \begin{cases} (1 - P_D)\omega_{k-}^{j} & if \quad \sum_{i=0}^{|\mathcal{Z}|} \omega_k^{i,j} \geq P_D \omega_{k-}^{j} \\ \omega_{k-}^{j} - \sum_{i=0}^{|\mathcal{Z}|} \omega_k^{i,j} & else \end{cases} \tag{3.21}$$

This heuristic, shown in Equation (3.21), tries to avoid the most destructive case, in which the probability of detection is zero but the algorithm assumes it is a non-zero value. While the heuristic intuitively makes sense, there are other alternatives to deal with this problem. These include changing the filter for another RFS-based method, such as the CB-MemBer filter, or using other heuristics. For example keeping the total weight generated by each Gaussian constant (for the Gaussian components that are in the buffer zone). The performance of the proposed heuristic will be evaluated in Chapter 7.

Chapter 4

Inferring Detection Statistics for Semantic Features

4.1 Estimating a Feature's Probability of Detection

The aim of this chapter is to provide a quantified model of the probabilities of detection and false alarm, corresponding to range based features. This model does not use any information on the feature detector itself and can therefore be used with any detector that estimates both the position and shape of an object. However, if only part of the model generates range values, then predicting whether the feature will be detected becomes impossible in the proposed approach. This means that models such as line detectors that do not provide beginning and end points, or circle detectors that do not exclude arc segments cannot be used together with the proposed method. Figure 4.1 shows the problem of modelling an arc segment using a circle, the predicted range values using the model are not accurate and are not useful for calculating the feature probability of detection.

As shown in Figure 4.2, given an estimate of the robot's location and the location and other attributes, such as the shape, of features (i.e., a SLAM estimate), the number of range points that the feature is expected to return can be estimated. This point estimation process is a sensor modelling technique referred to as *ray tracing* in the robotics literature [15]. Comparing estimated and measured distances allows expected range values to be labelled as either occluded or unoccluded. The number of estimated unoccluded points per feature can be considered to be a critical descriptor which determines the proportion of the landmark that is in the FoV of the sensor, similarly to the concepts of the Swerling models [19] applied in the radar literature which use the signal to noise ratio d as the critical descriptor. In other words it is assumed that the number of unoccluded points $n_p(\boldsymbol{m}, \boldsymbol{x}_k)$ is a sufficient statistic

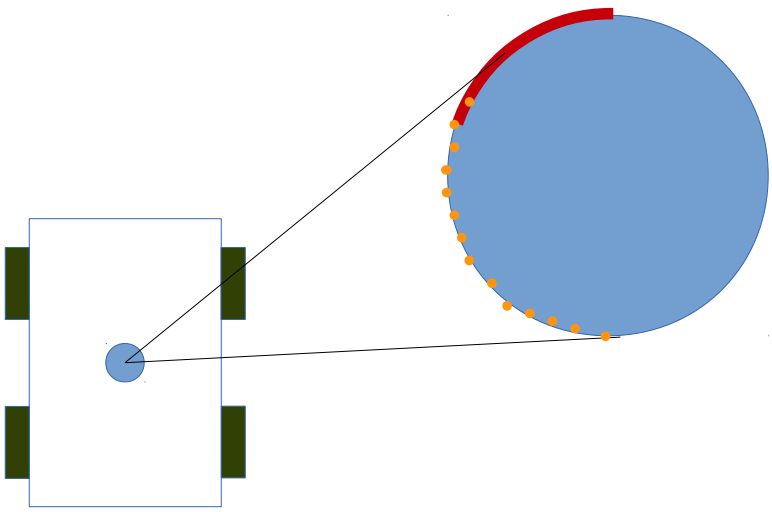

Figure 4.1: An arc segment (red line) that has been modelled using a circle (blue area). The yellow points represent predicted range values using the model.

of $P_D(\boldsymbol{m})$:

$$P_D(\boldsymbol{m}|\boldsymbol{x}_k, \Gamma, n_p(\boldsymbol{m}, \boldsymbol{x}_k)) \approx P_D(\boldsymbol{m}|n_p(\boldsymbol{m}, \boldsymbol{x}_k)) \tag{4.1}$$

As carried out in [18] this dependency can be calculated by integrating the variables used by the detector over the detectable area. Unlike a radar detector, which uses only the returned power for detection decisions, a general feature detector can use several quantities to make its decision. Equation (4.2) shows the probability of detection of a range based feature extractor that uses a descriptor $\boldsymbol{d}$ to make its decision:

$$P_D(\boldsymbol{m}|n_p(\boldsymbol{m}, \boldsymbol{x}_k)) = \int_{n>n_{min}, \boldsymbol{d}\in\boldsymbol{d}_{vol}} p(n, \boldsymbol{d}|n_p) dn \; d\boldsymbol{d} \; , \tag{4.2}$$

where n is the number of actual returns from the target, $\boldsymbol{d}$ is a descriptor used by the detector to make its decision, n_{min} is the minimum number of range points required to make a detection, and $\boldsymbol{d}_{vol}$ is the multi-dimensional volume in the descriptor space where the detector will decide to return a feature. One method to estimate the probability of detection is to estimate the multivariate distribution, $p(n, \boldsymbol{d}|n_p)$, and the using Equation (4.2) to determine P_D, however, the high dimensionality of the problem means that a large amount of data is required. A simpler method is to estimate $P_D(\boldsymbol{m})$ directly from the measurements. If the type of distribution of $p(n, \boldsymbol{d}|n_p)$ is known only its parameters need to be estimated and therefore the former method is preferable, but since the that information is not available, the latter method will be adopted in this Thesis.

The analysis in this section demonstrates that feature probabilities of detection can be experimentally quantified based on the number of unoccluded points, $n_p(\boldsymbol{m}, \boldsymbol{x}_k)$, via statistical

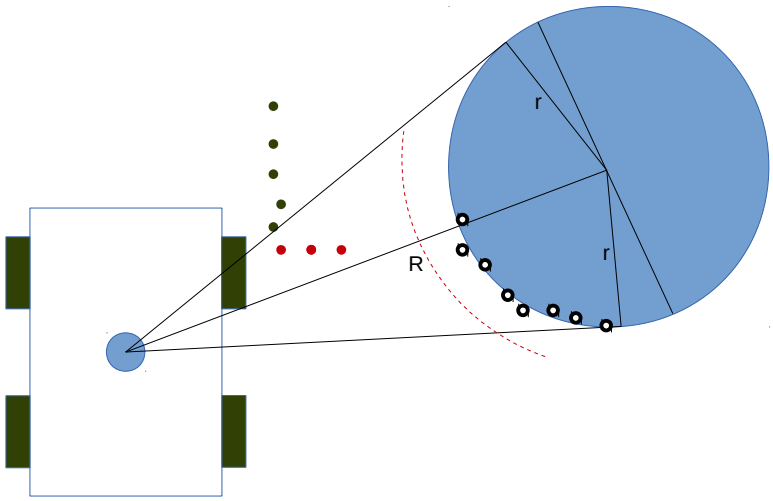

Figure 4.2: Analysis of range data from a circular shaped feature. Based on the SLAM state estimate, the range sensor beams that would hit the feature, if not occluded, can be determined (red and hollow circles). Beams with range values several times the range standard deviation shorter than expected (red points) are discarded from the detection probability analysis. The remaining (hollow) points are used to estimate the feature's probability of detection.

analyses on laser range data sets. Initially, a dataset is required from an environment where the ground truth positions of features are known, via independent means. A simple way to achieve this is through the use of features identifiable by humans - i.e. semantic features.

4.1.1 An application to the random matrix measurement model

In the context of [17] discussed in Section 2.3, the method to calculate number of unoccluded range values proposed in this chapter can be used to estimate the mean of the proposed extended feature RFS, allowing the probability of detection to be a constant value. In [17], the requirement for a feature detector is removed by modelling laser range data, in which multiple measurements can be produced, by single "extended" targets. The extended target model is called a random matrix and was initially proposed by Koch [23] within the target-tracking community. In this model an extended target is modelled using an extended state $\boldsymbol{m}_k^i$,

$$\boldsymbol{m}_k^i = \left\{\boldsymbol{x}_k^i, \boldsymbol{X}_k^i\right\} \ , \tag{4.3}$$

where $\boldsymbol{x}_k^i$ is referred to as the kinematic state and $\boldsymbol{X}_k^i$ is referred to as the extension state. The target process model is defined as

$$\boldsymbol{x}_{k+1}^i = \left(\boldsymbol{F}_{k+1|k} \otimes \boldsymbol{I}_d\right) \boldsymbol{x}_k^i + \boldsymbol{w}_{k+1} \ , \tag{4.4}$$

where $\otimes$ is the Kronecker product[1], $\boldsymbol{F}_{k+1|k}$ is an $s \times s$ state transition matrix with $(s-1)$ being the order to which the dynamics are modelled, $\boldsymbol{I}_d$ is the identity matrix of dimension d (with d being the number of dimensions of the target state space), and $\boldsymbol{w}_{k+1}$ is Gaussian random noise with covariance $\boldsymbol{W}_{k+1}$,

$$\boldsymbol{W}_{k+1} = \boldsymbol{Q}_{k+1|k} \otimes \boldsymbol{X}_k^i \ . \tag{4.5}$$

The measurement model is then defined as

$$\boldsymbol{z}_k^j = (\boldsymbol{H}_K \otimes \boldsymbol{I}_d)\, \boldsymbol{x}_k^i + e_k \ , \tag{4.6}$$

where e_k is white Gaussian noise with covariance equal to the target extension matrix $\boldsymbol{X}_k^i$. Each target generates a Poisson rate $\gamma(\boldsymbol{m}_k^i)$ number of measurements according to this model. And the clutter measurements are modelled as a Poisson RFS with a uniform distribution over the field of view.

To use these measurement and process models the extension state $\boldsymbol{X}_k^i$ has to be estimated along with the kinematic state $\boldsymbol{x}_k^i$. To accomplish this the Gaussian inverse Wishart distribution [24] is used to estimate the extended state $\boldsymbol{m}_k^i$. Using this distribution the Gaussian inverse Wishart PHD filter is defined. In this filter, the intensity function $v_k^+(\boldsymbol{m})$ is approximated as a mixture of Gaussian inverse Wishart distributions,

$$v_k^+(\boldsymbol{m}) = \sum_i \omega_k^i \mathcal{N}\left(\boldsymbol{x}\middle|\boldsymbol{\mu}_k^i, \boldsymbol{\Sigma}_k^i \otimes \boldsymbol{X}\right) \mathcal{W}^{-1}\left(\boldsymbol{X}\middle|\boldsymbol{\nu}_k^i, \boldsymbol{\Psi}_k^i\right) \ , \tag{4.7}$$

where each Gaussian inverse Wishart component is defined by its weight (ω_k^i), Gaussian distribution parameters (mean $\boldsymbol{\mu}_k^i$ and covariance $\boldsymbol{\Sigma}_k^i$), and inverse Wishart parameters (degrees of freedom $\boldsymbol{\nu}_k^i$ and scale matrix $\boldsymbol{\Psi}_k^i$), and corresponds to a Gaussian distribution, $\mathcal{N}(\cdot|\cdot,\cdot)$, to model the kinematic part of $\boldsymbol{m}$, $\boldsymbol{x}$, and an inverse Wishart distribution, $\mathcal{W}^{-1}(\cdot|\cdot,\cdot)$, to model the extension part, $\boldsymbol{X}$. Under these assumption the prediction step becomes

$$v_{k+1}^-(\boldsymbol{m}) = \sum_i \omega_{k+1}^{i-} \mathcal{N}\left(\boldsymbol{x}\middle|\boldsymbol{\mu}_{k+1}^{i-}, \boldsymbol{\Sigma}_{k-1}^{i-} \otimes \boldsymbol{X}\right) \mathcal{W}^{-1}\left(\boldsymbol{X}\middle|\boldsymbol{\nu}_{k+1}^{i-}, \boldsymbol{\Psi}_{k+1}^{i-}\right) \ , \tag{4.8}$$

where the values of the parameters of the predicted Gaussian inverse Wishart distributions are calculated according to the process model (Equation (4.4)). The specific equations can

[1]The Kronecker product for two matrices, $\boldsymbol{A}$ and $\boldsymbol{B}$, of dimensions $n_1 \times m_1$ and $n_2 \times m_2$ respectively, produces a result matrix of dimension $n_1 n_2 \times m_1 m_2$ and is defined as $\boldsymbol{A} \otimes \boldsymbol{B} = \begin{bmatrix} \boldsymbol{A}[1][1]\boldsymbol{B} & \cdots & \boldsymbol{A}[1][m_1]\boldsymbol{B} \\ \vdots & \ddots & \vdots \\ \boldsymbol{A}[n_1][1]\boldsymbol{B} & \cdots & \boldsymbol{A}[n_1][m_1]\boldsymbol{B} \end{bmatrix}$

be found in Appendix A. The update step becomes

$$v_{k+1}^{+}(\boldsymbol{m}) = \left(1 - \left(1 - e^{-\gamma(\boldsymbol{m})}\right) P_D(\boldsymbol{m})\right) v_{k+1}^{-}(\boldsymbol{m}) + \sum_{p \angle \mathcal{Z}_k} \sum_{\mathcal{U} \in p} \sum_{i} \omega_{k+1}^{i,\mathcal{U}} \mathcal{N}\left(\boldsymbol{x} \middle| \boldsymbol{\mu}_{k+1}^{i,\mathcal{U}}, \boldsymbol{\Sigma}_{k-1}^{i,\mathcal{U}} \otimes \boldsymbol{X}\right) \mathcal{W}^{-1}\left(\boldsymbol{X} \middle| \boldsymbol{\nu}_{k+1}^{i,\mathcal{U}}, \boldsymbol{\Psi}_{k+1}^{i,\mathcal{U}}\right) , \tag{4.9}$$

where $p \angle \mathcal{Z}_k$ means that p is a partition of the measurement set and the summation is over all possible partitions, $\mathcal{U} \in p$ means that $\mathcal{U}$ is one of the non-empty elements of p, and the updated parameters are calculated from the previous parameters using the measurement model and the probability of detection $P_D(\boldsymbol{m})$, clutter rate κ and average number of measurements per feature $\gamma(\boldsymbol{m})$ (see Appendix A).

In [17] the authors propose a variable probability of detection that accounts for occlusions caused by estimated targets by reducing the probability of detection of targets behind them. The mean number of measurements per target is modelled assuming that every target is of the same size. The method proposed in this Thesis can be used to estimate an expected number of measurements that includes the modelling of occlusions, either from estimated targets or from unknown sources, allowing the probability of detection parameter of the filter to be set to a constant value. To accomplish this, ray tracing algorithm be performed on ellipses calculated from the position an covariance of the random matrix. A 95% confidence ellipse would be a reasonable method to obtain such ellipses. Then the estimated number of unoccluded points n_p can be used as the mean of the extended feature RFS used. The probability of detection can then be set to a constant value or even one, leaving all missed detections to be modelled by the probability that a Poisson distribution of mean $\gamma(\boldsymbol{m})$ will generate zero measurements.

4.2 Estimating Probabilities of False Alarm

In the case of the probability of false alarm, it is infeasible to theoretically model every possible range scan that does not contain a semantic feature. Importantly, the statistical representation of false alarms in RB-PHD-SLAM is a Poisson random set, which only requires an estimate of their expected number. In a manner similar to the probabilities of detection outlined above, the statistical analysis of range based data, known to not contain the chosen semantic features, can yield an informative estimate of the probability of false alarm. This data can be obtained by either taking a scan in an environment known not to contain the features of interest or by identifying and removing real detections from a dataset. Finding an environment without features of interest that is representative of the environment where the robot will be expected to operate can be difficult. Therefore the approach taken in this Thesis is to remove the true detections from a real dataset. This can be accomplished by

obtaining a dataset with known robot poses and feature positions and removing detections that are close to the feature positions. A statistical analysis on the rest of the detections can then be performed. Given the Poisson limitation of the PHD filter, only the average number of false alarms is required. The clutter intensity κ, is determined as the average number of false alarms divided by the area of the sensor's FoV.

These concepts of estimating a features detection and false alarm probabilities will be applied to a laser range finder based circle detector in Chapter 6. The next chapter introduces the circle detector itself, in the context of well known feature detectors within the robotics field.

Chapter 5

Laser Based Features and Their Detection Statistics

This chapter provides a brief overview of the main feature detection algorithms applied to range data and, at the same time, highlights the known problems in their application to data in which the chosen feature types yield few returns per scan. A simple circular feature detector will then be presented, which can be applied in outdoor scenarios in which approximately circular cross-sectioned features such as trees, pillars and lamp posts are abundant. This circular feature detector is an extension of that proposed by Dissanayake, Newman, Clark, Durrant-Whyte, and Csorba [4], and is applied to laser range scans.

5.1 Why Semantic Features?

It is possible to incorporate raw range finder decisions into a SLAM algorithm. Various mapping algorithms achieve this via scan matching techniques [25]. In scan matching the relative pose of a laser scan with respect to another scan, usually the previous one, is found by minimizing an error measure between both scans. Most of these techniques rely on non-linear optimization algorithms that usually require a good initial estimate of the robot pose to converge to the correct solution. They also have accumulative error, meaning that pose estimates will diverge eventually. This is dealt with by periodically applying global optimization algorithms.

The reasons why most SLAM algorithms do not attempt to process every laser range value are as follows. Firstly, contrary to many radar and sonar devices, commercially available laser range finders usually internally process the received power values to provide range decisions at distinct bearing angles, instead of providing the entire received power array (A-scope) at predefined range increments. This means that such devices make their own hypothesis test on a per A-scope basis, and provide only the final decision of this test, yielding a single range

decision. Under favourable operating conditions, these range decisions typically correspond well with the true distances to objects, however they are still prone to the problems of false alarms and missed detections under sub-optimal target/environmental conditions. These include dust or fog in the air, resulting in clutter measurements, and reflective surfaces that generate artefacts. Secondly, because of the high angular resolution of laser range finders, they usually provide the user with a multitude of range decisions, far in excess of which most SLAM algorithms can process.

These two facts have advocated the compression of laser range data into so called high level, and typically semantic, features. This is to minimize the negative impact of individual false alarms and missed detections and simultaneously keep SLAM input data levels manageable. High level features are also usually more salient (i.e., distinct), which makes it easier to solve the data association problem. This can be performed as part of the Bayesian state estimation problem (as in the RFS based methods), or separately from the Bayesian state estimation problem, using heuristics (as in most vector based solutions).

5.2 Current Laser Range Based Feature Detectors

Global detectors, such as random sample and consensus (RANSAC) and the Hough transform, have been applied to laser range data, mostly to extract lines [26]. In this context, a global feature detector simultaneously uses all the available sensor information to extract features. In the case of laser sensors, this would be the entire scan. These methods have several advantages, such as tolerance to partial occlusions and the ability to detect larger objects. However, they rely on many feature inliers being available within the laser data sets. Local feature detectors examine only a small section of the data at a time, potentially missing larger or partially occluded features. However, they usually have a much lower computational complexity than their global counterparts.

Núñez et al. [27] demonstrated a detector capable of extracting both line segments and circular curves from a scan using a curvature measure. Other laser point based feature detectors include the recursive split and merge algorithm [28] and Gauss-Newton extraction algorithm [29]. These algorithms were designed to work with indoor scans where circles are usually observed at close range, from man-made objects such as pillars and furniture. However, the environment where the experiments for this Thesis were conducted has very few such features and therefore these detectors were not implemented.

Despite the varying degrees of mathematical rigour in state of the art feature detection algorithms, global detectors have been shown to not improve the results enough to compensate for their increased computational complexity, compared to local detectors [26]. Dissanayake, Newman, Clark, Durrant-Whyte, and Csorba [4] presented a simple detector which seeks clusters of points and assigns a circle to represent these, with diameter equal to the distance

between the first and last points of the cluster. The detector presented was shown to perform well enough for SLAM applications in an outdoor environment similar to the one used in this Thesis. Given the low complexity and effectiveness of the algorithm it was chosen as a basis for the feature detector used in this Thesis. This algorithm will be augmented in the next section, for the robust detection of circular cross sectioned objects including pillars, trees and lamp posts. Both the estimation of the detection statistics explained in Chapter 6 and the improvements proposed in the following section are generic and can be applied to any semantic feature detector.

5.3 Detection of Circular Objects

The circle based detector of [4] is extended here by, replacing it's heuristic estimation of the parameters of the circle by a non-linear optimization approach, which should give a circle that fits more closely to the data. Also an additional step has been added to remove some of the false alarms produced by the algorithm.

The detector works in 3 steps: Clustering; circle fitting; and false alarm reduction.

5.3.1 Clustering

The first step of the algorithm is to segment the laser scan into a collection of simple clusters of closely spaced points. To obtain these clusters the whole scan is processed in the order in which data was collected. If the Euclidean distance between two consecutive points is greater than a threshold, a break between clusters is declared.

5.3.2 Circle Fitting

A circle is fitted to each cluster by minimizing the mean squared error of the fit as shown in Equation (5.1).

$$\underset{x_c,y_c,r_c}{\text{minimize}} \qquad \sum_i (\sqrt{(x_i - x_c)^2 + (y_i - y_c)^2} - r_c)^2 \ , \tag{5.1}$$

where (x_c, y_c, r_c) is the center and radius of the circle and (x_i, y_i) are the coordinates of each laser range point in the cluster. This optimization problem is the same as the one presented in [29] but is solved using the Levenberg-Marquardt algorithm, which has been shown to be more robust to bad initialization and high non-linearities than the Gauss-Newton method [30]. To initialize the algorithm the mean point position is used as the circle center and its radius is set as half the distance between the first and last point.

5.3.3 False Alarm Reduction

At this stage, every cluster has an associated, fitted circle. Cluster pruning is then necessary, in which clusters and their corresponding circles are removed based on a detection theoretic, statistical analysis of their parameters.

Each cluster is characterized by three parameters, which are expected to differ in circular cross sectioned objects from the rest of the clusters:

- Fitting Error for the circle fit (FE)

$$\text{FE} = \sum_i (\sqrt{(x_i - x_c)^2 + (y_i - y_c)^2} - r_c)^2 \tag{5.2}$$

- Radius of the detected circle (r_c)

- Convexity (c) of the circle. This is a measure of the difference between the distance from the robot to the center of the fitted circle and the mean of the points (See Figure 5.1) and is given by

$$c = \sqrt{(x_r - x_c)^2 + (y_r - y_c)^2} - \sqrt{(x_r - \frac{1}{n}\sum_i^n x_i)^2 + (y_r - \frac{1}{n}\sum_i^n y_i)^2} \tag{5.3}$$

  This is expected to differentiate circular objects from circular arcs, for which the circular model does not fully apply and will therefore impede the ray tracing algorithm from estimating the laser measurements (the relevance of this can be observed in Chapter 6). Circular arcs that are convex with respect to the robot will still produce detections, but an effective way to eliminate such errors was not found.

To achieve false alarm reduction, based on the above parameters, concepts from detection theory can be applied [31]. Histograms representing correctly and falsely detected circular features were generated with respect to each of the above parameters. This required the generation of ground truth information within a test area, containing the true centers and radii of circular sectioned objects, such as trees. This test area comprised a ground truth map of a park area near the Universidad de Chile.

Naturally, the generality of such an environment is questionable, in terms of the circular features contained within it. However, since the SLAM experiments were to take place in an environment containing a significant number of trees, this environment was deemed sufficiently general. In general, if the sought features are based on *any type* of semantic information, detectors for those semantics can be tuned in a similar manner, using ground

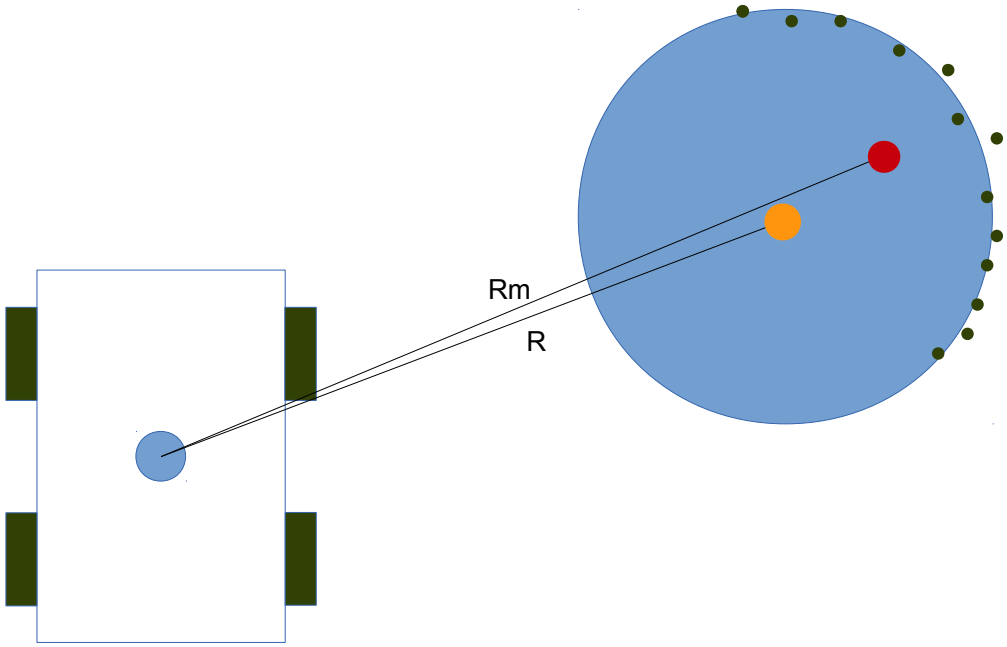

Figure 5.1: If the mean of the laser returns (red point) is further away from the robot than the center of the estimated circle (orange point) then the object does not form a convex circular cross section with respect to the robot's location.

truth data sets from environments known to contain a significant number of the type of feature sought.

Within the park environment, multiple 2D laser scans from different positions were recorded and approximately manually aligned to form an initialization for the iterative closest point (ICP) [32] algorithm, which in turn generated a more exact alignment of the data. This resulted in a registered 2D point cloud of the Park. Point clusters were then manually extracted and compared to the actual environment to determine if they corresponded to actual circular sectioned features. Based on positive matches, the cluster centers and actual features radii (measured by hand) were noted. This resulted in a list of circular feature (typically tree trunk) center coordinates and their respective radii. ICP [32] also determined the position at which each scan was recorded.

After the ground truth list was attained, the laser range finder and circular feature detector were used to automatically detect multiple circular sectioned features, at multiple locations, based on the procedure outlined in Sections 5.3.1 and 5.3.2 within the test area. From these multiple detections, the histograms in Figures 5.2, 5.3 and 5.4 were generated. These histograms could be used directly to achieve false alarm reduction. From each histogram it is evident that the application of appropriate, independent detection thresholds on the fitting error, radius and convexity measure could be applied so as to reject the false alarms which correspond to feature parameters outside of the bounds that contain the detec-

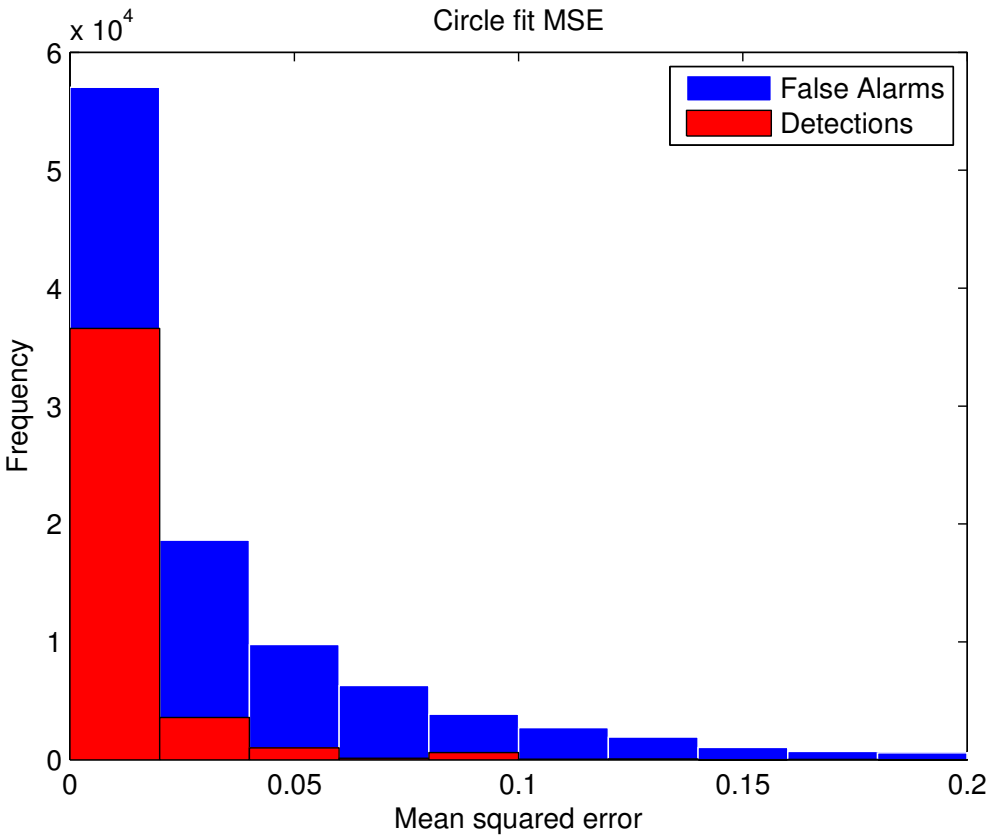

Figure 5.2: Histogram for the fitting error. Note: The tail for the false alarm distribution is very long and not all of it is shown in this

tions. However, care is necessary before the application of such a simplistic treatment, since any correlation between these parameters must first be determined.

To make use of such correlations standard techniques such as Fisher's linear discriminant [33] or Hotelling ellipsoids [31] can be used to discard some of the false alarms. After comparing these two methods, higher detection rates, for given false alarm rates (i.e., superior receiver operating characteristics (ROCs)) were noted for the Hotelling ellipsoidal method. Therefore, the Hotelling ellipsoidal method is adopted here. Figure 5.5 shows the ROC curve, which shows that for any given number of false alarms the Hotelling ellipsoid method produces a higher number of true detections.

Hotelling Ellipsoid Method

This method fits a multivariate Normal distribution to the detected circle's parameters and uses the distribution parameters to create a confidence ellipsoid, as shown in Figure 5.6. The figure shows the distribution of the three variables for the case of true detections and a corresponding 99.9% confidence ellipsoid representing a multivariate Normal distribution,

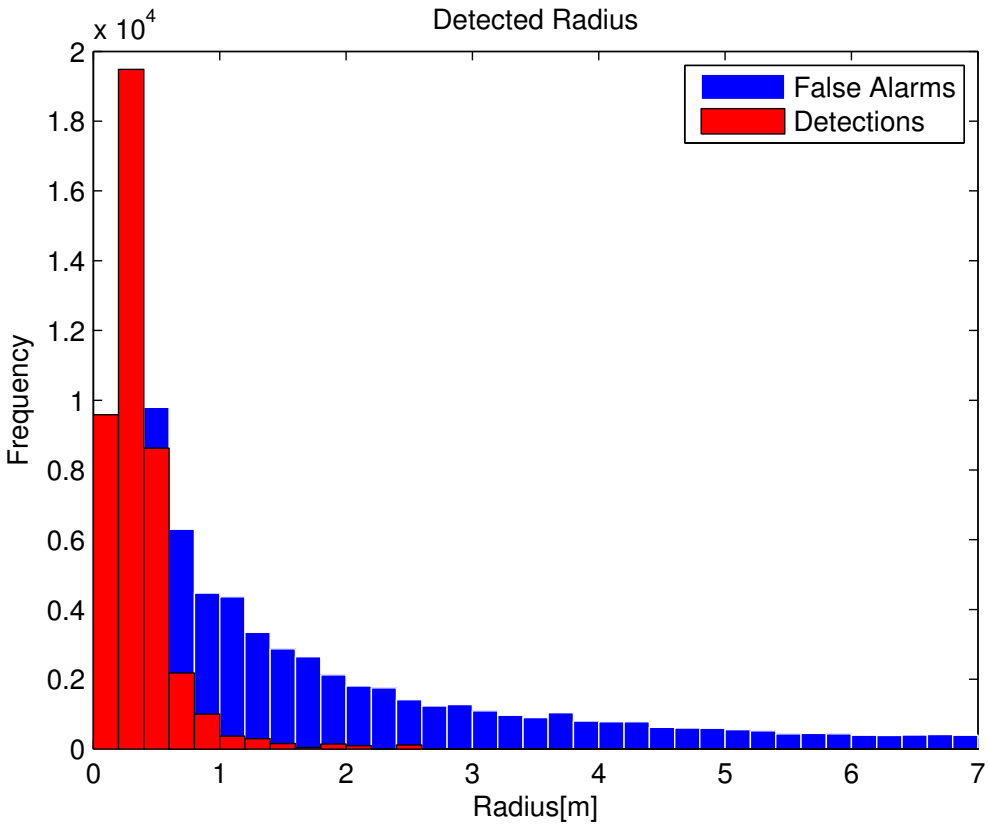

Figure 5.3: Histogram for the detected radius. Note: The tail for the false alarm distribution is very long and not all of it is shown in this figure.

based on the parametric data mean $\boldsymbol{\mu}$ and covariance $\boldsymbol{\Sigma}$. This ellipse can be obtained by applying a threshold to the squared Mahalanobis distance of the data to the estimated mean, $\boldsymbol{\mu}$, and covariance, $\boldsymbol{\Sigma}$,

$$(\boldsymbol{\mu} - \boldsymbol{x})^T \boldsymbol{\Sigma}^{-1} (\boldsymbol{\mu} - \boldsymbol{x}) = \gamma \ . \tag{5.4}$$

The value of the threshold γ (i.e., the size of the ellipse) can be determined using the fact that the square of the Mahalanobis distance has a χ^2 distribution and looking at the appropriate table. The Hotelling Ellipsoidal method will reject any measurement that falls outside of the ellipse. To accompany the method of estimating the detection statistics explained in Chapter 4 a technique similar to Clark et al. [18] is now applied. In addition to rejecting false alarms, the multivariate Gaussian model allows the parameter vector to be used as an amplitude value for the measurement, as carried out in [18] with a Rayleigh distribution. By using the multivariate Gaussian to model true detections, and a uniform distribution in the detectable volume for false alarms we can determine equivalent distributions to g_a and g_{FA},

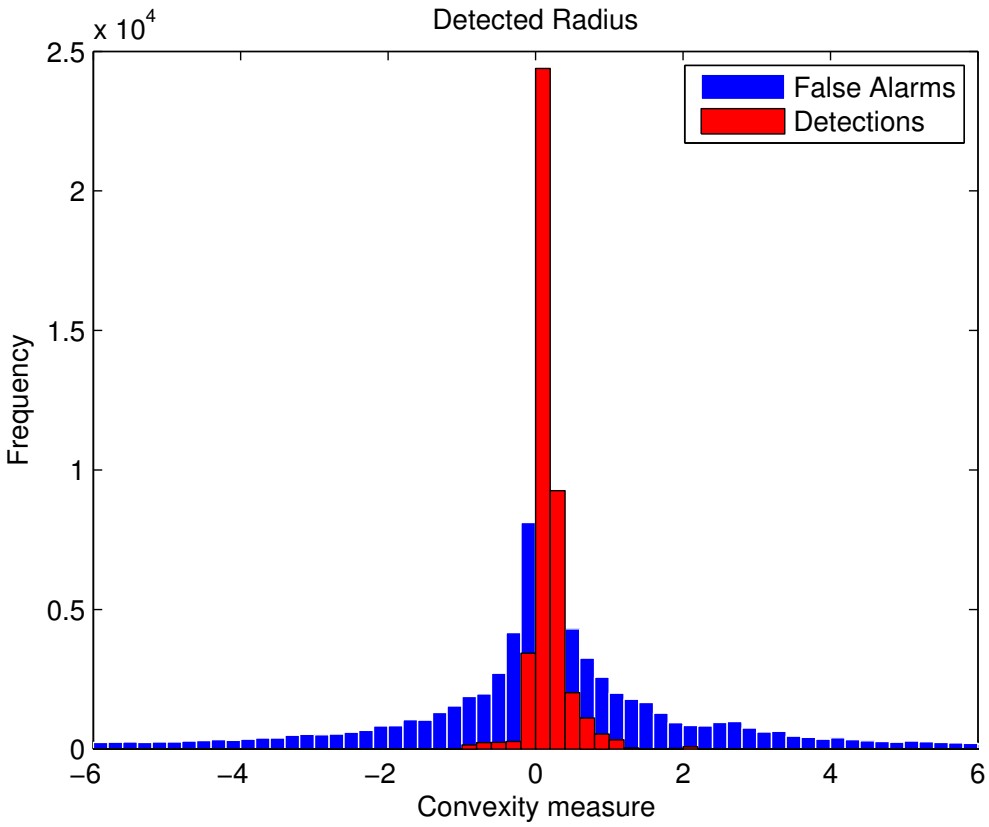

Figure 5.4: Histogram for the Convexity measure. Note: The tails for the false alarm distribution are very long and not all of it is shown in this figure.

respectively.

$$g_{FA}^{\gamma}(x) = \begin{cases} \dfrac{1}{\displaystyle\int_{(\boldsymbol{\mu}-\boldsymbol{x})^T\boldsymbol{\Sigma}^{-1}(\boldsymbol{\mu}-\boldsymbol{x})<\gamma} 1 \; d\boldsymbol{x}} & if \; (\boldsymbol{\mu}-\boldsymbol{x})^T\boldsymbol{\Sigma}^{-1}(\boldsymbol{\mu}-\boldsymbol{x}) \leq \gamma \\ 0 & if \; (\boldsymbol{\mu}-\boldsymbol{x})^T\boldsymbol{\Sigma}^{-1}(\boldsymbol{\mu}-\boldsymbol{x}) > \gamma \end{cases} \tag{5.5}$$

$$g_{x}^{\gamma}(x|\Sigma,\mu) = \begin{cases} \dfrac{\mathcal{N}\left(\boldsymbol{x}|\boldsymbol{\mu},\boldsymbol{\Sigma}\right)}{\displaystyle\int_{(\boldsymbol{\mu}-\boldsymbol{x})^T\boldsymbol{\Sigma}^{-1}(\boldsymbol{\mu}-\boldsymbol{x})<\gamma} \mathcal{N}\left(\boldsymbol{x}|\boldsymbol{\mu},\boldsymbol{\Sigma}\right) d\boldsymbol{x}} & if \; (\boldsymbol{\mu}-\boldsymbol{x})^T\boldsymbol{\Sigma}^{-1}(\boldsymbol{\mu}-\boldsymbol{x}) \leq \gamma \\ 0 & if \; (\boldsymbol{\mu}-\boldsymbol{x})^T\boldsymbol{\Sigma}^{-1}(\boldsymbol{\mu}-\boldsymbol{x}) > \gamma \end{cases} \tag{5.6}$$

These distributions can then be directly applied, along with the detections statistics determined in Chapter 4, to the modified PHD filter update from Equation (2.26). The application of this method in RB-PHD-SLAM will be evaluated in Chapter 7.

Using the features extracted by this detector a SLAM solution can be implemented. The next chapter will address how the detection statistics of this detector are obtained.

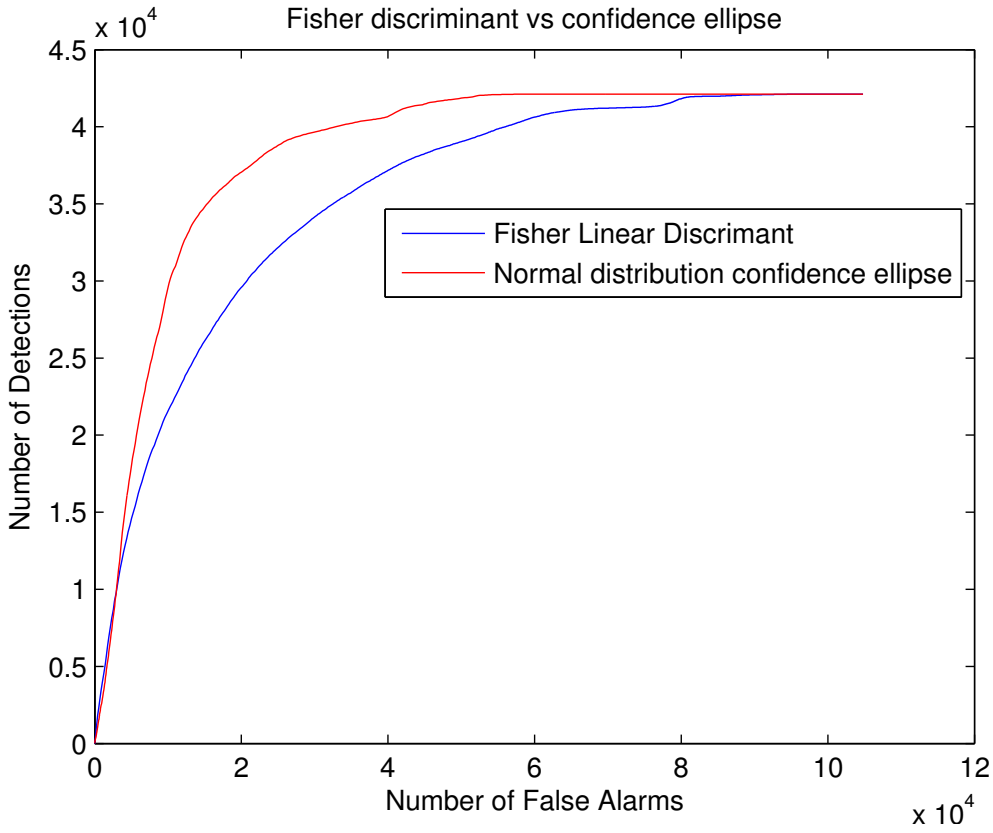

Figure 5.5: ROC curve for Fisher's linear discriminant (blue line) and Hotelling ellipsoid (red line) methods. The Hotelling ellipsoid method shows a superior performance.

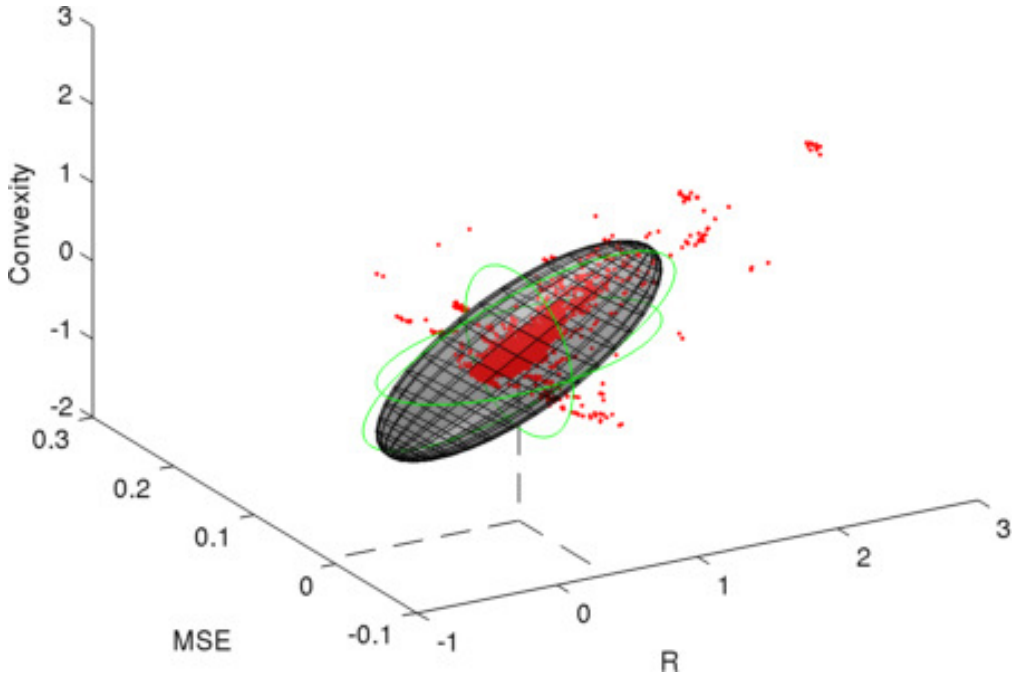

Figure 5.6: A Gaussian approximation of the detections can be used to generate an ellipse to remove much of false alarms and keep most of the detections.

Chapter 6

On-Line Determination of Detection statistics

This chapter will show the results of applying the method described in Chapter 4 to the detector presented in Chapter 5 using the ground truth dataset to obtain detection probabilities. The idea, shown in Equation (6.1), is to approximate the probability of detection of a feature given all available information by the probability of detection given the value of a single parameter, in this case the number of unoccluded points

$$P_D(\boldsymbol{m}_i|\mathcal{M}_k, \boldsymbol{x}_k) \approx P_D(\boldsymbol{m}_i|n_p = N_p(\mathcal{M}_k, \vec{x}_k)) \ , \tag{6.1}$$

where:

- n_p is the number of unoccluded points,
- $N_p()$ is the function that estimates n_p via ray tracing

The circular feature detector (from the previous chapter) was applied to the scans collected from the park data set. By using the known poses of the robot and landmark, every measurement was associated with it's closest feature. If the distance between these measurements and their closest associated features was larger than one meter, the measurements were deemed to be false alarms.

The associated measurements were used to determine the probability of detection of the algorithm conditioned on the number of unoccluded points. First, the expected number of unoccluded points for each feature-robot pose pair was calculated based on the estimated robot pose and feature positions. For every possible number of unoccluded points, a different number of data points (i.e., instances where that specific number of unoccluded points was calculated) were obtained. By determining the fraction of these data points that resulted in

Table 6.1: The number of unoccluded points greatly influences the probability of detection. The amount of data points used to calculate P_D is also shown.

$N_p(\mathcal{M}_k, \vec{x}_k)$	P_D	Number of data points
0	0.0014	273025
1	0.0244	62025
2	0.0696	28582
3	0.3595	19423
4	0.7562	10374
5	0.8854	6909
6	0.8987	4185
7	0.7824	2656
8	0.7607	1404
9	0.7679	1133

real detections the probability of detection for each detected feature was determined (Equation (6.2)).

$$P_D(\boldsymbol{m}_i|n_p) \approx \frac{\sum_{j=0}^{N_s} \sum_{\boldsymbol{m}\in\mathcal{M}_j} N_p(\boldsymbol{m}, \vec{x}_j) = n_p \wedge C(\boldsymbol{m}, j)}{\sum_{j=0}^{N_s} \sum_{\boldsymbol{m}\in\mathcal{M}_j} N_p(\boldsymbol{m}, \vec{x}_j) = n_p}, \tag{6.2}$$

where:

- N_s is the total number of scans in the dataset,
- $C(\boldsymbol{m}, j)$ is an indicator function that shows whether feature $\boldsymbol{m}$ was detected at time j.

As can be seen in Table 6.1 the probability of detecting a circular object (in this case a tree) is highly dependant on the number of unoccluded points. It should be noted that there is less data for high number of unoccluded points, so for higher number of unoccluded points the uncertainty of the estimated P_D will be higher. To avoid using the values of P_D calculated with only a few data points, in the RFS-SLAM implementation, if more than six points were not occluded, then the probabilities of detection were replaced with the probability of six unoccluded points. For a low enough number of unoccluded points the probability of detection is expected to drop to zero. This is enforced by replacing the probabilities that were lower than 10% with a zero. Otherwise, any filter that uses this information would continuously expect to observe the features that are out of the FoV and slowly remove them from the map. Figure 6.1 shows the result of calculating the probability of detection for a 0.1 meter radius feature over the entire FoV of the robot. The red points show the laser scan which produced occlusions while the color scale represents P_D. Note that, in the absence of occlusions, the number of point returns that can hit the feature, and therefore the probability

of detection, reduces slowly with distance. Additionally, occlusions produce a sharp decrease in the probability of detection behind them.

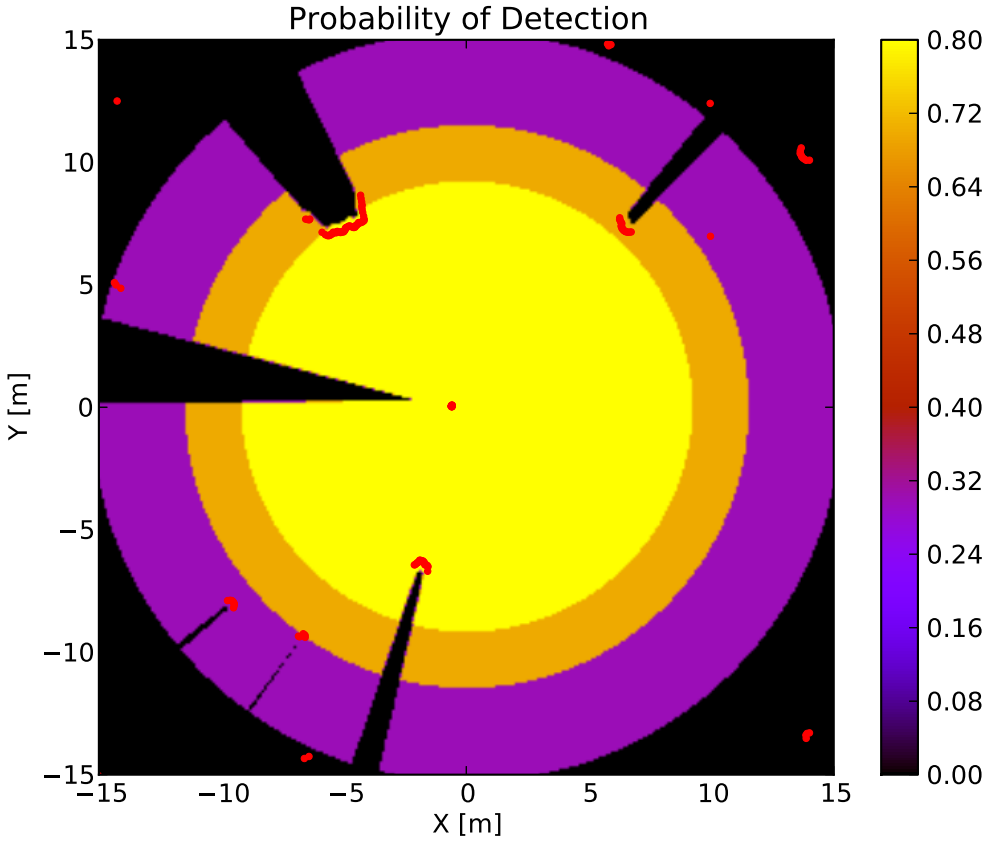

Figure 6.1: Probability of detection for a specific laser-scan and a feature radius of 0.1 meters.

6.1 Implementing the buffer zone

As described in Section 3.4.1, adding a buffer zone at the edge of the FoV is expected to improve the results of the RFS SLAM algorithm because of the poor performance of the PHD filter with low P_D and with P_D mismatches. In order to achieve this for the proposed method for calculating P_D the following definition for the buffer zone, where the rule from Equation (3.21) will be applied, is proposed:

- Any point where $0 < P_D < 0.5$ is considered to form part of the buffer zone.
- P_D is calculated in several positions along the direction perpendicular to the robot-landmark line. Then if this collection of probabilities contains both zero and non-zero values the landmark is declared to be in the buffer zone and the maximum probability is used.

These two rules define a buffer zone which is expected to cover the edge of the FoV of the robot (i.e., the areas close to transitions between $P_D = 0$ and $P_D > 0$). Figure 6.2 shows the probability of detection calculated using this new process, while in Figure 6.3 the shape of the buffer zone can be seen in white (i.e. within the white area the rule from Equation (3.21) is applied). As can be seen from Figure 6.3 the buffer zone covers areas close to the edge of the FoV. Occlusions from large or close objects still make the probability of detection approach zero for the space behind them. Objects that are narrow or far away produce occlusions that are covered completely by the buffer zone and therefore the probability of detection behind them does not decrease. A performance evaluation of using this method will be provided in Chapter 7.

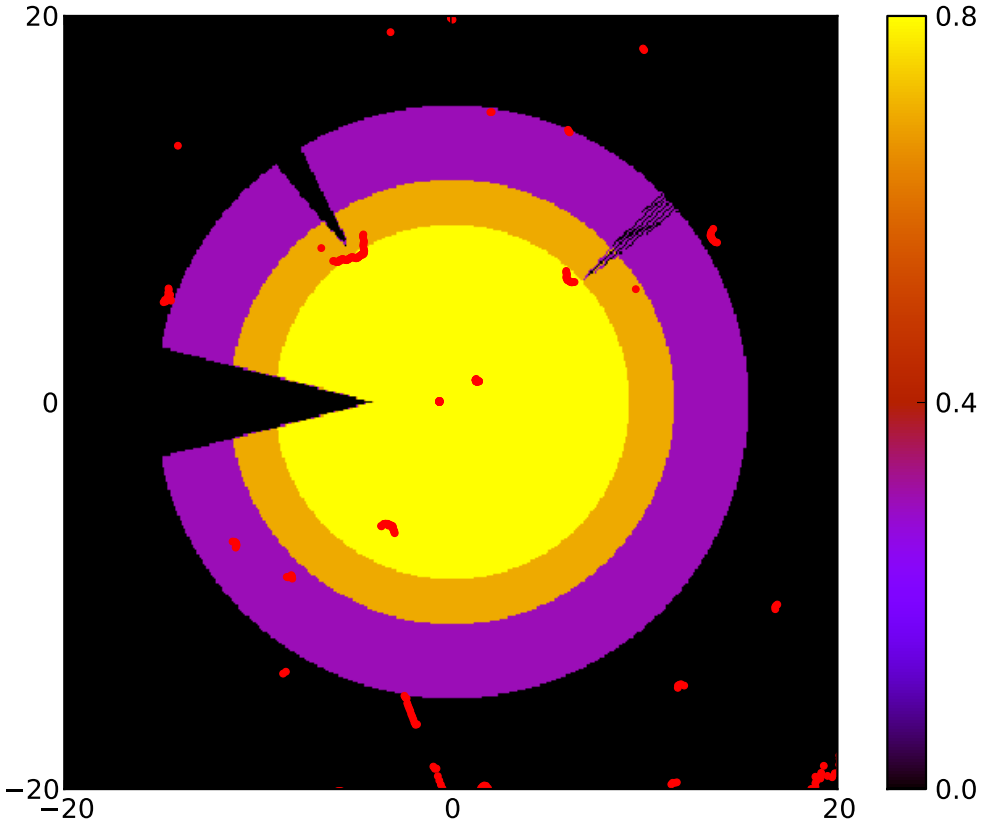

Figure 6.2: Probability of detection for a specific laser-scan and a feature radius of 0.1 meters, using a buffer zone.

Unlike the probability of detection, false alarms are typically modelled in a much simpler fashion. A uniform false alarm distribution is generally considered appropriate [13]. Therefore only the distribution for the number of false alarms per laser scan is necessary, which does not require any SLAM map estimates. Accordingly, the false alarm histogram, obtained by plotting the number of times a particular false alarm number occurs over N_s scans, is shown in Figure 6.4 where a continuous Poisson distribution is also fitted to the data. The PHD

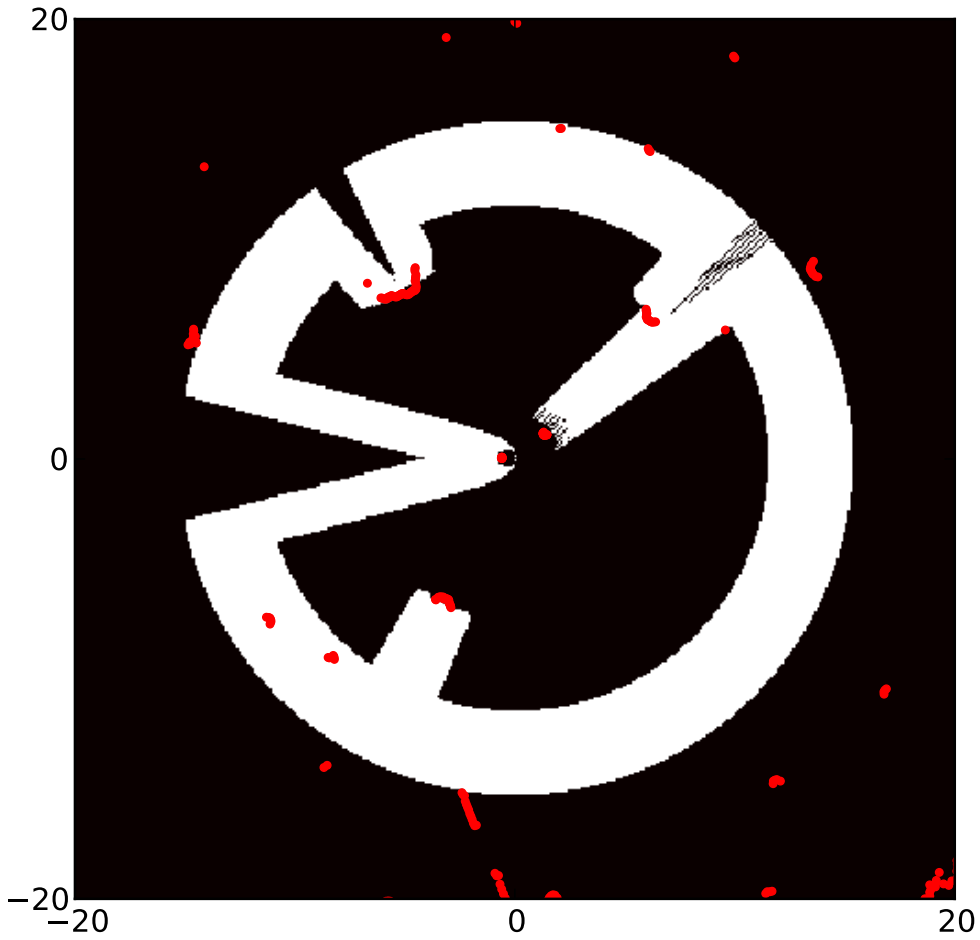

Figure 6.3: The shape of the buffer zone for a specific laser-scan and a feature radius of 0.1 meters can be seen in white.

filter assumes this type of distribution for the number of false alarms, therefore it must be used even if other distributions would be more appropriate to model the number of false alarms. To evaluate the goodness of fit of the Poisson distribution to the actual data, a Pearson's Chi Squared test [34] was used. This test rejected the hypothesis that the data comes from a Poisson distribution (with a p-value of 2.24×10^{-240} and a significance level of 0.05). Nevertheless, the Poisson distribution must be used in order to apply the PHD filter. The average number of false alarms N_{fa} per laser scan was estimated to be 6.18 false alarms per scan.

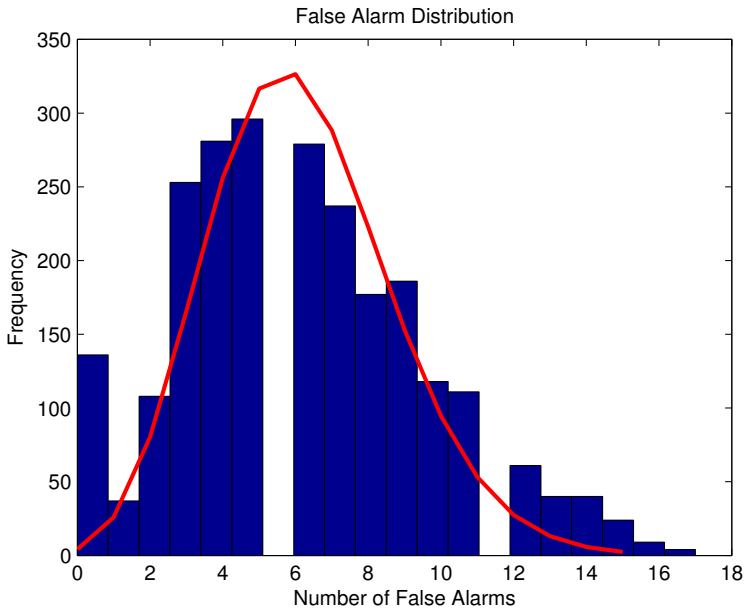

Figure 6.4: Histogram for the false alarms. A Poisson distribution is fitted to the data.

Chapter 7

Experimental Results

This chapter will show a comparison between RB-PHD-SLAM and a state of the art SLAM solution, Multi-Hypothesis FastSLAM [10]. Both algorithms use either a constant probability of detection, within the 40m radius FoV of the laser range finder, or the derived detection statistics, in order to determine the benefits of using the state dependent detection statistics.

The robotic platform for collecting the experimental dataset was a Clearpath Husky A-200 robot equipped with a Sick LD-LRS-1000 laser range finder. The Husky's wheel encoders provided odometry measurements, $\boldsymbol{u}_{0:k-1}$, for the motion model in Equation (3.6). The experiments were conducted in the same environment where the detection statistics were determined, albeit with a different dataset. This provides the best estimate of the real detection statistics that will be encountered in the experiments.

The environment used to conduct the experiments was Parque O'Higgins, located in Santiago de Chile's downtown area. Figure 7.1 shows the section of the park in which the experimental dataset was collected. The area of the park used had an abundance of trees, and the roads where the robot moved consisted of dirt tracks. The trees do not have perfectly circular trunks. A satellite image of the environment used can be seen in Figure 7.2.

Figures 7.3, 7.4, 7.7, and 7.8 show the performance of RB-PHD-SLAM and MH-FastSLAM. The map estimates are represented by plotting an ellipse representing the covariance (3-sigma ellipse) of each Gaussian in the GM, v_k^+ (Equation (3.9)), in the case of RB-PHD-SLAM. For MH-FastSLAM an ellipse is plotted for every feature in the state vector. Estimated vehicle trajectories are constructed using the current position for the particle $\boldsymbol{x}_k^i$ (Equation (3.7)) with the highest weight w_k (Equation (3.13)) at every time step. The red dashed line represents the dead reckoning trajectory while the red stars show the ground truth positions of landmarks determined using manually initialized ICP (as decribed in Section 5.3.3). A superior SLAM performance is indicated in Figures 7.3 and 7.7, in which the vehicle trajectories contain less errors than the assumed constant detection probability based results shown in Figures 7.4 and 7.8. The quality of the associated map estimates, in terms of cir-

Figure 7.1: Environment used to conduct the experiments.

Figure 7.2: Satellite image of the environment used to conduct the experiments.

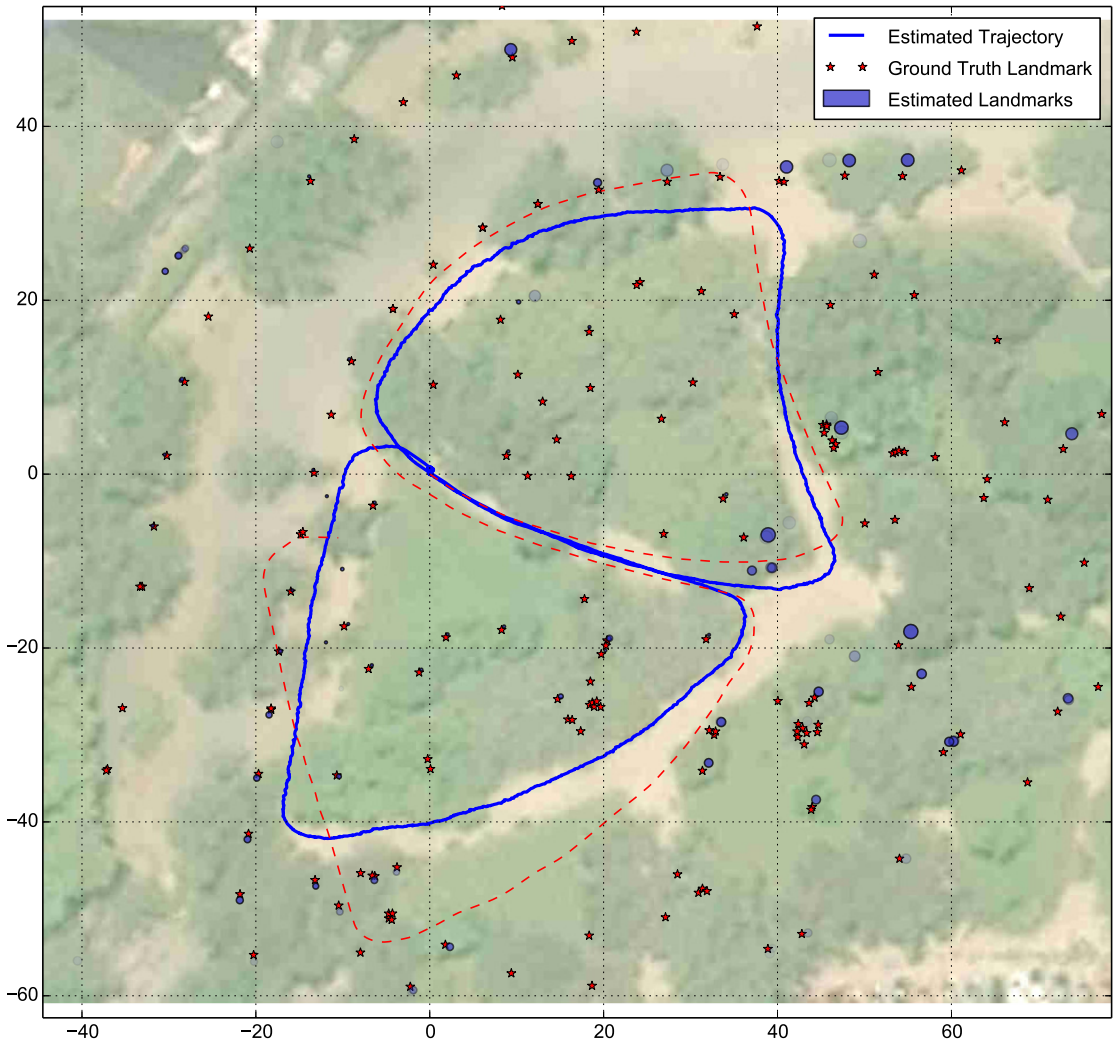

Figure 7.3: RB-PHD-SLAM results in a park environment. The vehicle traversed an approximate "figure eight" shape along the dirt track shown in the superimposed satellite images. The blue circles represent the spatial feature estimates, stars represent the ground truth map, the blue line is the estimated robot trajectory and the red dashed line corresponds to dead reckoning.

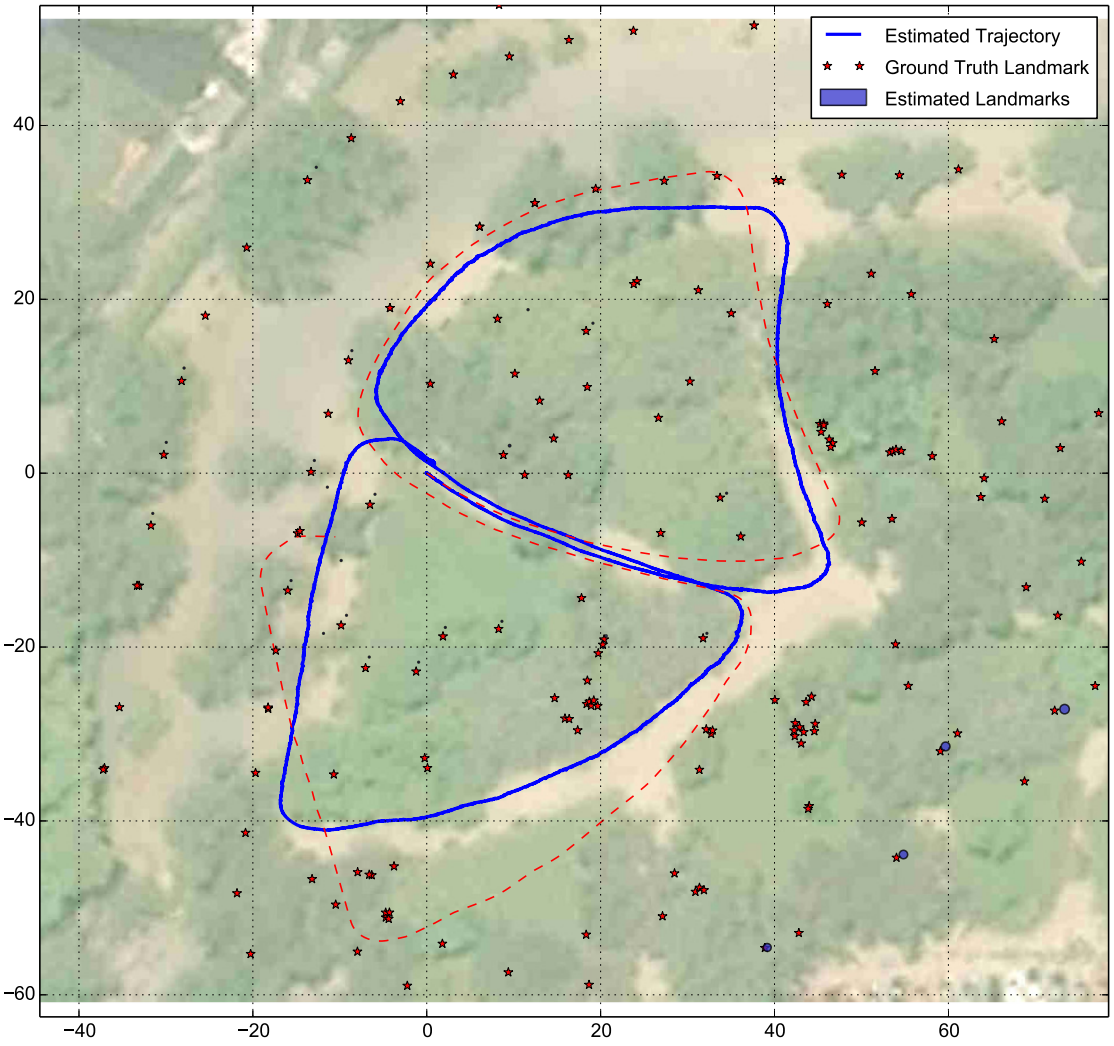

Figure 7.4: RB-PHD-SLAM results in a park environment with constant P_D.

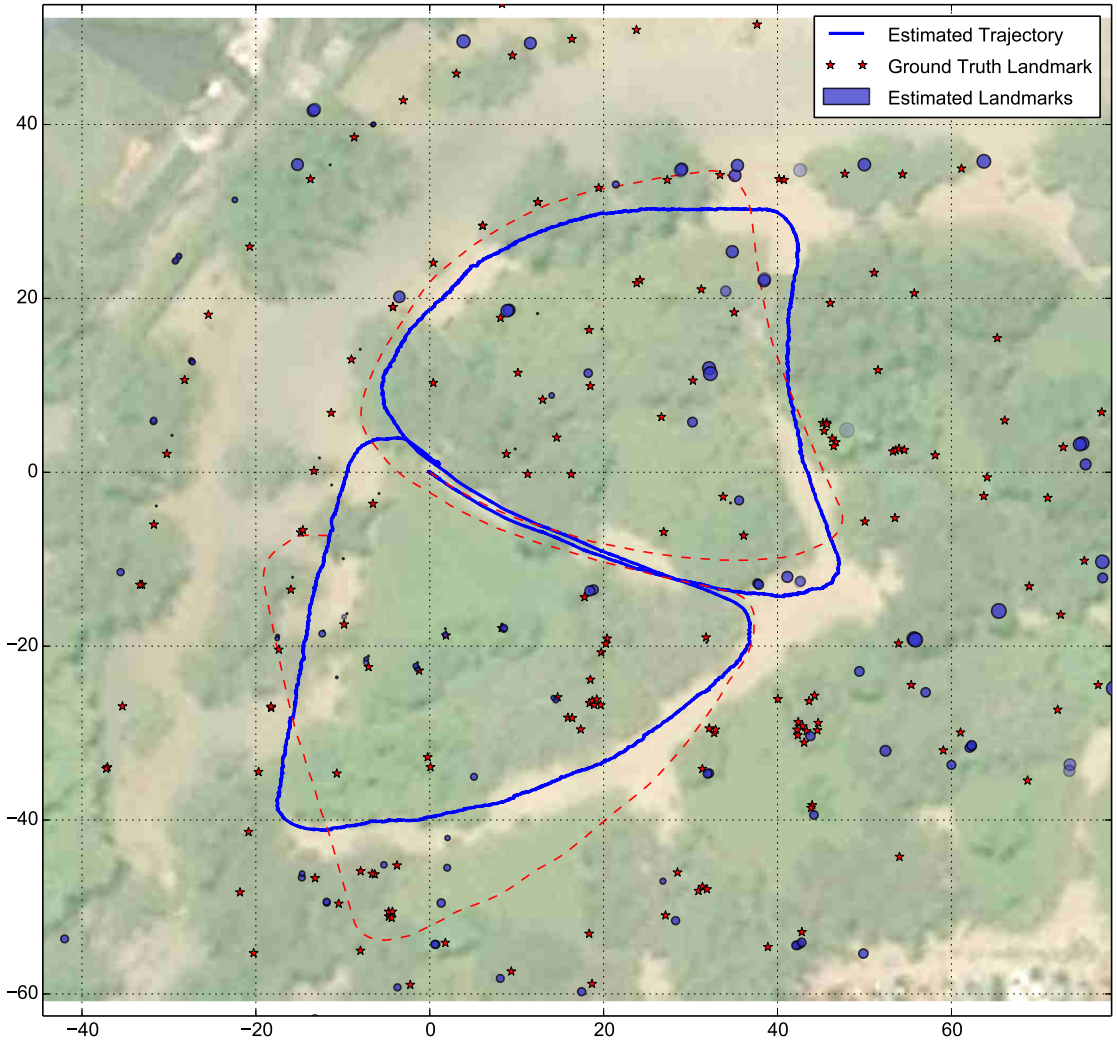

Figure 7.5: RB-PHD-SLAM results without using a bufferzone as described in Section 3.4.1

cular feature location and number, is also better in Figures 7.3 and 7.7 than in Figures 7.4 and 7.8. In the latter cases, the mismatch between the real and estimated probabilities of detection causes features to be removed from the map soon after they leave the FoV of the robot laser range finder. By comparing Figure 7.3 with Figure 7.7 it is possible to see that trajectory-wise both filters performed similarly well, while the maps differ significantly. The RB-PHD-SLAM filter has a much smaller map with very few false features. On the other hand, MH-FastSLAM generates a map with a lot more features several of them being false. In Figure 7.6 the results from using target amplitude feature are shown. When compared with the results in Figure 7.3, the performance is slightly decreased, both in the trajectory and map estimates. Figure 7.5 shows the performance of the RB-PHD-SLAM without using the buffer zone proposed in Section 3.4.1. As can be seen both the map and trajectory are less accurate compared to the proposed method from Figure 7.3.

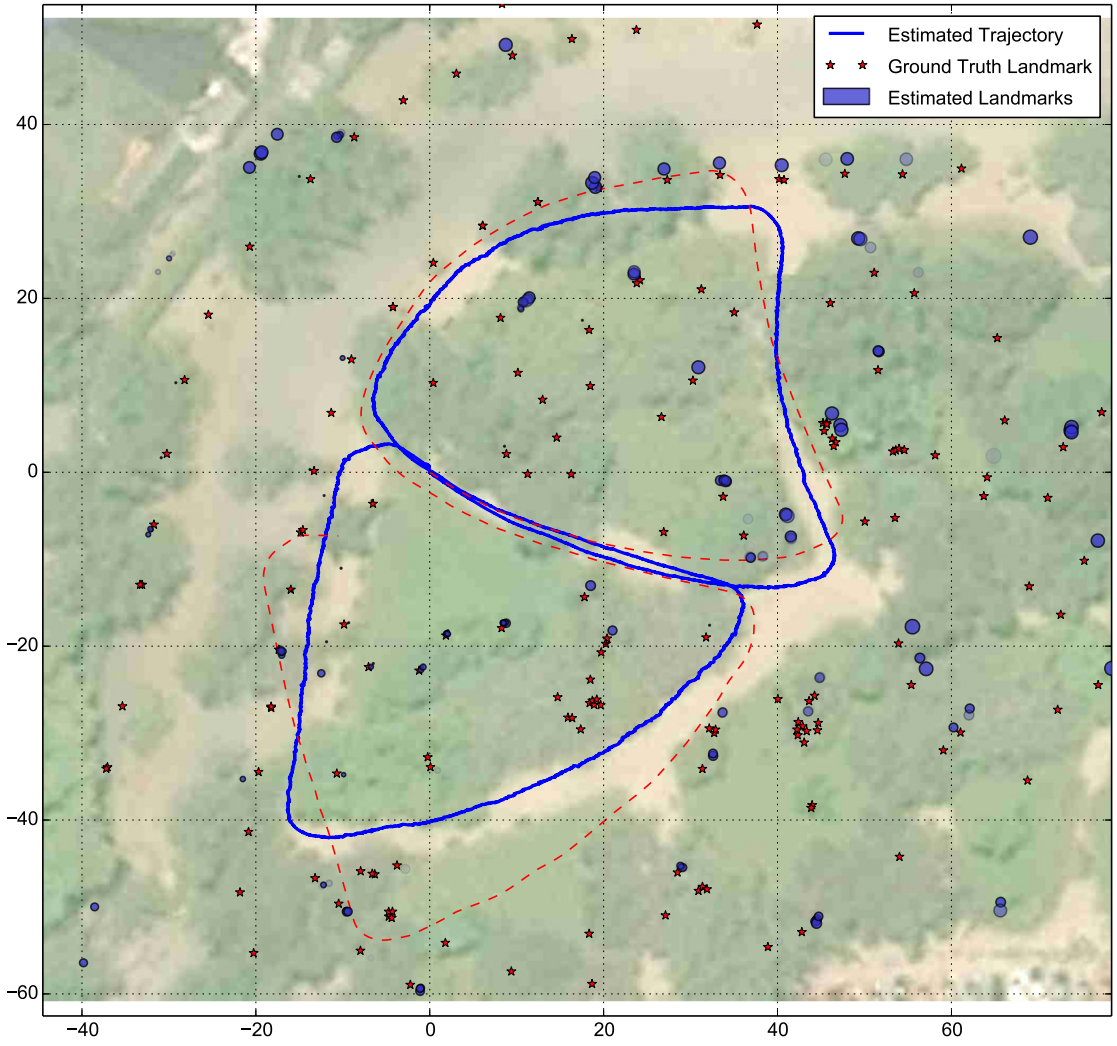

Figure 7.6: RB-PHD-SLAM results using target amplitude feature [18] as described in Section 5.3.3.

To evaluate the mapping results quantitatively the OSPA distance was used, the OSPA distance has been used in the multi-target tracking community as good measure of the distance between two sets, taking into account both spatial and cardinality error [35]. It is important to note that the ground truth map contains several tree trunks that have a small diameter and were detected either never or very few times, therefore it is expected for all SLAM algorithms to have a high mapping error due to the tree trunks that were never detected. Figure 7.9 shows the OSPA distance for each of the algorithms described above. Both RB-PHD-SLAM and MH-FastSLAM perform considerably worse when using a constant value for the probability of detection, with MH-FastSLAM outperforming RB-PHD-SLAM in this case. When using the proposed model for P_D the error of both algorithms are considerably lower, with MH-FastSLAM, RB-PHD-SLAM (without using a buffer zone), and RB-PHD-SLAM using target amplitude feature having very similar errors. Finally the use

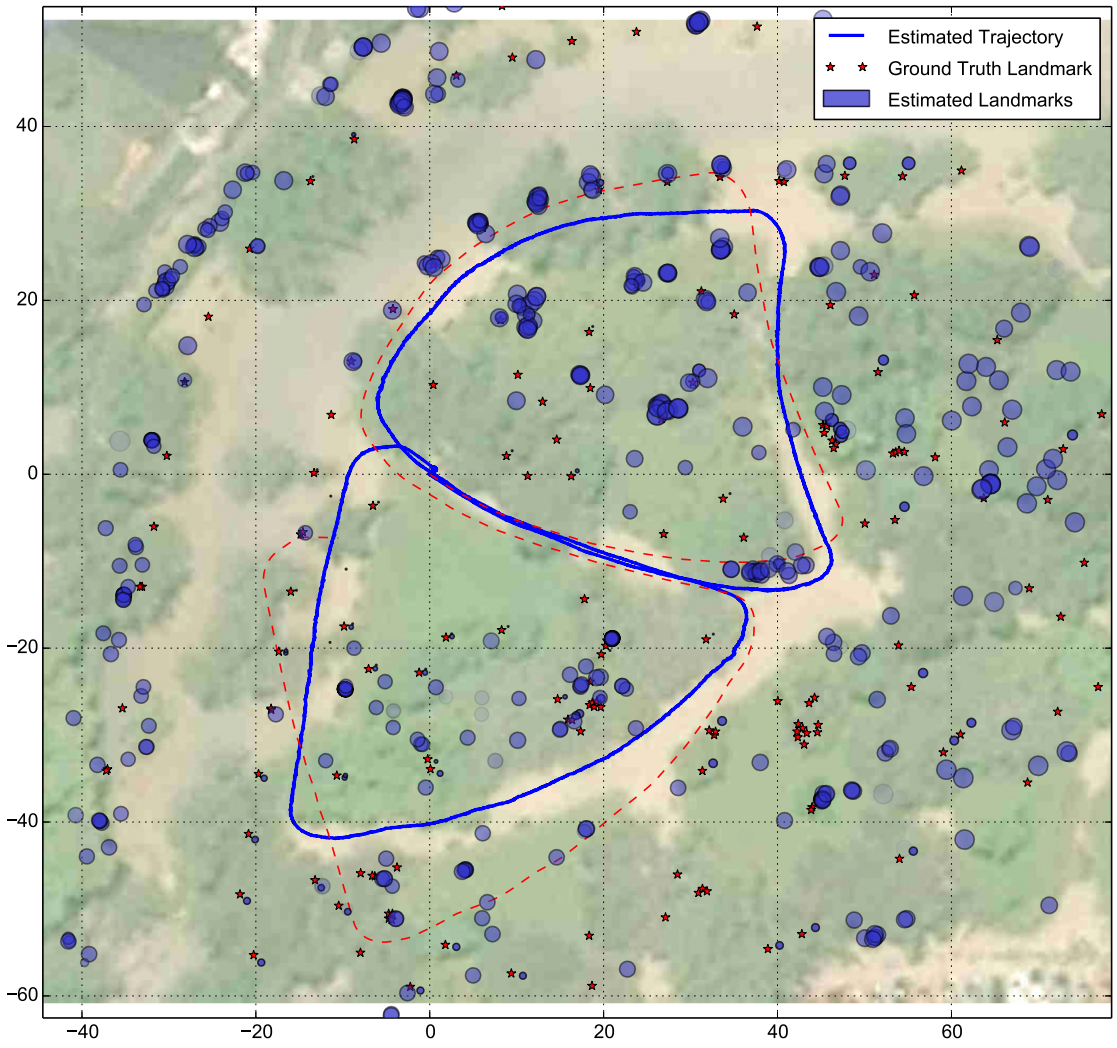

Figure 7.7: MH-FastSLAM results in a park environment.

of a buffer zone lowers the error of RB-PHD-SLAM even further and slightly outperforms MH-FastSLAM.

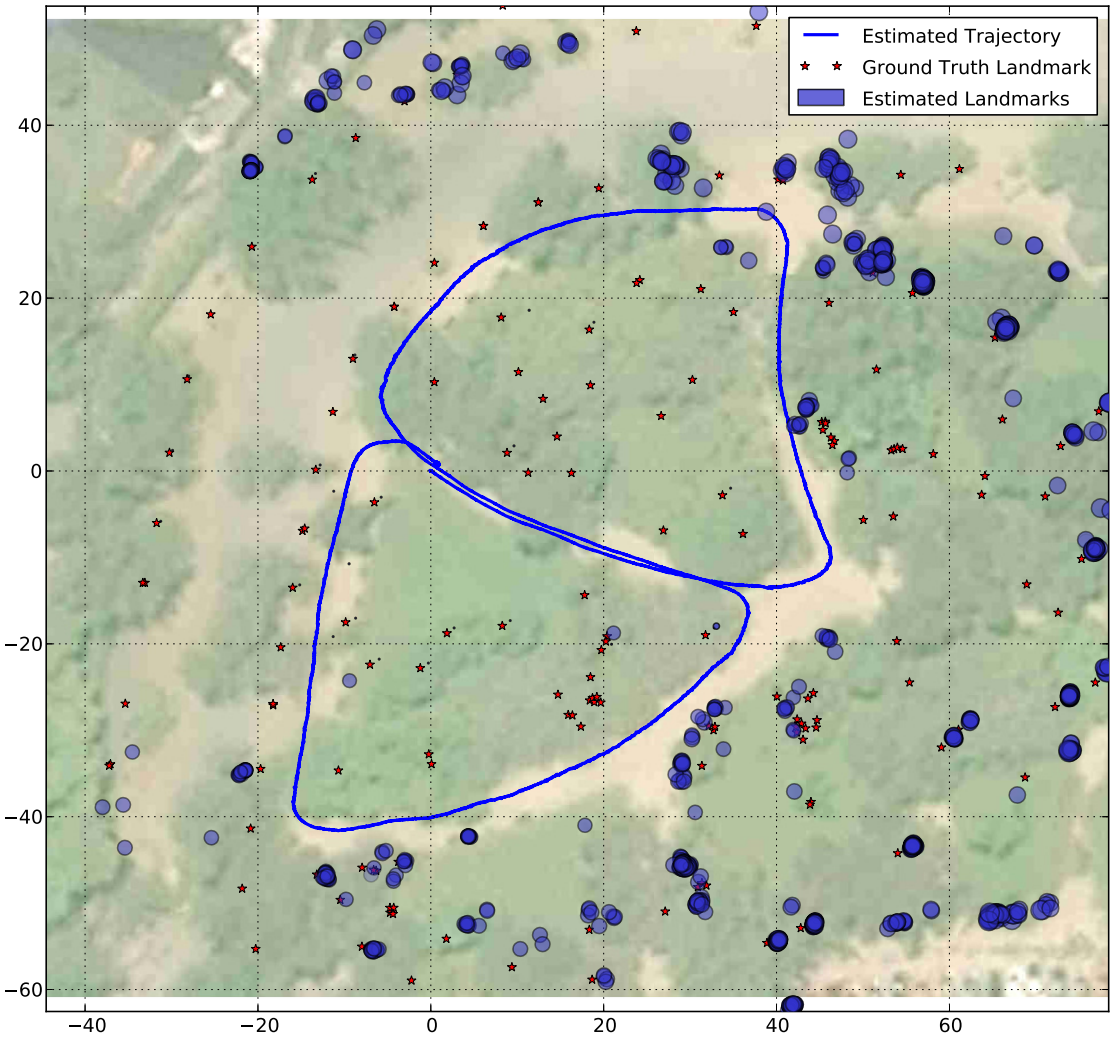

Figure 7.8: MH-FastSLAM results in a park environment with constant P_D.

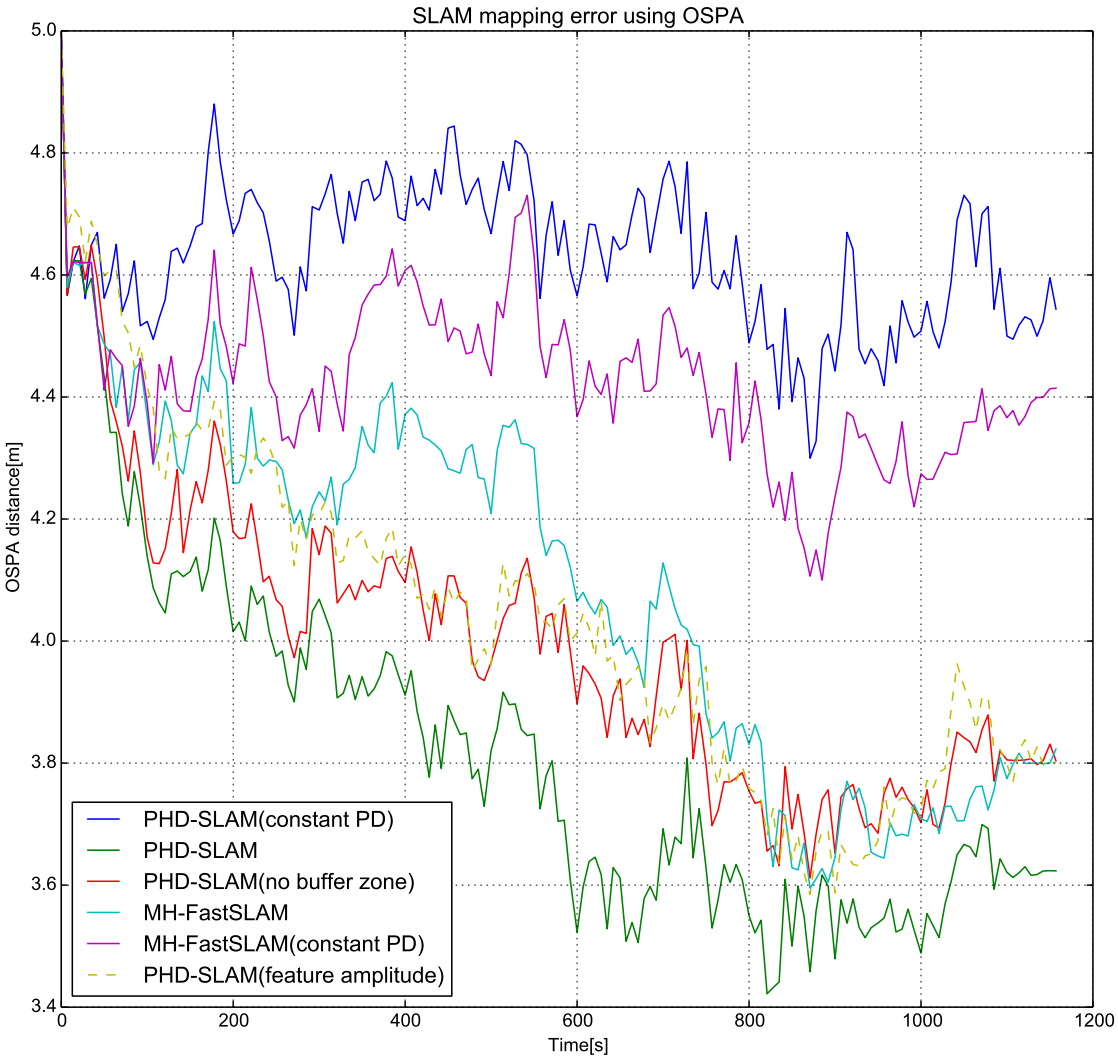

Figure 7.9: Mapping error of the slam solutions measured using the OSPA distance.

Chapter 8

Conclusions

The importance of modelling detection statistics within real SLAM experiments has been highlighted in this Thesis. A simple semantic feature based detector was presented, together with detection theoretic based methods for the evaluation of each feature's probabilities of detection and false alarm. The methods presented can be used with any feature detector that estimates the shape of an object. The techniques were applied to a simple circle detector, for use in environments in which multiple circular cross sectioned features were expected.

The derived detection statistics were used in the RB-PHD-SLAM as well as MH-FastSLAM frameworks, in a park environment, in which the primary circular sectioned features were trees. The results demonstrated superior SLAM estimates for both frameworks, in terms of vehicle trajectory, feature number and location estimates (measured visually and using the OSPA distance) when the derived probability of detection model was compared with a typical implementation which assumed constant detection statistics within the sensor FoV. The introduction of the proposed detection statistics into both MH-FastSLAM and RB-PHD-SLAM decreased the OSPA distance between the map estimate and the ground truth map by 13% and 17%, respectively. A clear advantage of using RB-PHD-SLAM was expected but not found. A buffer zone in which Gaussians are prevented from diminishing their weights was implemented to handle the edge of the FoV in the RB-PHD-SLAM algorithm, and was shown to increase the performance of the filter, further decreasing the OSPA distance by 5%. The target amplitude feature PHD filter was also implemented, but the amplitude information added to the filter proved unhelpful. This could be because the chosen descriptor vector distributions are not sufficiently different between true detections and false alarms within the detection volume, unlike amplitude values for radar based feature detectors, in which measurements with higher amplitudes are less likely to be clutter. The author expects to obtain better results by applying newer RFS-Based filters such as the Cardinality Balanced Multi-Target Multi-Bernoulli filter [14].

Glossary

CB-MemBer cardinality balanced multi-target multi-Bernoulli. 9, 20

CFAR constant false alarm rate. 2

CPHD cardinalized PHD. 9, 18

EIF extended information filter. 6

EKF extended Kalman filter. 5, 6, 14, 17

FastSLAM factored solution to SLAM. v, vii, viii, 2, 6, 7, 42, 46–49, 51

FISST finite set statistics. 2

FoV field of view. 1, 2, 11, 17, 20, 21, 26, 37–39, 42, 46, 51

GM Gaussian mixture. 10, 17, 19, 42

ICP iterative closest point. 31, 42

KF Kalman filter. 17

MH multiple hypothesis. v, vii, viii, 2, 7, 42, 46–49, 51

OSPA optimal sub-pattern assignment. v, 12, 47, 50, 51

PDF probability density function. 5, 13–15

PF particle filter. 6, 14

PHD probability hypothesis density. iii, v, vii, viii, 2, 9–12, 14–20, 24–26, 34, 38–40, 42, 44–48, 51, 52

RANSAC random sample and consensus. 28

RB Rao-Blackwellized. iii, v, vii, viii, 2, 6, 14, 16, 18, 25, 34, 42, 44–48, 51

RFS random finite set. iii, vii, viii, 1–5, 8, 9, 11, 13, 16, 18, 20, 23–25, 28, 37, 38, 51

ROC receiver operating characteristic. v, 32, 35

SEIF sparse EIF. 6

SLAM simultaneous localization and mapping. iii, v, vii, viii, 1–8, 10–14, 16, 18–21, 23, 25, 27–30, 34, 37, 38, 42, 44–48, 51

Bibliography

[1] H. Durrant-Whyte and T. Bailey. Simultaneous localization and mapping: part I. *Robotics & Automation Magazine, IEEE*, 13(2):99–110, 2006.

[2] J. Mullane, B.-N. Vo, M. D. Adams, and B.-T. Vo. A random-finite-set approach to Bayesian SLAM. *Robotics, IEEE Transactions on*, 27(2):268–282, 2011.

[3] John Mullane, Ba-Ngu Vo, Martin Adams, and Ba-Tuong Vo. *Random Finite Sets for Robot Mapping and SLAM - New Concepts in Autonomous Robotic Map Representations*, volume 72 of *Springer Tracts in Advanced Robotics*. Springer, 2011. ISBN 978-3-642-21389-2. URL http://dblp.uni-trier.de/db/series/star/index.html#MullaneVAV11.

[4] M. W M G Dissanayake, P. Newman, S. Clark, H.F. Durrant-Whyte, and M. Csorba. A solution to the simultaneous localization and map building (SLAM) problem. *Robotics and Automation, IEEE Transactions on*, 17(3):229–241, Jun 2001. ISSN 1042-296X. doi: 10.1109/70.938381.

[5] Kwang Wee Lee, W Sardha Wijesoma, and Javier Ibanez Guzman. On the observability and observability analysis of SLAM. In *Intelligent Robots and Systems, 2006 IEEE/RSJ International Conference on*, pages 3569–3574. IEEE, 2006.

[6] Zhan Wang and Gamini Dissanayake. Observability analysis of SLAM using Fisher information matrix. In *Control, Automation, Robotics and Vision, 2008. ICARCV 2008. 10th International Conference on*, pages 1242–1247. IEEE, 2008.

[7] Sebastian Thrun, Yufeng Liu, Daphne Koller, Andrew Y Ng, Zoubin Ghahramani, and Hugh Durrant-Whyte. Simultaneous localization and mapping with sparse extended information filters. *The International Journal of Robotics Research*, 23(7-8):693–716, 2004.

[8] M. Montemerlo, S. Thrun, D. Koller, B. Wegbreit, et al. FastSLAM: A factored solution to the simultaneous localization and mapping problem. In *Proceedings of the National conference on Artificial Intelligence*, pages 593–598, 2002.

[9] D Roller, M Montemerlo, S Thrun, and B Wegbreit. FastSLAM 2.0: an improved particle filtering algorithm for simultaneous localization and mapping that provably converges. In *Proceedings of the International Joint Conference on Artificial Intelligence*, 2003.

[10] Juan Nieto, Jose Guivant, Eduardo Nebot, and Sebastian Thrun. Real time data association for fastSLAM. In *Proceedings of IEEE International Conference on Robotics and Automation (ICRA)*, volume 1, pages 412–418. IEEE, 2003.

[11] Sebastian Thrun and Michael Montemerlo. The graph SLAM algorithm with applications to large-scale mapping of urban structures. *The International Journal of Robotics Research*, 25(5-6):403–429, 2006.

[12] J Mullane, B-N Vo, and M. D. Adams. Rao-Blackwellised PHD SLAM. In *Proceedings of IEEE International Conference on Robotics and Automation (ICRA)*, pages 5410–5416, 2010.

[13] R. P. S. Mahler. *Statistical multisource-multitarget information fusion*, volume 685. Artech House Boston, 2007.

[14] Ba-Tuong Vo, Ba-Ngu Vo, and Antonio Cantoni. The cardinality balanced multi-target multi-Bernoulli filter and its implementations. *Signal Processing, IEEE Transactions on*, 57(2):409–423, 2009.

[15] Sebastian Thrun, Wolfram Burgard, and Dieter Fox. *Probabilistic robotics*, volume 1. MIT press Cambridge, 2005.

[16] R. Mahler. A survey of PHD filter and CPHD filter implementations. In *Proc. SPIE Defense & Security Symposium of Signal Processing, Sensor Fusion and Target Recognition XII*, April 2007.

[17] Karl Granstrom and Umut Orguner. A PHD filter for tracking multiple extended targets using random matrices. *Signal Processing, IEEE Transactions on*, 60(11):5657–5671, 2012.

[18] D. Clark, B. Ristic, Ba-Ngu Vo, and Ba-Tuong Vo. Bayesian multi-object filtering with amplitude feature likelihood for unknown object SNR. *Signal Processing, IEEE Transactions on*, 58(1):26–37, Jan 2010. ISSN 1053-587X. doi: 10.1109/TSP.2009.2030640.

[19] I Skolnik Merrill. Introduction to radar systems. *Mc Grow-Hill*, 2001.

[20] S. Kay. *Fundamentals of Statistical Signal Processing, Vol I - Estimation Theory*. Prentice Hall, 1993.

[21] K. Y. K. Leung, F. Inostroza, and M Adams. An improved weighting strategy for Rao-Blackwellized probability hypothesis density simultaneous localization and mapping. In *Proceedings of International Conference on Control, Automation, and Information Sciences*, 2013.

[22] Ba-Ngu Vo and Wing-Kin Ma. The Gaussian mixture probability hypothesis density filter. *Signal Processing, IEEE Transactions on*, 54(11):4091–4104, 2006.

[23] Johann Wolfgang Koch. Bayesian approach to extended object and cluster tracking using random matrices. *Aerospace and Electronic Systems, IEEE Transactions on*, 44 (3):1042–1059, 2008.

[24] Andrew Gelman, John B Carlin, Hal S Stern, and Donald B Rubin. *Bayesian Data Analysis, (Chapman & Hall/CRC Texts in Statistical Science)*. Chapman and Hall/CRC, 2003.

[25] F. Lu and E. Milios. Robot pose estimation in unknown environments by matching 2D range scans. *Journal of Intelligent and Robotic Systems*, 18(3):249–275, 1997.

[26] V. Nguyen, A. Martinelli, N. Tomatis, and R. Siegwart. A comparison of line extraction algorithms using 2D laser rangefinder for indoor mobile robotics. In *Intelligent Robots and Systems, 2005. (IROS 2005). 2005 IEEE/RSJ International Conference on*, pages 1929–1934, Aug 2005. doi: 10.1109/IROS.2005.1545234.

[27] P. Núñez, R. Vázquez-Martín, J.C. del Toro, A. Bandera, and F. Sandoval. Natural landmark extraction for mobile robot navigation based on an adaptive curvature estimation. *Robotics and Autonomous Systems*, 56(3):247 – 264, 2008. ISSN 0921-8890. doi: http://dx.doi.org/10.1016/j.robot.2007.07.005. URL http://www.sciencedirect.com/science/article/pii/S0921889007000942.

[28] Steven L Horowitz and Theodosios Pavlidis. Picture segmentation by a tree traversal algorithm. *Journal of the ACM (JACM)*, 23(2):368–388, 1976.

[29] Sen Zhang, M Adams, Fan Tang, and Lihua Xie. Geometrical feature extraction using 2D range scanner. In *Control and Automation, 2003. ICCA'03. Proceedings. 4th International Conference on*, pages 901–905. IEEE, 2003.

[30] Jorge J Moré. The Levenberg-Marquardt algorithm: implementation and theory. In *Numerical analysis*, pages 105–116. Springer, 1978.

[31] S. Kay. *Fundamentals of Statistical Signal Processing, Vol II - Detection Theory*. Prentice Hall, 1998.

[32] P.J. Besl and Neil D. McKay. A method for registration of 3-D shapes. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 14(2):239–256, Feb 1992. ISSN 0162-8828. doi: 10.1109/34.121791.

[33] Ronald A Fisher. The statistical utilization of multiple measurements. *Annals of eugenics*, 8(4):376–386, 1938.

[34] Karl Pearson. X. on the criterion that a given system of deviations from the probable in the case of a correlated system of variables is such that it can be reasonably supposed to have arisen from random sampling. *The London, Edinburgh, and Dublin Philosophical Magazine and Journal of Science*, 50(302):157–175, 1900.

[35] Dominic Schuhmacher, Ba-Tuong Vo, and Ba-Ngu Vo. A consistent metric for performance evaluation of multi-object filters. *Signal Processing, IEEE Transactions on*, 56 (8):3447–3457, 2008.

Appendix A

The Gaussian inverse Wishart PHD filter

In this appendix the specific equations used in of the Gaussian inverse Wishart PHD filter of [17] are shown. As explained in section 4.1.1 the prediction step of the filter is:

$$v_{k+1}^{-}(\boldsymbol{m}) = \sum_{i} \omega_{k+1}^{i-} \mathcal{N}\left(\boldsymbol{x} \middle| \boldsymbol{\mu}_{k+1}^{i-}, \boldsymbol{\Sigma}_{k-1}^{i-} \otimes \boldsymbol{X}\right) \mathcal{W}^{-1}\left(\boldsymbol{X} \middle| \boldsymbol{\nu}_{k+1}^{i-}, \boldsymbol{\Psi}_{k+1}^{i-}\right) \tag{A.1}$$

where the values of the parameters of the predicted Gaussian inverse Wishart distributions are calculated according to the process model

$$\boldsymbol{\mu}_{k+1}^{i-} = \left(F_{k+1|k} \otimes \boldsymbol{I}_d\right) \boldsymbol{\mu}_k^i \tag{A.2}$$

$$\boldsymbol{\Sigma}_{k-1}^{i-} = F_{k+1|k} \boldsymbol{\Sigma}_k^i F_{k+1|k}^T + \boldsymbol{Q}_{k+1|k} \tag{A.3}$$

$$\boldsymbol{\nu}_{k+1}^{i-} = e^{-T_s/\tau} \boldsymbol{\nu}_k^i \tag{A.4}$$

$$\boldsymbol{\Psi}_{k+1}^{i-} = \frac{\boldsymbol{\nu}_{k+1}^i - d - 1}{\boldsymbol{\nu}_k^i - d - 1} \boldsymbol{\Psi}_k^i \tag{A.5}$$

As can be seen from the equations the Gaussian part of the prediction step is identical to the Kalman filter prediction step, while the inverse Wishart part proposed is an increase in uncertainty, according to the decay constant τ, with no changes to the mean of the distribution. The update step in the filter is

$$\begin{aligned} v_{k+1}^{+}(\boldsymbol{m}) = & \left(1 - \left(1 - e^{-\gamma(\boldsymbol{m})}\right) P_D(\boldsymbol{m})\right) v_{k+1}^{-}(\boldsymbol{m}) \\ & + \sum_{p \angle \mathcal{Z}_k} \sum_{\mathcal{U} \in p} \sum_{i} \omega_{k+1}^{i,\mathcal{U}} \mathcal{N}\left(\boldsymbol{x} \middle| \boldsymbol{\mu}_{k+1}^{i,\mathcal{U}}, \boldsymbol{\Sigma}_{k-1}^{i,\mathcal{U}} \otimes \boldsymbol{X}\right) \mathcal{W}^{-1}\left(\boldsymbol{X} \middle| \boldsymbol{\nu}_{k+1}^{i,\mathcal{U}}, \boldsymbol{\Psi}_{k+1}^{i,\mathcal{U}}\right) \end{aligned} \tag{A.6}$$

where $p \angle \mathcal{Z}_k$ means that p is a partition of the measurement set and the summation is over all possible partitions, $\mathcal{U} \in p$ means that $\mathcal{U}$ is one of the non-empty elements of p,

and the updated parameters are calculated from the previous parameters using the measurement model and the probability of detection $P_D(\boldsymbol{m})$, clutter rate κ and average number of measurements per feature $\gamma(\boldsymbol{m})$:

$$\boldsymbol{\mu}_{k+1}^{i,\mathcal{U}} = \boldsymbol{\mu}_{k+1}^{i-} + \left(\boldsymbol{K}^{i,\mathcal{U}} \otimes \boldsymbol{I}_d\right) \epsilon^{i,\mathcal{U}} \tag{A.7a}$$

$$\boldsymbol{\Sigma}_{k-1}^{i,\mathcal{U}} = \boldsymbol{\Sigma}_{k-1}^{i-} - \boldsymbol{K}^{i,\mathcal{U}} \boldsymbol{S}^{i,\mathcal{U}} \left(\boldsymbol{K}^{i,\mathcal{U}}\right)^T \tag{A.7b}$$

$$\boldsymbol{\nu}_{k+1}^{i,\mathcal{U}} = \boldsymbol{\nu}_{k+1}^{i-} + |\mathcal{U}| \tag{A.7c}$$

$$\boldsymbol{\Psi}_{k+1}^{i,\mathcal{U}} = \boldsymbol{\Psi}_{k+1}^{i-} + \boldsymbol{N}^{i,\mathcal{U}} + \boldsymbol{G}^{[\mathcal{U}]} \tag{A.7d}$$

$$\omega_{k+1}^{i,\mathcal{U}} = \frac{\omega_p}{d\mathcal{U}} e^{-\gamma(\boldsymbol{\mu}_k^i)} \left(\frac{\gamma(\boldsymbol{\mu}_{k+1}^{i-})}{\kappa}\right)^{|\mathcal{U}|} P_D(\boldsymbol{\mu}_{k+1}^{i-}) \mathcal{L}^{i,\mathcal{U}} \omega_{k+1}^{i-} \tag{A.7e}$$

where

$$\bar{\boldsymbol{\Psi}}_k^{[\mathcal{U}]} = \frac{1}{|\mathcal{U}|} \sum_{\boldsymbol{\Psi}_k^i \in \mathcal{U}} \boldsymbol{\Psi}_k^i \tag{A.8a}$$

$$\boldsymbol{G}^{[\mathcal{U}]} = \sum_{\boldsymbol{\Psi}_k^i \in \mathcal{U}} (\boldsymbol{\Psi}_k^i - \bar{\boldsymbol{\Psi}}_k^{[\mathcal{U}]})(\boldsymbol{\Psi}_k^i - \bar{\boldsymbol{\Psi}}_k^{[\mathcal{U}]})^T \tag{A.8b}$$

$$\boldsymbol{S}^{i,\mathcal{U}} = \boldsymbol{H}_k \boldsymbol{\Sigma}_{k-1}^{i-} \boldsymbol{H}_k^T + \frac{1}{|\mathcal{U}|} \tag{A.8c}$$

$$\boldsymbol{K}^{i,\mathcal{U}} = \boldsymbol{\Sigma}_{k-1}^{i-} \boldsymbol{H}_k^T \left(\boldsymbol{S}^{i,\mathcal{U}}\right)^{-1} \tag{A.8d}$$

$$\epsilon^{i,\mathcal{U}} = \bar{\boldsymbol{\Psi}}_k^{[\mathcal{U}]} \left(\boldsymbol{H}_k \otimes \boldsymbol{I}_d\right) \boldsymbol{\mu}_{k+1}^{i-} \tag{A.8e}$$

$$\boldsymbol{N}^{i,\mathcal{U}} = \left(\boldsymbol{S}^{i,\mathcal{U}}\right)^{-1} \epsilon^{i,\mathcal{U}} \left(\epsilon^{i,\mathcal{U}}\right)^T \tag{A.8f}$$

$$\omega_p = \frac{\prod_{\mathcal{U} \in p} d\mathcal{U}}{\sum_{p' \angle \mathcal{Z}_k} \prod_{\mathcal{U}' \in p'} d\mathcal{U}'} \tag{A.8g}$$

$$d\mathcal{U} = \delta_{|\mathcal{U}|,1} + \sum_i e^{-\gamma(\boldsymbol{\mu}_k^i)} \left(\frac{\gamma(\boldsymbol{\mu}_{k+1}^{i-})}{\kappa}\right)^{|\mathcal{U}|} P_D(\boldsymbol{\mu}_{k+1}^{i-}) \mathcal{L}^{i,\mathcal{U}} \omega_{k+1}^{i-} \tag{A.8h}$$

$$\mathcal{L}^{i,\mathcal{U}} = \frac{1}{\left(\pi^{|\mathcal{U}|} |\mathcal{U}| \boldsymbol{S}^{i,\mathcal{U}}\right)^{\frac{d}{2}}} \frac{|\boldsymbol{\Psi}_{k+1}^{i-}|^{\frac{\boldsymbol{\nu}_{k+1}^{i-}}{2}} \Gamma_d\left(\frac{\boldsymbol{\nu}_{k+1}^{i,\mathcal{U}}}{2}\right)}{|\boldsymbol{\Psi}_{k+1}^{i,\mathcal{U}}|^{\frac{\boldsymbol{\nu}_{k+1}^{i,\mathcal{U}}}{2}} \Gamma_d\left(\frac{\boldsymbol{\nu}_{k+1}^{i-}}{2}\right)} \tag{A.8i}$$

define the parameters of equation A.7.

Appendix B

Monte Carlo Runs

In this appendix all the Monte Carlo runs that were computed for this thesis are shown, the most representative run of each set is shown in chapter 7.

B.1 PHD Filter

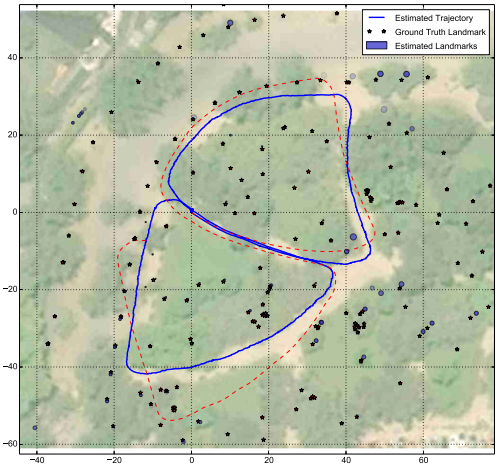

Figure B.1: PHD Filter result 1.

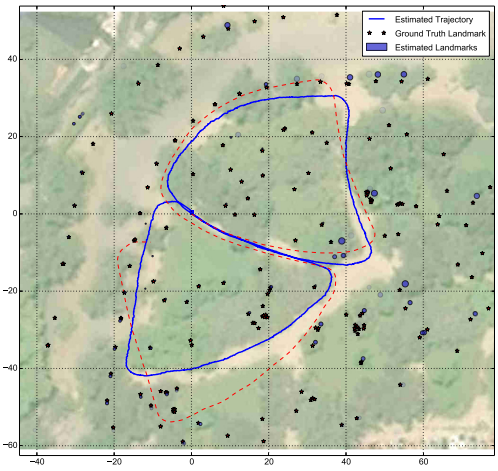

Figure B.2: PHD Filter result 2.

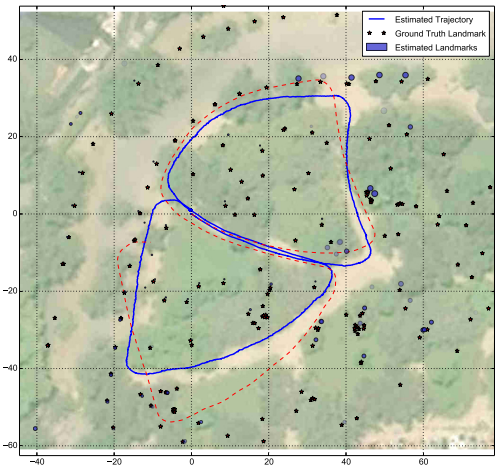

Figure B.3: PHD Filter result 3.

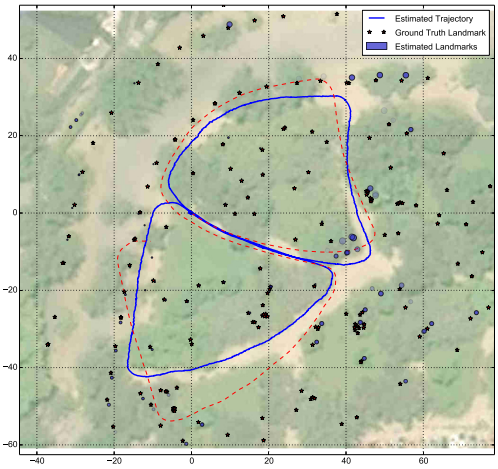

Figure B.4: PHD Filter result 4.

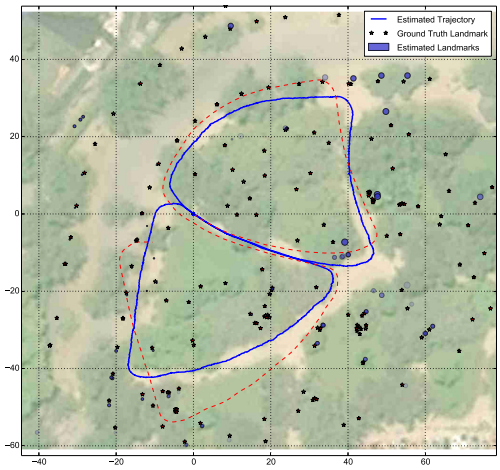

Figure B.5: PHD Filter result 5.

B.2 MH-FastSLAM

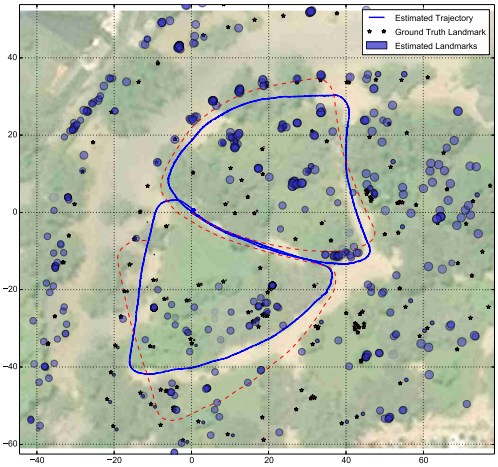

Figure B.6: MH-FastSLAM result 1.

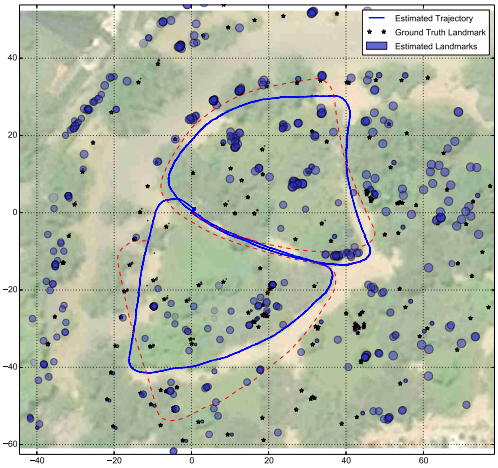

Figure B.7: MH-FastSLAM result 2.

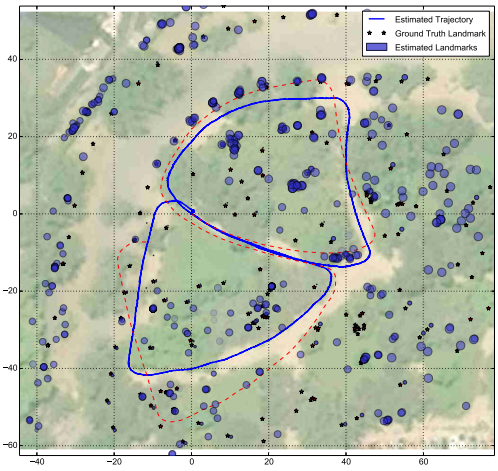

Figure B.8: MH-FastSLAM result 3.

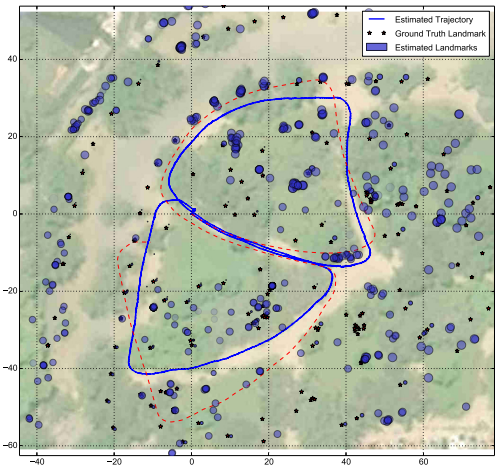

Figure B.9: MH-FastSLAM result 4.

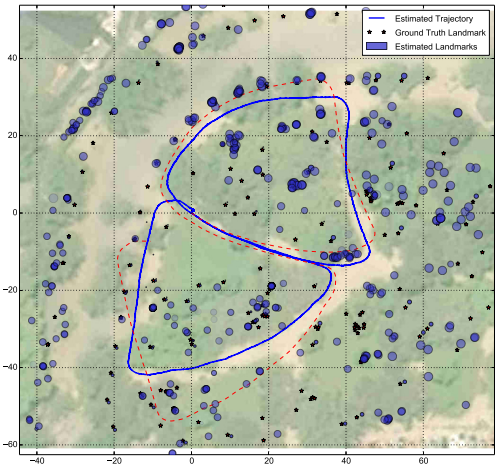

Figure B.10: MH-FastSLAM result 5.

B.3 PHD Filter without a buffer zone

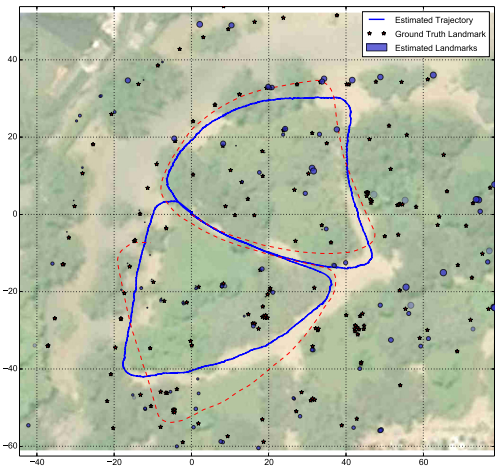

Figure B.11: PHD Filter with no buffer zone result 1.

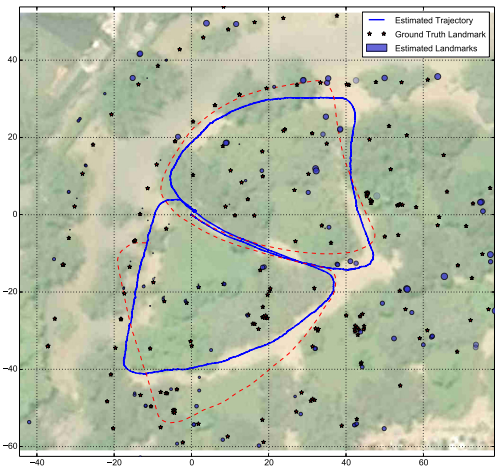

Figure B.12: PHD Filter with no buffer zone result 2.

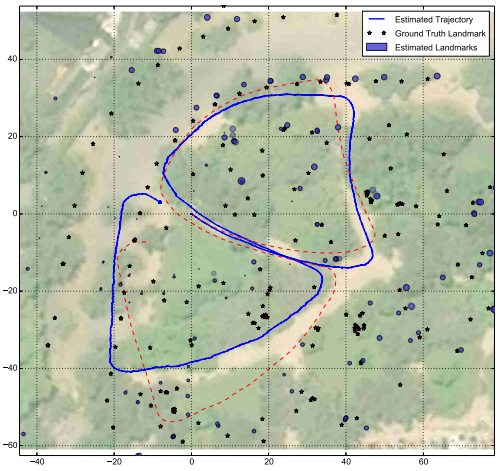

Figure B.13: PHD Filter with no buffer zone result 3.

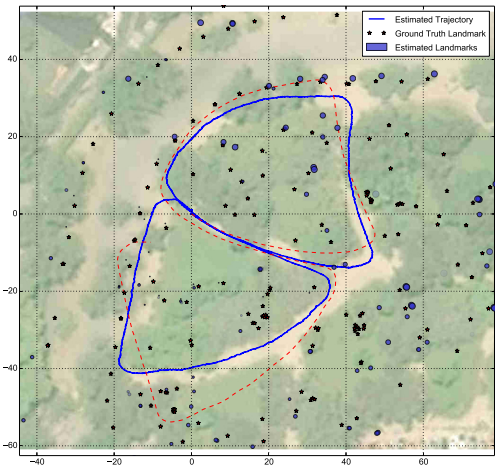

Figure B.14: PHD Filter with no buffer zone result 4.

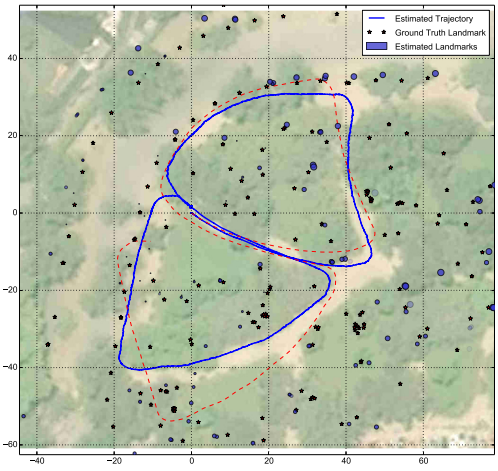

Figure B.15: PHD Filter with no buffer zone result 5.

B.4 PHD Filter with Constant P_D

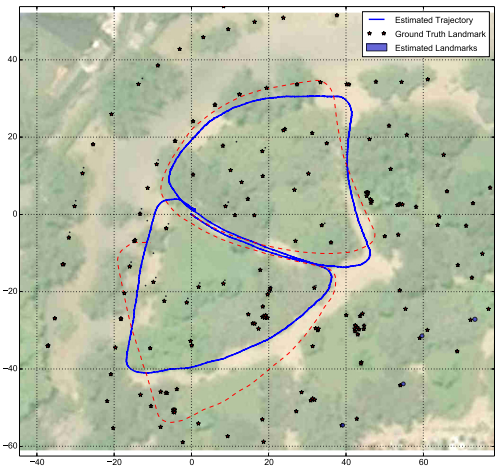

Figure B.16: PHD Filter with constant P_D result 1.

B.5 PHD Filter with amplitude feature

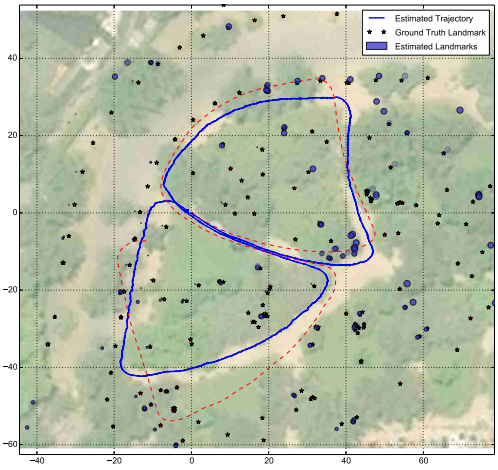

Figure B.17: PHD Filter with constant P_D result 1.

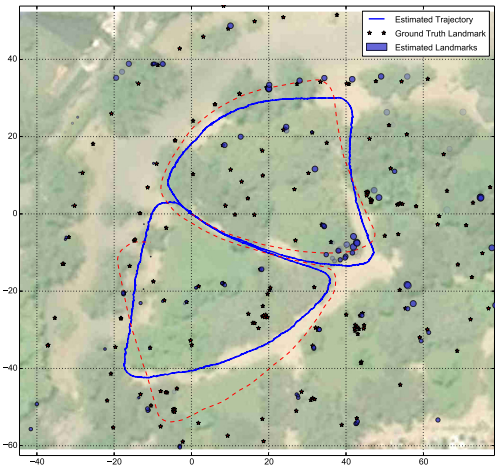

Figure B.18: PHD Filter with constant P_D result 2.

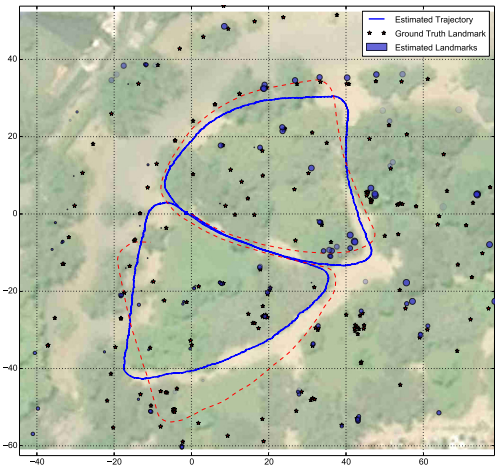

Figure B.19: PHD Filter with constant P_D result 3.

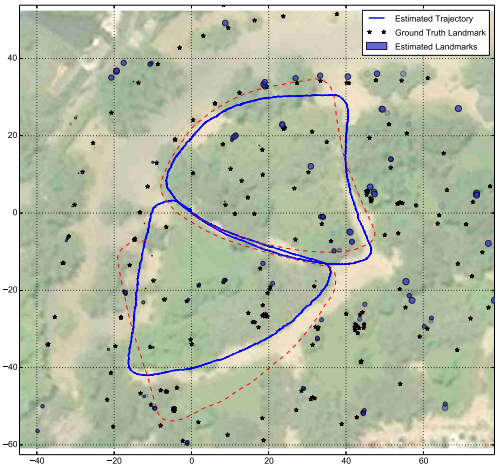

Figure B.20: PHD Filter with constant P_D result 4.

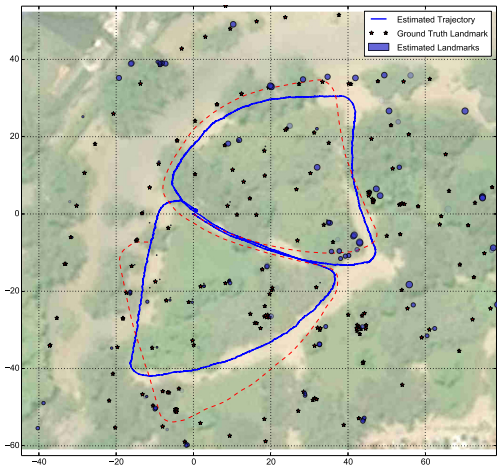

Figure B.21: PHD Filter with constant P_D result 5.

B.6 MH-FastSLAM with Constant P_D

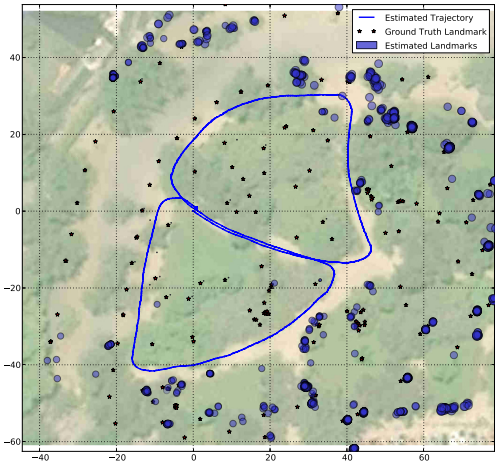

Figure B.22: MH-FastSLAM with constant P_D result 1.