Modeling Detection Statistics in Feature-Based Robotic Navigation for Range Sensors

FELIPE INOSTROZA and MARTIN ADAMS
Advanced Mining Technology Center, Dept. of Electrical Engineering, Universidad de Chile, Santiago, Chile

KEITH LEUNG
Applanix Corporation (Trimble Inc.), Richmond Hill, ON, Canada

Received March 2017; Revised June 2018

ABSTRACT: *This paper proposes using the number of range measurements that a detector utilizes to generate a detection as its descriptor. This one dimensional descriptor can be calculated with many range-based detectors, and its expected value is used to derive detection statistics which take into account feature occlusions to improve robotic navigation performance. To demonstrate the advantages of estimating detection statistics, they are estimated and tested within Random Finite Set and vector-based Simultaneous Localization and Mapping (SLAM) algorithms. Results from simulations and real experiments demonstrate the advantages of explicitly modeling feature detection statistics in both frameworks. © 2018 Institute of Navigation.*

INTRODUCTION

In the field of target tracking, detection statistics are considered to be of prime importance. For example, it has long been recognized that a sensor's received signal amplitudes, corresponding to true targets, should be higher than those corresponding to false alarms and that this information should be utilized [1]. However, this requires the Signal-to-Noise Ratio (SNR) corresponding to targets to be known from any sensor to target view point. Since such information is typically unavailable, detection probabilities are usually naively considered to be constant (but not necessarily zero or unity). This is despite the fact that the relative positions of objects with respect to the sensor and occlusions have a large effect on those objects' detection probabilities [1].

Within the autonomous robotic feature-based navigation literature, a vehicle's onboard sensors are used to obtain exteroceptive measurements. Measurement and feature estimate uncertainties are typically considered to lie solely in the spatial, rather than the detection, domain and are often modeled with range and bearing uncertainties [2, 3]. The joint estimation of feature locations and the trajectory of a robotic vehicle, which obtains measurements from these features, within a common coordinate frame is referred to as feature-based Simultaneous Localization and Mapping (SLAM). In most SLAM algorithms, it is considered the task of external map management, outlier rejection, and data association algorithms to minimize the problems of false alarms and missed detections, before map estimation takes place. Therefore, mathematically, 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. Similarly, the probabilities of detection of unassociated features are assumed to be zero, while the probabilities of false alarm of unassociated measurements are assumed to be unity [4].

In SLAM, the probability of detection is particularly important, since features that exit the field of view (FoV) of the sensor(s) are expected to remain in the map estimate, contrary to target tracking problems, in which the maintenance of tracks for targets that exit the FoV is usually not required. However, principled methods which currently provide such statistics calculate them based upon the measurements themselves, such as Constant False Alarm Rate (CFAR) processors [5]. In both the binary Bayes filter and RFS SLAM, feature state-based detection statistics are required both in the presence and absence of corresponding measurements, since the existence probabilities of feature states must be updated in both cases. Further, in RFS SLAM, the detection statistics required are state dependent, and there are no feature to measurement association decisions. Therefore, measurement-based detection statistics, such as those provided by CFAR processors, cannot be directly applied. This paper therefore addresses methods to estimate these statistics on a per estimated feature basis. This takes into account feature descriptor information, which is not part of the spatial location of the measurement, to aid both in map-management and data association, within vector-based SLAM approaches, and for direct use within RFS SLAM methods.

To apply this concept to spatial-based features, we propose the use of a feature descriptor, based on the estimated number of unoccluded range points sensed from that feature. This feature descriptor depends on the current best estimate of the SLAM state and the current range-based sensor scan, but not any detected features from that scan. This avoids the necessity for data association between detected and currently estimated features when applying the descriptor. In contrast to standard feature-based SLAM methods, which discard any data which have not contributed to a feature detection, a subset of this remaining data provides the extra information necessary to estimate the probability of detection of features, even when partially occluded. It will be demonstrated that this descriptor can provide an approximate sufficient statistic to define a SLAM state-dependent distribution on the probability of detection of a feature. Results will demonstrate superior SLAM performance over that obtained from the equivalent algorithms, with the usually assumed constant feature probabilities of detection within the sensor's FoV.

The work in this article was partially published in [6] and [7]. The work in [6] shows the determination of the detection statistics for the detector used in a local park environment. Meanwhile, [7] shows via simulation that the incorporation of a descriptor into RFS-based SLAM can be advantageous. This article extends [6, 7] by providing more detailed simulation analyses, demonstrating the applicability of the descriptor to standard detection concepts based on the Random Sampling Consensus (RANSAC) algorithm [8], and also by generating detection statistics for tree detection in a local park environment (Santiago, Chile) and successfully applying it to the publicly available Victoria Park dataset, from Sydney, Australia.

The paper has two main contributions. The first shows that including detection statistics into SLAM solutions can be advantageous. The second is the introduction of a descriptor, namely, the number of points used by the detector, whose prediction can be used to calculate the detection statistics for features extracted from range sensor data, particularly lidar data. The contributions of this article are based on the derivation of a range-based feature descriptor. An algorithm capable of estimating the probability of detection for any range-based detector, based on the expected value of the descriptor, is then presented. This is followed by an evaluation of the effects of including both the probability of detection and the descriptor information into set-based Rao-Blackwellized, Probability Hypothesis Density (RB-PHD)-SLAM, and the conceptually similar vector-based SLAM algorithm-Multiple Hypothesis (MH)-FastSLAM. Each framework is compared both with and without the use of the described descriptor.

The Related Work section shows some of the publications which are closely related to the present article. In the Detection Statistics in SLAM section, the use of detection statistics in SLAM is reviewed. The Estimating Feature Detection Statistics based on Range Data: The Methodology section explains how detection statistics are estimated, and the Including Descriptor Information into SLAM section then shows how descriptor information is included into the SLAM problem. The Learning from Simulated Range Data section shows the details of estimating the required statistics in simulation. The results of using these statistics are then shown in the Simulated SLAM Results section. This process is repeated with real data in the Learning from a Park Environment and Experimantal SLAM Results sections.

RELATED WORK

In some SLAM solutions, detection statistics are incorporated via the use of a binary Bayes filter to update each feature's probability of existence [2, 9]. Random Finite Set (RFS)-based filters include probability of detection and false alarm statistics into the filter's update step, making the feature detector's detection statistics an intrinsic part of the Bayesian state estimation process [10–13]. All of these techniques use estimates of the detection statistics, and this paper therefore provides principled methods for their estimation for use with RFS and vector-based SLAM frameworks. Erdinc et al. [14] showed that the PHD filter can be interpreted as an occupancy grid approach in the limit where the grid cell size approaches zero, showing a link between RFS methods and grid-based mapping [15]. A further similarity exists since both of these methods make use of 'negative information.' In grid-based mapping, a cell with no measurement reduces its occupancy probability, while in RFS-based filters, estimated features with no close measurements will have their weights/existence probabilities reduced. However, most grid-based methods to date include the detection uncertainty within the spatial uncertainty of the sensor model [15–17]. For example, most laser range finders will output a range value (e.g., maximum range or zero) even when an object is not detected. For this reason, the range measurements themselves are constant in number and therefore modeled as a vector. Contrary to traditional grid-based methods, which utilize a model of the physical sensor (e.g., sonar, lidar, or stereo cameras), the method proposed in this article uses the detection statistics of a feature detector. These detections are then modeled as a set, in which undetected objects are not present. Such models could be applied to grid-based methods which directly include detection uncertainty, such as [18].

More recent alternatives to the filtering approach to SLAM are batch-based estimators, which use non-linear optimization to obtain a Maximum Likelihood (ML) solution. Such methods include [19–23]. These methods rely on external routines to perform data association and map management, usually based on either maximum likelihood or place recognition algorithms. The RFS formulation of the problem in this form complicates the optimization process considerably, even when assuming a known number of features.

In [24], the requirement for a feature detector was removed by directly modeling laser range data in a track-before-detect approach. Each range point was therefore considered as a measurement produced by single extended features. Each extended feature state was assigned a variable probability of detection, which accounts for occlusions based only on other estimated extended features. Although this can be a good solution in tracking problems, it is common in SLAM to have unmodeled objects that can occlude the sensor. In contrast, the model proposed in this article uses the range measurements that are affected by modeled and unmodeled objects alike.

In [25], a similar strategy to the one proposed here is used to calculate the probability of detection within a pedestrian tracking framework. The size and shape of the targets is assumed known and constant, and the probability of detection is calculated in each cell of a grid using a method inspired by the grid mapping literature [2]. In contrast, the method proposed here calculates the probability of detection based on the estimated target shape and size and does not use a grid.

Clark et al. incorporated RADAR amplitude information into the PHD filter. The RADAR target Signal-to-Noise Ratio (SNR) was used both to calculate the detection statistics and to enhance the target measurement likelihood [26]. The details of its use will be explored and adapted in the Including Descriptor Information into SLAM section.

DETECTION STATISTICS IN SLAM

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. Common to both random vector and RFS SLAM approaches, the underlying stochastic system representing the robot's pose component, $\boldsymbol{x}_k$, is modeled by the, in general, non-linear discrete-time equation

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

where $\boldsymbol{g}$ is the robot motion model, $\boldsymbol{u}_{k-1}$ the odometry input, and $\boldsymbol{\delta}_{k-1}$ the process noise, both at time-step $k-1$.

The importance, and incorporation of, detection statistics into vector and RFS-based SLAM solutions is now summarized, to establish the reason for their estimation in the Estimating Feature Detection Statistics based on Range Data: The Methodology section.

Vector-Based SLAM Filtering and Detection Statistics

In the random vector formulation of feature-based SLAM, a spatial measurement model of the form

$$\boldsymbol{z}_k^i \sim \boldsymbol{h}(\boldsymbol{z}_k^i \,|\, \boldsymbol{m}_k^j, \boldsymbol{x}_k) \tag{2}$$

is necessary, where $\boldsymbol{z}_k^i$ is the ith detected feature vector at time-step k and $\boldsymbol{m}_k^j$ is a random vector containing the, possibly time varying, Cartesian position of feature j. Therefore, data association is required to relate $\boldsymbol{z}_k^i$ to $\boldsymbol{m}_k^j$ as well as a routine to manage the addition and removal of features from the map estimate. While some methods are heuristic in nature, others use a binary Bayes filter [2, 9, 15].

In the binary Bayes filter, the probability of existence $P_{\mathrm{E}}(\boldsymbol{m}_k^j \,|\, \boldsymbol{x}_{0:k}, \mathcal{Z}_{0:k})$ of the feature with state vector $\boldsymbol{m}_k^j$, given the history of robot poses $\boldsymbol{x}_{0:k}$ from discrete time 0 to k, and all feature measurement sets, $\mathcal{Z}_{0:k}$, is updated at each step. This is achieved using the probabilistic evidence provided by the current measurement set $\mathcal{Z}_k$, and assumed or known data association, so that

$$\begin{aligned} &P_{\mathrm{E}}\left(\boldsymbol{m}_k^j \,|\, \boldsymbol{x}_{0:k}, \mathcal{Z}_{0:k}\right) \\ &\quad = \frac{P_{\mathrm{E}}\left(\boldsymbol{m}_k^j \,|\, \boldsymbol{x}_k, \mathcal{Z}_k\right) P_{\mathrm{E}}\left(\boldsymbol{m}_k^j \,|\, \boldsymbol{x}_{0:k-1}, \mathcal{Z}_{0:k-1}\right)}{p(\mathcal{Z}_k \,|\, \mathcal{Z}_{0:k-1})}, \end{aligned} \tag{3}$$

which can be expressed in log-odds form as follows,

$$\begin{aligned} l_k\left(\boldsymbol{m}_k^j \,|\, \boldsymbol{x}_{0:k}, \mathcal{Z}_{0:k}\right) = l_{k-1}&\left(\boldsymbol{m}_k^j \,|\, \boldsymbol{x}_{0:k}, \mathcal{Z}_{0:k-1}\right) \\ &+ log \frac{P_{\mathrm{E}}\left(\boldsymbol{m}_k^j \,|\, \boldsymbol{x}_{0:k}, \mathcal{Z}_k\right)}{1 - P_{\mathrm{E}}\left(\boldsymbol{m}_k^j \,|\, \boldsymbol{x}_{0:k}, \mathcal{Z}_k\right)} \\ &- log \frac{P_{\mathrm{E}}\left(\boldsymbol{m}_k^j\right)}{1 - P_{\mathrm{E}}\left(\boldsymbol{m}_k^j\right)}, \end{aligned} \tag{4}$$

where the log-odds $l_k\left(\boldsymbol{m}_k^j\,|\,\boldsymbol{x}_{0:k}, \mathcal{Z}_{0:k}\right)$ is defined as

$$l_k\left(\boldsymbol{m}_k^j\,|\,\boldsymbol{x}_{0:k}, \mathcal{Z}_{0:k}\right) \equiv log\frac{P_{\mathrm{E}}\left(\boldsymbol{m}_k^j\,|\,\boldsymbol{x}_{0:k}, \mathcal{Z}_{0:k}\right)}{1-P_{\mathrm{E}}\left(\boldsymbol{m}_k^j\,|\,\boldsymbol{x}_{0:k}, \mathcal{Z}_{0:k}\right)}. \tag{5}$$

By setting the prior probability of existence $P_{\mathrm{E}}\left(\boldsymbol{m}_k^j\right)$ to a non-informative prior (0.5), the final term in (4) is zero. Then, by incorporating modeled or known probabilities of detection and false alarm, the probabilistic evidence provided by the measurements can be calculated as shown in [27]

$$P_{\mathrm{E}}\left(\boldsymbol{m}_k^j\,|\,\boldsymbol{x}_k, \mathcal{Z}_k\right) = \frac{\left(1-P_{\mathrm{D}}\left(\boldsymbol{m}_k^j|\boldsymbol{x}_k\right)\right)P_{\mathrm{FA}}(\boldsymbol{x}_k)P_{\mathrm{E}}\left(\boldsymbol{m}_k^j\right)+P_{\mathrm{D}}\left(\boldsymbol{m}_k^j\,|\,\boldsymbol{x}_k\right)P_{\mathrm{E}}\left(\boldsymbol{m}_k^j\right)}{P_{\mathrm{FA}}\left(\boldsymbol{z}_k^l|\,\boldsymbol{x}_k\right)+\left(1-P_{\mathrm{FA}}\left(\boldsymbol{z}_k^l|\boldsymbol{x}_k\right)\right)P_{\mathrm{D}}\left(\boldsymbol{m}_k^j\,|\,\boldsymbol{x}_k\right)P_{\mathrm{E}}\left(\boldsymbol{m}_k^j\right)}, \tag{6}$$

when $\boldsymbol{m}_k^j$ is associated with a measurement $\boldsymbol{z}_k^l \in \mathcal{Z}_k$, and as

$$P_{\mathrm{E}}\left(\boldsymbol{m}_k^j\,|\,\boldsymbol{x}_k, \mathcal{Z}_k\right) = \frac{\left(1-P_{\mathrm{D}}\left(\boldsymbol{m}_k^j\,|\,\boldsymbol{x}_k\right)\right)P_{\mathrm{E}}\left(\boldsymbol{m}_k^j\right)}{\left(1-P_{\mathrm{E}}\left(\boldsymbol{m}_k^j\right)\right)+\left(1-P_{\mathrm{D}}\left(\boldsymbol{m}_k^j\,|\,\boldsymbol{x}_k\right)\right)P_{\mathrm{E}}\left(\boldsymbol{m}_k^j\right)} \tag{7}$$

when $\boldsymbol{m}_k^j$ is unassociated with any measurement in $\mathcal{Z}_k$. $P_{\mathrm{D}}\left(\boldsymbol{m}_k^j\,|\,\boldsymbol{x}_k\right)$ is the probability of detection of feature $\boldsymbol{m}_k^j$ and $P_{\mathrm{FA}}\left(\boldsymbol{z}_k^l\,|\,\boldsymbol{x}_k\right)$ is the probability of $\boldsymbol{z}_k^l$ being a false alarm, both of which can be dependent on the robot's state $\boldsymbol{x}_k$ [27]. Both quantities must be substituted into (6) and (7), when an association or non-association occurs, which in turn should be substituted into (4). From (4), it can be seen that a simple measurement counting heuristic can be interpreted as a log-odds binary Bayes filter with an implicitly assumed probability of detection and false alarm. Prior research which has adopted this approach includes [2, 9].

RFS SLAM and Detection Statistics

By recognizing that the SLAM state can be naturally modeled as an RFS, Mullane et al. [10] were able to include data association and detection statistical parameters into the Bayesian estimation paradigm. Since an RFS implementation of SLAM will be used as a key demonstration of the importance of determining detection statistics in this article, an overview of Rao Blackwellized (RB)-PHD-SLAM now follows.

SLAM Definitions with RFSs

In the RFS SLAM approach, the feature-based map, up to and including time-step k, is defined as an RFS

$$\mathcal{M}_k \equiv \{\boldsymbol{m}_k^1, \boldsymbol{m}_k^2, \ldots, \boldsymbol{m}_k^{\mathfrak{m}_k}\} \tag{8}$$

where the number of features, $\mathfrak{m}_k = |\mathcal{M}_k|$, as well as the feature state values, $\boldsymbol{m}_k^i$, are random variables.

In general, the feature from which a measurement is generated is unknown. Furthermore, there is a probability of detection, $P_{\mathrm{D}}\left(\boldsymbol{m}_k^j\,|\,\boldsymbol{x}_k\right)$, associated with every feature, implying that it may be missed with probability $1 - P_{\mathrm{D}}\left(\boldsymbol{m}_k^j\,|\,\boldsymbol{x}_k\right)$. Measurements may also be generated from sensor noise or objects of non-interest (clutter), with assumed known distributions. The set of all $\mathfrak{n}_k$ measurement vectors at time k is defined as

$$\mathcal{Z}_k \equiv \left\{\boldsymbol{z}_k^1, \boldsymbol{z}_k^2, \ldots, \boldsymbol{z}_k^{\mathfrak{n}_k}\right\}, \tag{9}$$

where the number of measurements $\mathfrak{n}_k = |\mathcal{Z}_k|$, as well as their values, $\boldsymbol{z}_k^i$, are random variables.

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

$$\mathcal{Z}_k \equiv \mathcal{H}(\boldsymbol{x}_k, \mathcal{M}_k, \boldsymbol{\epsilon}_k) \cup \mathcal{E}_k\ , \tag{10}$$

where $\mathcal{H}(\boldsymbol{x}_k, \mathcal{M}_k, \boldsymbol{\epsilon}_k)$ models all expected measurements based on $\boldsymbol{x}_k$ and the map set $\mathcal{M}_k$, $\boldsymbol{\epsilon}_k$ models the spatial noise associated with the measurements at time k. $\mathcal{E}_k$ models the clutter or unexpected measurements (false alarms) at time k.

Using a Bayesian framework and a filtering approach, the Probability Density Function (PDF)

$$p\left(\boldsymbol{x}_{0:k}, \mathcal{M}_k\,|\,\mathcal{Z}_{1:k}, \boldsymbol{u}_{0:k}\right) \tag{11}$$

is sought through RFS approaches, requiring Finite Set Statistics (FISST) [28]. The estimates at each time step are made relative to a single reference frame, usually corresponding to the robot's initial pose.

In contrast to vector-based RB-Particle Filter (PF) approaches, which use the EKF to update the Gaussians for individual features [9, 29, 30], a PHD filter is used to update the map intensity function in RB-PHD-SLAM [10]. A brief overview of the main steps in the RB-PHD-SLAM filter now follows, highlighting the importance of detection statistics.

Particle Propagation

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

$$\boldsymbol{x}_{k-1}^{[i]} \sim p\left(\boldsymbol{x}_{0:k-1}\,|\,\mathcal{Z}_{1:k-1}, \boldsymbol{u}_{0:k-1}\right) \tag{12}$$

are propagated forward in time by sampling the motion noise, $\delta_{k-1}^{[i]}$ and using the robot motion model (1)

$$\boldsymbol{x}_k^{[l]} = \boldsymbol{g}\left(\boldsymbol{x}_{k-1}^{[l]}, \boldsymbol{u}_{k-1}, \delta_{k-1}^{[l]}\right) \sim p\left(\boldsymbol{x}_{0:k} \mid \mathcal{Z}_{1:k-1}, \boldsymbol{u}_{0:k-1}\right). \tag{13}$$

This step is common to vector-based Rao-Blackwellized solutions to SLAM, such as MH-FastSLAM [30].

Map Update

The predicted map intensity $v_k^-(\boldsymbol{m})$ for each particle is a function of the general feature state $\boldsymbol{m}$ and is updated as $v_k^+(\boldsymbol{m})$ with the latest measurements with the PHD filter update equation

$$\begin{aligned} v_k^+(\boldsymbol{m}) &= v_k^-(\boldsymbol{m})(1 - P_{\mathrm{D}}(\boldsymbol{m} \mid \boldsymbol{x}_k)) \\ &+ v_k^-(\boldsymbol{m}) \sum_i^{|\mathcal{Z}_k|} \frac{P_{\mathrm{D}}(\boldsymbol{m} \mid \boldsymbol{x}_k)\boldsymbol{h}(\boldsymbol{z}_k^i \mid \boldsymbol{m}, \boldsymbol{x}_k)}{\kappa\left(\boldsymbol{z}_k^i \mid \boldsymbol{x}_k\right) + \int P_{\mathrm{D}}(\boldsymbol{m} \mid \boldsymbol{x}_k)\boldsymbol{h}(\boldsymbol{z}_k^i \mid \boldsymbol{m}, \boldsymbol{x}_k) v_k^-(\boldsymbol{m}) d\boldsymbol{m}} \end{aligned} \tag{14}$$

where $\kappa\left(\boldsymbol{z}_k^i \mid \boldsymbol{x}_k\right)$ is the intensity of the clutter RFS at time k. The first term in (14) is a copy of $v_k^-(\boldsymbol{m})$ reduced by the factor $(1 - P_{\mathrm{D}}(\boldsymbol{m} \mid \boldsymbol{x}_k))$ to account for the possibility that the predicted features are undetected. In the second term, note that instead of determining data association with an external algorithm, the PHD filter performs a calculation as to how much each measurement should influence each and every feature estimate. In [10, 31], the implementation of RB-PHD-SLAM Equations 13 and 14, using Gaussian Mixtures (GM), is shown, known as RB-GM-PHD-SLAM.

Importantly, within the above steps, the map update and particle weighting steps require the knowledge of both the probability of detection, $P_{\mathrm{D}}(\boldsymbol{m} \mid \boldsymbol{x}_k)$, and the distribution (defined by its first moment or PHD) of its false alarms, $\kappa\left(\boldsymbol{z}_k^i \mid \boldsymbol{x}_k\right)$.

How do Detection Statistics Affect SLAM?

In SLAM, the map feature probability of detection $P_{\mathrm{D}}\left(\boldsymbol{m}_k^j \mid \boldsymbol{x}_k\right)$ (in (6), (7), and (14)), the probability of false alarm $P_{\mathrm{FA}}\left(\boldsymbol{z}_k^l \mid \boldsymbol{x}_k\right)$ (in (6)), and the false alarm intensity $\kappa\left(\boldsymbol{z}_k^i \mid \boldsymbol{x}_k\right)$ (in (14)) depend on the state of the robot and environment, i.e.,

$$P_{\mathrm{D}}\left(\boldsymbol{m}_k^j \mid \boldsymbol{x}_k\right) \longleftarrow P_{\mathrm{D}}\left(\boldsymbol{m}_k^j \mid \boldsymbol{x}_k, \Gamma\right) \tag{15}$$

$$P_{\mathrm{FA}}\left(\boldsymbol{z}_k^l \mid \boldsymbol{x}_k\right) \longleftarrow P_{\mathrm{FA}}\left(\boldsymbol{z}_k^l \mid \boldsymbol{x}_k, \Gamma\right) \tag{16}$$

$$\kappa\left(\boldsymbol{z}_k^i \mid \boldsymbol{x}_k\right) \longleftarrow \kappa\left(\boldsymbol{z}_k^i \mid \boldsymbol{x}_k, \Gamma\right). \tag{17}$$

Here, Γ represents the total state of the environment including, but not limited to, the subset of it being estimated (i.e., the vicinity of the feature locations) [2]. For example, if the sensor being used is affected by occlusions, then it may be the case that not all objects that can cause the occlusions are represented in the SLAM map $\mathcal{M}_k$. Therefore, theoretically, the total environmental state Γ, which includes these objects, would be necessary. As an example, Figure 1 shows a hypothetical shape (pentagon) detector, which can be occluded by walls. The value of $P_{\mathrm{D}}\left(\boldsymbol{m}_k^j \mid \boldsymbol{x}_k\right)$ associated with each pentagon is therefore highly dependent on the state of both the robot and the environment, in this case, the quantity and relative location of the walls. Even though the walls may not constitute map features of interest, they still affect the probability of detection of the pentagons. The spatial distribution of false alarms is 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 [24]. The expected number of false alarms can be estimated by utilizing a feature detector in an environment with a known number and location of features and analyzing the number of false alarms, or it can be left as a parameter of the algorithm to tune.

ESTIMATING FEATURE DETECTION STATISTICS BASED ON RANGE DATA: THE METHODOLOGY

Estimating Probability of Detection, $P_D\left(\boldsymbol{m}_k^j|\boldsymbol{x}_k\right)$

This section provides a quantified model of the probability of detection of features, based on a descriptor related to range/bearing data for use in any robotic navigation formulation. Although the descriptor can be inspired by the chosen detector algorithm, a more general solution, which is independent of the detector used to extract those features, and which takes into account occlusions,

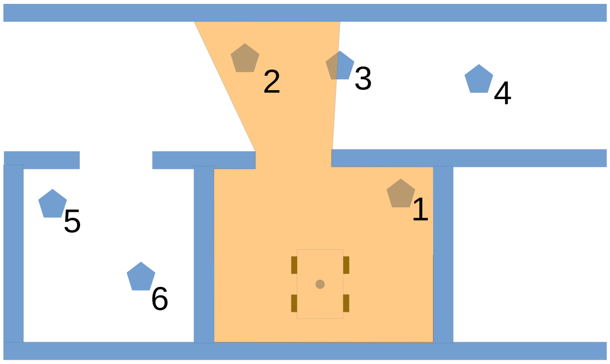


Fig. 1–A hypothetical pentagon detector, occluded by walls. The shaded yellow area shows a zone where $P_D\left(m_k^j \mid x_k\right)$ is high and in the white zones, $P_D\left(m_k^j \mid x_k\right) = 0$. From a geometric perspective, objects 1 and 2 should have high values of $P_D\left(m_k^j \mid x_k\right)$, while object 3 should have a significantly reduced $P_D\left(m_k^j \mid x_k\right)$ due to its partial occlusion by the wall. Objects 4, 5, and 6 would be expected to have $P_D\left(m_k^j \mid x_k\right) = 0$. [Color figure can be viewed at wileyonlinelibrary.com and www.ion.org]

is presented here. Therefore, the chosen descriptor model does not use any information about the feature detector itself and can be used with any detector that estimates both the position and shape attributes of a feature, such as RANSAC [8]. Further, although the effectiveness of the proposed approach is only evaluated with 2D circular features, the requirements to apply the methodology presented in this section are only that ray casting can be performed on the detected feature and that a ground truth dataset can be obtained to generate the statistical model of $P_{\mathrm{D}}\left(\boldsymbol{m}_k^j|\boldsymbol{x}_k\right)$. Therefore, this procedure can be applied to any feature to which ray casting can be applied, including but not limited to 2D line segments and 3D planar areas.

As shown in Figure 2, given the robot's location, $\boldsymbol{x}_k$, and the location and other attributes, such as the shape of features, $\boldsymbol{m}_k^j$ (i.e., a SLAM state), the number of range points that the feature is predicted to return (red crosses and green hexagons points in the figure) can be calculated via ray casting [2].

These predicted range values are then compared with the actual range values from the sensor (black points). If the actual range values at particular bearing angles are considerably lower than predicted, then the predicted range points at those bearing angles are labeled as occluded (red crosses) and the number of remaining unoccluded (green hexagons) points is defined as $\hat{n}\left(\boldsymbol{m}_k^j,\boldsymbol{x}_k\right)$. $\hat{n}\left(\boldsymbol{m}_k^j,\boldsymbol{x}_k\right)$ will be examined as an approximate sufficient statistic, which determines the probability of detection of the feature. Note that $\hat{n}\left(\boldsymbol{m}_k^j,\boldsymbol{x}_k\right)$ can be theoretically calculated for any general values of $\boldsymbol{m}_k^j$ and $\boldsymbol{x}_k$. For implementation purposes, however, if the SLAM estimate is modeled in vector form, using Normal distributions to model the robot and feature states (e.g., standard EKF SLAM), then $\boldsymbol{x}_k$ and $\boldsymbol{m}_k^j$ would be replaced with $\hat{\boldsymbol{x}}_k$ and $\hat{\boldsymbol{m}}_k^j$ representing the estimated means of the Normal distributions representing the vehicle and feature states, respectively. Given the large pose covariances typically obtained in this approach, the assumption that $\boldsymbol{x}_k$ can be approximated by its value at the mean may be problematic. However, if an RB solution to SLAM is used (as will be in this article), in which a Gaussian model of the features is assumed (e.g., FastSLAM or RB-GM-PHD-SLAM), then $\boldsymbol{x}_k$ would be replaced with the state of the ith particle $\boldsymbol{x}_k^{[i]}$ under consideration and again $\boldsymbol{m}_k^j$ would be replaced with $\hat{\boldsymbol{m}}_k^j$. Since $\hat{n}\left(\boldsymbol{m}_k^j,\boldsymbol{x}_k\right)$ is calculated using ray casting, its effectiveness will depend on the validity of using ray casting to predict the measurements of the range sensor. In its current form, this should apply to any narrow beam range sensor, being best suited to lidar sensors.

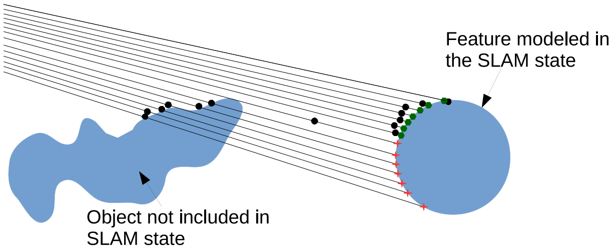

Feature modeled in the SLAM state
Object not included in SLAM state

Fig. 2–Analysis of range data from a circular shaped feature. The number of predicted (green hexagons) points, $\hat{n}\left(m_k^j,x_k\right)$, which are unoccluded, is used as a sufficient statistic of the probability of detection $P_D\left(m_k^j|x_k\right)$ of the feature. [Color figure can be viewed at wileyonlinelibrary.com and www.ion.org]

It should be noted that $\hat{n}\left(\boldsymbol{m}_k^j,\boldsymbol{x}_k\right)$ is expressed only as a function of the SLAM state ($\boldsymbol{x}_k$ and $\boldsymbol{m}_k^j$). This is despite the fact that the current range scan is necessary to determine which, if any, of the predicted points determined via ray casting are occluded by any objects not included within the SLAM map state $\mathcal{M}_k$. Although the current scan is necessary to determine this, the detected features themselves $\mathcal{Z}_k$ are not required, meaning that no data association between detections $\boldsymbol{z}_k^i \in \mathcal{Z}_k$ and $\hat{\boldsymbol{m}}_k^j \in \widehat{\mathcal{M}}_k$, where $\widehat{\mathcal{M}}_k$ is the estimated map at time k, is necessary. In fact, it is a subset of the range values from each scan, which are not necessarily used by the SLAM feature state detector, which are used to determine $\hat{n}(\boldsymbol{m}_k^j,\boldsymbol{x}_k)$. In contrast to standard feature-based SLAM methods, which discard such data within each scan, in this article, these data provide the extra information necessary to estimate the probability of detection of features, even when partially occluded. Therefore, this article analyzes the feasibility of assuming $\hat{n}\left(\boldsymbol{m}_k^j,\boldsymbol{x}_k\right)$ to be a sufficient statistic of $P_{\mathrm{D}}\left(\boldsymbol{m}_k^j|\boldsymbol{x}_k\right)$, i.e., it poses the question of whether

$$P_{\mathrm{D}}\left(\boldsymbol{m}_k^j|\boldsymbol{x}_k,\Gamma,\hat{n}\left(\boldsymbol{m}_k^j,\boldsymbol{x}_k\right)\right) \approx P_{\mathrm{D}}\left(\boldsymbol{m}_k^j|\hat{n}\left(\boldsymbol{m}_k^j,\boldsymbol{x}_k\right)\right)? \tag{18}$$

It should also be noted that the analysis presented here could be applied in tracking approaches such as [24], allowing the inclusion of unmodeled objects within the calculation of the detection probabilities. In particular, $\hat{n}\left(\boldsymbol{m}_k^j,\boldsymbol{x}_k\right)$ could be used to estimate the mean of the Poisson RFS used in [24], representing the number of measurements per extended feature.

As carried out in [26], the dependency of $P_{\mathrm{D}}\left(\boldsymbol{m}_k^j|\hat{n}\left(\boldsymbol{m}_k^j,\boldsymbol{x}_k\right)\right)$ on $\hat{n}\left(\boldsymbol{m}_k^j,\boldsymbol{x}_k\right)$ could be calculated by integrating the variables used by the detector over the detectable area:

$$P_{\mathrm{D}}\left(\boldsymbol{m}_k^j \mid \hat{n}\left(\boldsymbol{m}_k^j, \boldsymbol{x}_k\right)\right) = \int_{\boldsymbol{\theta} \in \boldsymbol{\theta}_{vol}} p_D\left(\boldsymbol{\theta} \mid \hat{n}\left(\boldsymbol{m}_k^j, \boldsymbol{x}_k\right)\right) d\boldsymbol{\theta}, \tag{19}$$

where $\boldsymbol{\theta}$ is the descriptor used by the detector to make its decision, $\boldsymbol{\theta}_{vol}$ is the multidimensional volume in the descriptor space where the detector will decide to return a feature detection, and $p_D()$ is a multivariate distribution corresponding to detections. Unlike typical radar detectors, which use only the returned power for detection decisions, a general feature detector can use several quantities to make its decision. Equation 19 generalizes the probability of detection of a range-based feature extractor, given occlusion information $\hat{n}(\boldsymbol{m}_k^j, \boldsymbol{x}_k)$).

One method to estimate the probability of detection is to estimate $p_D\left(\boldsymbol{\theta} \mid \hat{n}\left(\boldsymbol{m}_k^j, \boldsymbol{x}_k\right)\right)$ and then use (19) to determine $P_{\mathrm{D}}\left(\boldsymbol{m}_k^j \mid \hat{n}\left(\boldsymbol{m}_k^j, \boldsymbol{x}_k\right)\right)$. The potentially high dimensionality of both $\boldsymbol{\theta}$ and $\boldsymbol{\theta}_{vol}$, however, means that a large dataset containing a significant number of descriptor $\boldsymbol{\theta}$, $\hat{n}\left(\boldsymbol{m}_k^j, \boldsymbol{x}_k\right)$ pairs would be required. A simpler method is to learn $P_{\mathrm{D}}\left(\boldsymbol{m}_k^j \mid \hat{n}\left(\boldsymbol{m}_k^j, \boldsymbol{x}_k\right)\right)$ directly from measurements within a test dataset as follows.

Experimental Determination of Detection Probabilities

This section demonstrates that feature probabilities of detection can be adequately and experimentally quantified based on $\hat{n}\left(\boldsymbol{m}_k^j, \boldsymbol{x}_k\right)$, via statistical analyses on range datasets. Initially, a dataset is required from an environment where the ground truth positions of features are known, via independent means. A way to achieve this is through the use of features identifiable by humans, i.e., semantic features. Equation 18 approximates the probability of detection of such a feature, which should be based on all the necessary information, Γ, by the probability of detection given the value of the single parameter $\hat{n}\left(\boldsymbol{m}_k^j, \boldsymbol{x}_k\right)$.

In the test dataset, measurements manually associated with known ground truth features were used to determine the probability of detection of those features, conditioned on the number of unoccluded points. Intuitively, the more 'representative' the test data is of the actual environment in which SLAM is to be achieved, the better the detection probability estimates would be expected to be. However, if the number of unoccluded points is a good approximation of a sufficient statistic of the detection probability, then it should have a positive impact on range data-based SLAM performance in general environments.

Therefore, within the test dataset, the predicted number of unoccluded points in the scan recorded at time k, $\hat{n}\left(\boldsymbol{m}_k^j, \boldsymbol{x}_k\right)$, for each feature, $\boldsymbol{m}_k^j \in \mathcal{M}_k$, – robot pose, $\boldsymbol{x}_k$, pair, was calculated. For every possible number i of calculated unoccluded points, the actual number of times $\hat{n}_{\mathrm{act}}$ that a particular value of $\hat{n}\left(\boldsymbol{m}_k^j, \boldsymbol{x}_k\right) = i$ was calculated is

$$\hat{n}_{\mathrm{act}} = \sum_{k=0}^{N_s} \sum_{\boldsymbol{m}_k^j \in \mathcal{M}_k} \delta\left(\hat{n}\left(\boldsymbol{m}_k^j, \boldsymbol{x}_k\right), i\right) \quad \forall i \geq 0, \tag{20}$$

where N_s is the total number of scans in the dataset and $\delta(\cdot, \cdot)$ is the Kronecker delta function.

The number of times that a feature, with a predicted number of unoccluded points, $\hat{n}\left(\boldsymbol{m}_k^j, \boldsymbol{x}_k\right)$, equal to i, produced a valid detection, $\hat{n}_{\mathrm{det}}$, is given by:

$$\hat{n}_{\mathrm{det}} = \sum_{k=0}^{N_s} \sum_{\boldsymbol{m}_k^j \in \mathcal{M}_k} \delta\left(\hat{n}\left(\boldsymbol{m}_k^j, \boldsymbol{x}_k\right), i\right) c\left(\boldsymbol{m}_k^j, k\right) \quad \forall i \geq 0 \tag{21}$$

where $c\left(\boldsymbol{m}_k^j, k\right)$ is an indicator function, which equals unity if feature $\boldsymbol{m}_k^j \in \mathcal{M}_k$ was detected in the scan recorded at time k, and zero otherwise. Then, the probability of detection for each feature with a predicted number of points, $\hat{n}\left(\boldsymbol{m}_k^j, \boldsymbol{x}_k\right)$, equal to i, can be approximated as the ratio of (21) and (20):

$$P_{\mathrm{D}}\left(\boldsymbol{m}_k^j \mid \hat{n}(\boldsymbol{m}_k^j, \boldsymbol{x}_k) = i\right) \approx \hat{n}_{\mathrm{det}}/\hat{n}_{\mathrm{act}}. \tag{22}$$

The estimated probabilities of detection, utilizing the above concept, will be shown graphically in the Estimating Detection Probability $P_{\mathrm{D}}\left(\boldsymbol{m}_k^j \mid \hat{n}\left(\boldsymbol{m}_k^j, \boldsymbol{x}_k\right)\right)$ section.

Estimating Probabilities of False Alarm

A full analysis of the probability of false alarm would require a model for every possible range scan, which does not contain any of the semantic features of interest. In practice, this is infeasible. Importantly, the statistical representation of the number of false alarms in RB-PHD-SLAM (and indeed in many target tracking formulations) is a Poisson random set, which only requires an estimate of their expected number. When using the binary Bayes filter, the probability of false alarm can be approximated by dividing the total number of false alarms by the total number of detected features in the dataset,

$$P_{\mathrm{FA}}\left(\boldsymbol{z}_k^l \mid \boldsymbol{x}_k\right) \approx \left(\sum_{k=0}^{N_s} N_{\mathrm{FA}}(k)\right) / \left(\sum_{k=0}^{N_s} N_m(k)\right) \tag{23}$$

where $N_m(k)$ is the total number of features detected in the scan recorded at time k and $N_{\mathrm{FA}}(k)$ is the total number of detected features that could not be associated with any feature in the scan recorded at time k. In a manner similar to the probabilities of detection outlined above, the statistical analysis of range-based data, known not to 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 detections which are manually associated with actual features from a sufficiently large dataset, as carried out here. The clutter distribution was modeled as Poisson in number, with mean value equal to the average number of false alarms per scan. Its spatial distribution was modeled as uniform, resulting in the following clutter intensity function for use in Equation (14):

$$\kappa(\boldsymbol{z}_k^i \,|\, \boldsymbol{x}_k) \approx \left(\sum_{k=0}^{N_s} N_{\mathrm{FA}}(k) \right) \Big/ \left(N_s \int_{\boldsymbol{z}_{\mathrm{FoV}}} d\boldsymbol{z} \right). \tag{24}$$

INCLUDING DESCRIPTOR INFORMATION INTO SLAM

Including Descriptor Information into PHD-SLAM

In the field of target tracking, Clark et al. [26] proposed a modification to the PHD filter that uses RADAR measurement amplitude information together with each accompanying range value. The idea behind this paper is to use this theory and replace the target amplitude with a general feature descriptor, by changing the likelihoods accordingly. Then, the extended measurement vectors $\mathring{\boldsymbol{z}}_k^i$ are defined as

$$\mathring{\boldsymbol{z}}_k^i \equiv \left[\boldsymbol{z}_k^i \; a_i \right]^{\mathsf{T}}, \tag{25}$$

where $\boldsymbol{z}_k^i$ corresponds to the spatial part of the ith measurement (i.e., what used to be the entire ith measurement), and a_i is the amplitude information (which in general could be replaced by a descriptor). Hence, $\mathring{\boldsymbol{h}}(\mathring{\boldsymbol{z}}_k^i \,|\, \boldsymbol{m}_k^j, \boldsymbol{x}_k)$ and $\mathring{\kappa}(\mathring{\boldsymbol{z}}_k^i \,|\, \boldsymbol{x}_k)$ account for the joint likelihood of target state and amplitude. In [26], the distributions of this amplitude, under false alarm and detection hypotheses, were modeled as Swerling Type I and II models [32], which provide probabilistic (Rayleigh) models of received power fluctuations when the RADAR-to-target viewing aspect changes. The dependency on the environment is modeled by a single parameter d, where the expected (mean) SNR from a target was $1 + d$, where $p_{\mathrm{FA}}(a)$ and $p_{\mathrm{D}}(a \,|\, d)$ are the distributions of a for false alarms and targets of interest, respectively. The modified PHD update equation used in [26] is then

$$v_k^+(\boldsymbol{m}) = v_k^-(\boldsymbol{m})(1 - P_{\mathrm{D}}(d)) + v_k^-(\boldsymbol{m}) \times \sum_{i}^{|\mathcal{Z}_k|} \frac{P_{\mathrm{D}}(d)\mathring{\boldsymbol{h}}\left(\mathring{\boldsymbol{z}}_k^i \,|\, \boldsymbol{m}, \boldsymbol{x}_k\right)}{\mathring{\kappa}\left(\mathring{\boldsymbol{z}}_k^i | \boldsymbol{x}_k\right) + \int P_{\mathrm{D}}(d)\mathring{\boldsymbol{h}}\left(\mathring{\boldsymbol{z}}_k^i | \boldsymbol{m}, \boldsymbol{x}_k\right) v_k^-(\boldsymbol{m}) d\boldsymbol{m}} \tag{26}$$

where

$$\mathring{\boldsymbol{h}}\left(\mathring{\boldsymbol{z}}_k^i \,|\, \boldsymbol{m}, \boldsymbol{x}_k\right) = g_a^\tau(a_i \,|\, d)\boldsymbol{h}\left(\boldsymbol{z}_k^i \,|\, \boldsymbol{m}, \boldsymbol{x}_k\right) \tag{27}$$

represents the product of the amplitude measurement likelihood and the spatial measurement likelihood, and

$$\mathring{\kappa}\left(\mathring{\boldsymbol{z}}_k^i \,|\, \boldsymbol{x}_k\right) = \kappa\left(\boldsymbol{z}_k^i \,|\, \boldsymbol{x}_k\right) g_{\mathrm{FA}}^\tau(a_i) \tag{28}$$

represents the extended clutter intensity, including the amplitude likelihood. Respectively, $g_a^\tau(a_i \,|\, d)$ and $g_{\mathrm{FA}}^\tau(a_i)$ are the measurement and false alarm likelihoods of the amplitude a_i, related to measurement z_i, based on a detection threshold τ. $P_{\mathrm{D}}(d)$ is the probability of detection given an object SNR. Therefore,

$$P_{\mathrm{D}}(d) = \int_{a>\tau} p_{\mathrm{D}}(a \,|\, d) da \tag{29}$$

$$g_{\mathrm{FA}}^\tau(a_i) = \begin{cases} \frac{p_{\mathrm{FA}}(a_i)}{\int_{a_i>\tau} p_{\mathrm{FA}}(a_i) da_i}, & a_i \geq \tau \\ 0, & a_i < \tau \end{cases} \tag{30}$$

$$g_a^\tau(a_i \,|\, d) = \begin{cases} \frac{p_{\mathrm{D}}(a_i)}{\int_{a_i>\tau} p_{\mathrm{D}}(a_i \,|\, d) da_i}, & a_i \geq \tau \\ 0, & a_i < \tau \,. \end{cases} \tag{31}$$

The difference between (26) and the standard PHD update (14) is the inclusion of the measurement and false alarm likelihoods. Measurements which are more likely to be true detections rather than false alarms will receive higher weights.

In this paper, instead of using the Swerling-based Rayleigh distributions of the received signal amplitude adopted for radar measurement likelihoods in [26], a generic model is applied. The extended measurement vectors $\mathring{\boldsymbol{z}}_k^i$ here are redefined as

$$\mathring{\boldsymbol{z}}_k^i \equiv \left[\boldsymbol{z}_k^i \; \boldsymbol{\theta}_k^i \right]^{\mathsf{T}}, \tag{32}$$

where $\boldsymbol{\theta}_k^i$ is a descriptor vector accompanying measurement $\boldsymbol{z}_k^i$. Along with this, a variable $\boldsymbol{d}_{\boldsymbol{\theta}}$ equivalent to the target SNR is used. A general probability distribution $p_D(\boldsymbol{\theta} \,|\, \boldsymbol{d}_{\boldsymbol{\theta}})$ for true detections can be used to define feature likelihood $g_{\boldsymbol{\theta}}^{\boldsymbol{\theta}_{\mathrm{vol}}}(\boldsymbol{\theta} \,|\, \boldsymbol{d}_{\boldsymbol{\theta}})$ equivalent to $g_a^\tau(a_i | d)$ in (31), for the general feature descriptor $\boldsymbol{\theta}$, i.e.,

$$g_{\boldsymbol{\theta}}^{\boldsymbol{\theta}_{\mathrm{vol}}}(\boldsymbol{\theta} \,|\, \boldsymbol{d}_{\boldsymbol{\theta}}) = \begin{cases} \frac{p_D(\boldsymbol{\theta} \,|\, \boldsymbol{d}_{\boldsymbol{\theta}})}{\int_{\boldsymbol{\theta}_{\mathrm{vol}}} p_D(\boldsymbol{\theta} \,|\, \boldsymbol{d}_{\boldsymbol{\theta}}) d\boldsymbol{\theta}} & \text{if } \boldsymbol{\theta} \in \boldsymbol{\theta}_{\mathrm{vol}} \\ 0 & \text{if } \boldsymbol{\theta} \notin \boldsymbol{\theta}_{\mathrm{vol}} \,. \end{cases} \tag{33}$$

An example of such a descriptor vector will be given in the Estimating Feature Likelihoods $g_{\boldsymbol{\theta}}^{\boldsymbol{\theta}\,\mathrm{vol}}(\boldsymbol{\theta})$ and $g_{FA}^{\boldsymbol{\theta}\,\mathrm{vol}}(\boldsymbol{\theta})$ section. In (33), $p_D(\boldsymbol{\theta}\,|\,\boldsymbol{d_\theta})$ is a distribution on $\boldsymbol{\theta}$, with known parameter $\boldsymbol{d_\theta}$ being equivalent to the SNR in the RADAR implementation. A descriptor distribution for false alarms $p_{\mathrm{FA}}(\boldsymbol{\theta})$ has to be normalized within the detectable volume, to create likelihood $g_{\mathrm{FA}}^{\boldsymbol{\theta}\,\mathrm{vol}}(\boldsymbol{\theta})$ equivalent to $g_{\mathrm{FA}}^{\tau}(a_i)$

$$g_{\mathrm{FA}}^{\boldsymbol{\theta}\,\mathrm{vol}}(\boldsymbol{\theta}) = \begin{cases} \dfrac{p_{\mathrm{FA}}(\boldsymbol{\theta})}{\int\limits_{\boldsymbol{\theta}_{\mathrm{vol}}} p_{\mathrm{FA}}(\boldsymbol{\theta})d\boldsymbol{\theta}} & \text{if } \boldsymbol{\theta} \in \boldsymbol{\theta}_{\mathrm{vol}} \\ 0 & \text{if } \boldsymbol{\theta} \notin \boldsymbol{\theta}_{\mathrm{vol}}\,. \end{cases} \tag{34}$$

These likelihoods can be used with the detection statistics $P_{\mathrm{D}}(\boldsymbol{m}\,|\,\hat{n}(\boldsymbol{m},\boldsymbol{x}_k))$ and $\kappa\left(\boldsymbol{z}_k^i\,|\,\boldsymbol{x}_k\right)$ in a modified PHD filter update equation equivalent to (26), now given by

$$\begin{aligned} v_k^+(\boldsymbol{m}) &= v_k^-(\boldsymbol{m})(1 - P_{\mathrm{D}}(\boldsymbol{m}\,|\,\hat{n}(\boldsymbol{m},\boldsymbol{x}_k))) + v_k^-(\boldsymbol{m}) \\ &\times \sum_{i}^{|\mathcal{Z}_k|} \frac{P_{\mathrm{D}}(\boldsymbol{m}\,|\,\hat{n}(\boldsymbol{m},\boldsymbol{x}_k))\mathring{\boldsymbol{h}}(\mathring{\boldsymbol{z}}_k^i\,|\,\boldsymbol{m},\boldsymbol{x}_k)}{\mathring{\kappa}(\mathring{\boldsymbol{z}}_k^i|\boldsymbol{x}_k) + \int P_{\mathrm{D}}(\boldsymbol{m}|\,\hat{n}(\boldsymbol{m},\boldsymbol{x}_k))\mathring{\boldsymbol{h}}\left(\mathring{\boldsymbol{z}}_k^i|\,\boldsymbol{m},\boldsymbol{x}_k\right)v_k^-(\boldsymbol{m})d\boldsymbol{m}}. \end{aligned} \tag{35}$$

In principle, $\boldsymbol{\theta}$ can be any descriptor based on the measurement. It is desirable that its distribution for detections $p_D(\boldsymbol{\theta}\,|\,\boldsymbol{d_\theta})$ and for false alarms $p_{\mathrm{FA}}(\boldsymbol{\theta})$ are separated as far as possible in the $\boldsymbol{\theta}$ space, so that the measurements that are more likely to be false alarms will have lower weights in the modified PHD update Equation (35). If the likelihoods $g_{\mathrm{FA}}^{\boldsymbol{\theta}\,\mathrm{vol}}(\boldsymbol{\theta})$ and $g_{\boldsymbol{\theta}}^{\boldsymbol{\theta}\,\mathrm{vol}}(\boldsymbol{\theta}\,|\,\boldsymbol{d_\theta})$ are approximately equal, the modified PHD filter update in (35) will tend to its traditional form given in (14). In this paper, the descriptor $\boldsymbol{\theta}$ will be the number of points n_{d} used to detect the feature. The variable $\boldsymbol{d_\theta}$, which is equivalent to the target SNR in Clark's work, is the predicted value of n_{d}, i.e., $\boldsymbol{d_\theta} = \hat{n}(\boldsymbol{m},\boldsymbol{x}_k)$.

Including Descriptor Information into MH-FastSLAM

Since $P_{\mathrm{D}}\left(\boldsymbol{m}_k^j,\boldsymbol{x}_k\right)$ is used within the binary Bayes filter addition to MH-FastSLAM, the descriptor information can be included by replacing (2) with the extended measurement likelihood

$$\mathring{\boldsymbol{h}}\left(\mathring{\boldsymbol{z}}_k^i\,|\,\boldsymbol{m}_k^j,\boldsymbol{x}_k,\boldsymbol{d_\theta}\right) = g_{\boldsymbol{\theta}}^{\boldsymbol{\theta}\,\mathrm{vol}}(\boldsymbol{\theta}\,|\,\boldsymbol{d_\theta})\boldsymbol{h}\left(\boldsymbol{z}_k^i\,|\,\boldsymbol{m}_k^j,\boldsymbol{x}_k\right) \tag{36}$$

in the data association and particle weighting steps. In MH-FastSLAM, there is no mechanism known to the authors to incorporate $g_{\mathrm{FA}}^{\boldsymbol{\theta}\,\mathrm{vol}}(\boldsymbol{\theta})$ and therefore it is not used.

Note that the modified PHD-SLAM update (35) requires estimates of both $P_{\mathrm{D}}(\boldsymbol{m}\,|\,\hat{n}(\boldsymbol{m},\boldsymbol{x}_k))$ and $g_{\boldsymbol{\theta}}^{\boldsymbol{\theta}\,\mathrm{vol}}(\boldsymbol{\theta}\,|\,\boldsymbol{d_\theta})$ as well as the clutter term $\kappa\left(\boldsymbol{z}_k^i\,|\,\boldsymbol{x}_k\right)g_{\mathrm{FA}}^{\boldsymbol{\theta}\,\mathrm{vol}}(\boldsymbol{\theta})$. Similarly, (36), which corresponds to a SLAM measurement likelihood which includes the descriptor information, requires an estimate of $g_{\boldsymbol{\theta}}^{\boldsymbol{\theta}\,\mathrm{vol}}(\boldsymbol{\theta}\,|\,\boldsymbol{d_\theta})$. Both of these SLAM formulations will therefore use this modified likelihood $\mathring{\boldsymbol{h}}\left(\mathring{\boldsymbol{z}}_k^i\,|\,\boldsymbol{m}_k^j,\boldsymbol{x}_k,\boldsymbol{d_\theta}\right)$ in their data association procedures, which is an external maximum likelihood algorithm in MH-FastSLAM and is naturally part of the Bayesian SLAM update in RB-PHD-SLAM. A similar concept is applied in the following sections, in which instead of received amplitude information, a range-bearing-based feature descriptor $\boldsymbol{\theta}$ is defined, which uses ray casting.

LEARNING FROM SIMULATED RANGE DATA

This section demonstrates the validity of the concept to determine the detection statistics of a feature detector, based on the number of unoccluded points $\hat{n}\left(\boldsymbol{m}_k^j,\boldsymbol{x}_k\right)$ described in the Estimating Feature Detection Statistics based on Range Data: The Methodology section. It is applied in a simulated environment with a feature extraction method based on RANSAC. The environment was simulated with the Gazebo simulator [33], composed of cylinders,

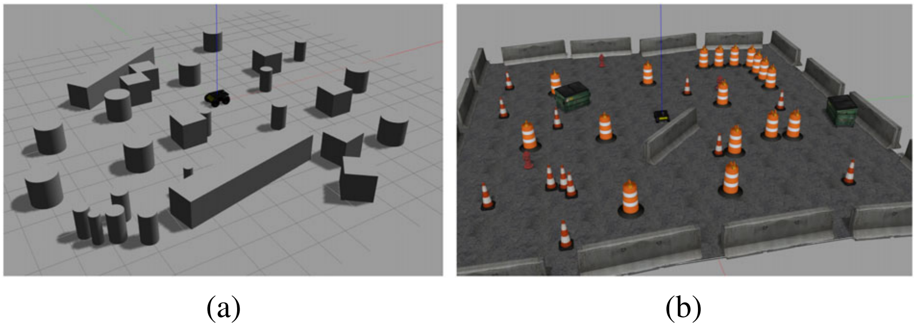


Fig. 3–Simulated environments used to obtain the detection statistics. The same environment used in simulations (a) and a second dataset (b). [Color figure can be viewed at wileyonlinelibrary.com and www.ion.org]

which are the features of interest, and cuboids, which are objects of non-interest. The SLAM state will be composed of the trajectory of the robot and the center coordinates of the cylinders in the vehicle's plane of motion. The simulated environment is shown in Figure 3(a) and its plan view is also shown in the background of the SLAM results later in Figure 9.

A simple two-step detector is used in the simulation, composed of a segmentation method (based on a distance threshold of 0.3 m), followed by RANSAC to detect circular objects in the segmented data. The RANSAC algorithm was implemented based on a required circular in lier number of 13 range points and an in lier distance threshold of 0.02 m and was executed 12,500 times per feature, corresponding to a RANSAC detection failure rate of 1.7%.

Estimating the Detection Probability $P_D\left(\boldsymbol{m}_k^j|\hat{n}\left(\boldsymbol{m}_k^j,\boldsymbol{x}_k\right)\right)$

By using the known poses of the robot and features provided by the simulator, (22) can be applied to obtain an estimate of the probability of detection given the predicted number of points, $P_\mathrm{D}\left(\boldsymbol{m}_k^j\,|\,\hat{n}\left(\boldsymbol{m}_k^j,\boldsymbol{x}_k\right)\right)$. Figure 4 shows the results.

A 99% confidence interval was calculated based on the number of points available at each value of $\hat{n}\left(\boldsymbol{m}_k^j,\boldsymbol{x}_k\right)$, showing the uncertainty in the estimate. As can be seen from the figure, the probability of detection given a predicted number of points $\hat{n}\left(\boldsymbol{m}_k^j,\boldsymbol{x}_k\right)$ is very close to zero when $\hat{n}\left(\boldsymbol{m}_k^j,\boldsymbol{x}_k\right))$ is lower than 7, then it tends to increase as $\hat{n}\left(\boldsymbol{m}_k^j,\boldsymbol{x}_k\right)$ increases to 14. For values of $\hat{n}\left(\boldsymbol{m}_k^j,\boldsymbol{x}_k\right))$ higher than 14, the probability of detection fluctuates about a value of approximately 0.8. This could be interpreted as the intrinsic probability of detection of the feature detector being 0.8, once it is known not to be occluded.

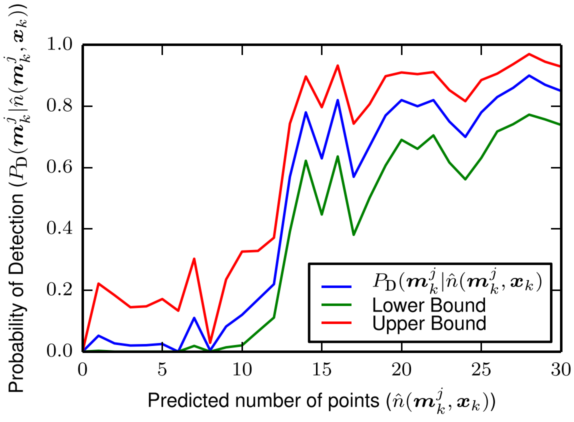


Fig. 4–The probability of detection as a function of the number of unoccluded points $\hat{n}\left(\boldsymbol{m}_k^j,\boldsymbol{x}_k\right)$. Based on the amount of data available, a 99% confidence interval is calculated, the bounds of which are shown. [Color figure can be viewed at wileyonlinelibrary.com and www.ion.org]

Estimating $P_{FA}\left(\boldsymbol{z}_k^l|\boldsymbol{x}_k\right)$ and Intensity Function $\kappa\left(\boldsymbol{z}_k^i|\boldsymbol{x}_k\right)$

To estimate the probability of false alarm and, for the case of the RB-PHD-SLAM filter, the false alarm intensity function, the same dataset can be used, labeling the unassociated measurements as false alarms and estimating the statistical parameters of those measurements. In particular, since the clutter distribution is assumed Poisson in number in the PHD filter, only the average number of false alarms per scan needs to be estimated. To assess the accuracy of the Poisson approximation, Figure 5 shows a distribution of the number of false alarms per scan, and a Poisson distribution (red) with parameter λ = 0.06 calculated from the sampled data. $\kappa\left(\boldsymbol{z}_k^i\,|\boldsymbol{x}_k\right)$ was then determined from (24):

$$\kappa\left(\boldsymbol{z}_k^i\,|\boldsymbol{x}_k\right) \approx \frac{\sum\limits_{k=0}^{N_s} N_\mathrm{FA}(k)}{N_s \int\limits_{\boldsymbol{z}_\mathrm{FoV}} d\boldsymbol{z}} = \frac{\lambda}{\int\limits_{\boldsymbol{z}_\mathrm{FoV}} d\boldsymbol{z}}\,, \tag{37}$$

while $P_\mathrm{FA}\left(\boldsymbol{z}_k^l\,|\boldsymbol{x}_k\right)$ was determined from (23) to be 2.34%.

Estimating Feature Likelihoods $g_\theta^{\theta_{vol}}(\theta)$ and $g_\mathrm{FA}^{\theta_{vol}}(\theta)$

In general, after choosing a suitable descriptor, data need to be obtained in an environment with features identified by independent means in order

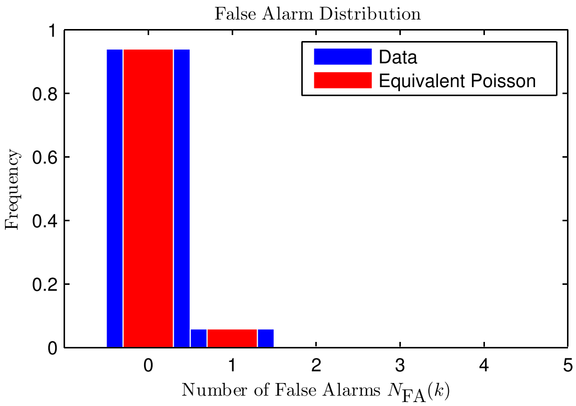


Fig. 5–Distribution of the number of false alarms per scan, from the simulation. The red graph shows the equivalent Poisson distribution, which appears almost identical in this case. Note that the red bars are superimposed on to the blue bars, with half their width, to improve visibility. [Color figure can be viewed at wileyonlinelibrary.com and www.ion.org]

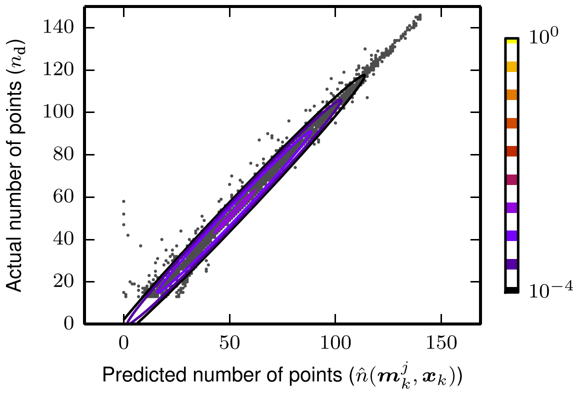


Fig. 6–Number of points n_d measured as a function of the predicted number of points $\hat{n}\left(\boldsymbol{m}_k^j, \boldsymbol{x}_k\right)$. The dependency of n_d on $\hat{n}\left(\boldsymbol{m}_k^j, \boldsymbol{x}_k\right)$ can be observed in this figure. The contours of an equivalent bivariate Normal distribution are also shown. [Color figure can be viewed at wileyonlinelibrary.com and www.ion.org]

to model the probability distributions $p_D(\boldsymbol{\theta})$ and $p_{\mathrm{FA}}(\boldsymbol{\theta})$.

As in [26], where the conditional distribution $p_D(a\,|\,d)$ was determined from a Rayleigh distribution, a descriptor which depends on the assumed sufficient statistic $\hat{n}\left(\boldsymbol{m}_k^j, \boldsymbol{x}_k\right)$ is necessary here. A descriptor $\boldsymbol{\theta} = n_\mathrm{d}$ is therefore defined, corresponding to the number of range points used in the detection of a feature. As stated in section "Including Descriptor Information into SLAM", n_d would be equivalent to a and $\hat{n}\left(\boldsymbol{m}_k^j, \boldsymbol{x}_k\right)$ to d in [26]. Therefore, $g_{\boldsymbol{\theta}}^{\boldsymbol{\theta}\,\mathrm{vol}}(\boldsymbol{\theta}) = p\left(n_\mathrm{d}\,|\,\hat{n}\left(\boldsymbol{m}_k^j, \boldsymbol{x}_k\right)\right)$ needs to be determined. To model the distribution $p\left(n_\mathrm{d}\,|\,\hat{n}\left(\boldsymbol{m}_k^j, \boldsymbol{x}_k\right)\right)$, first, the joint probability distribution $p\left(n_\mathrm{d}, \hat{n}\left(\boldsymbol{m}_k^j, \boldsymbol{x}_k\right)\right)$ is modeled (Figure 6). For this purpose, n_d is plotted as a function of $\hat{n}\left(\boldsymbol{m}_k^j, \boldsymbol{x}_k\right)$, from a total of 10,091 simulated laser scans which produced 27,193 measurements, in Figure 6. As can be seen in the figure, there is a strong dependency of n_d on $\hat{n}\left(\boldsymbol{m}_k^j, \boldsymbol{x}_k\right)$.

Estimating $p\left(n_\mathrm{d}\,|\,\hat{n}\left(\boldsymbol{m}_k^j, \boldsymbol{x}_k\right)\right)$ directly would be difficult given the sparsity of the data at some values of $\hat{n}\left(\boldsymbol{m}_k^j, \boldsymbol{x}_k\right)$ (see Figure 6 at $\hat{n}\left(\boldsymbol{m}_k^j, \boldsymbol{x}_k\right)$ values of 120 and higher). Contours of an equivalent Normal distribution are also shown in Figure 6. Using the estimated mean μ and covariance Σ, the conditional distribution can be obtained as

$$p\left(n_\mathrm{d}\,|\,\hat{n}\left(\boldsymbol{m}_k^j, \boldsymbol{x}_k\right)\right) = \mathcal{N}\left(n_\mathrm{d}\,|\,\mu\left(n_\mathrm{d}\,|\,\hat{n}\left(\boldsymbol{m}_k^j, \boldsymbol{x}_k\right)\right)\right. \left.\Sigma\left(n_\mathrm{d}\,|\,\hat{n}\left(\boldsymbol{m}_k^j, \boldsymbol{x}_k\right)\right)\right), \tag{38}$$

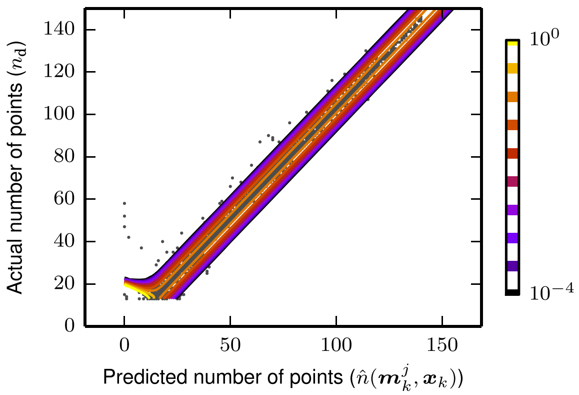


Fig. 7–The conditional distribution of n_d given $\hat{n}(\boldsymbol{m}_k^j, \boldsymbol{x}_k)$, calculated from the Normal distribution of Figure 6. [Color figure can be viewed at wileyonlinelibrary.com and www.ion.org]

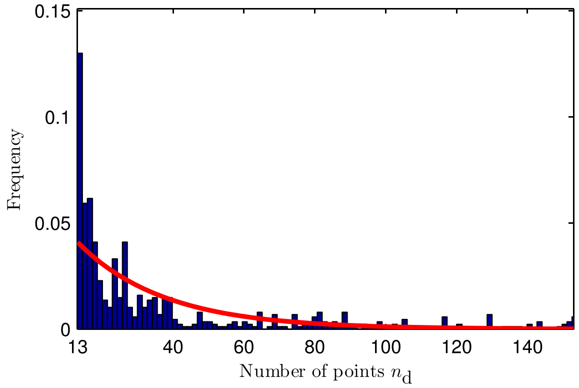


Fig. 8–Descriptor distribution for false alarms, which resembles an exponential distribution. [Color figure can be viewed at wileyonlinelibrary.com and www.ion.org]

where

$$\mu\left(n_\mathrm{d}\,|\,\hat{n}\left(\boldsymbol{m}_k^j, \boldsymbol{x}_k\right)\right) = \mu_1 + \frac{\Sigma_{12}}{\Sigma_{22}}\left(\hat{n}\left(\boldsymbol{m}_k^j, \boldsymbol{x}_k\right) - \mu_2\right)\,, \tag{39}$$

$$\Sigma\left(n_\mathrm{d}\,|\,\hat{n}\left(\boldsymbol{m}_k^j, \boldsymbol{x}_k\right)\right) = \Sigma_{11} - \frac{\Sigma_{12}\Sigma_{21}}{\Sigma_{22}}\,, \tag{40}$$

where Σ_{lm} is the element of the matrix Σ in row l and column m, and μ_j is the jth element of vector μ.

Finally, by using the fact that the RANSAC circle detector cannot generate detections with less points than a threshold $N_{\min}$ (= 13 in this case), the Normal distribution should be rescaled. This final conditional distribution is

$$g_{\boldsymbol{\theta}}^{\boldsymbol{\theta}\,\mathrm{vol}}(\boldsymbol{\theta}) = \begin{cases} \dfrac{p\left(n_\mathrm{d}\,|\,\hat{n}\left(\boldsymbol{m}_k^j, \boldsymbol{x}_k\right)\right)}{\int_{N_{\min}}^{\infty} p\left(n_\mathrm{d}\,|\,\hat{n}\left(\boldsymbol{m}_k^j, \boldsymbol{x}_k\right)\right) dn_\mathrm{d}} & \text{if } n_\mathrm{d} \geq N_{\min} \\ 0 & \text{if } n_\mathrm{d} < N_{\min} \end{cases}, \tag{41}$$

and is shown in Figure 7.

For the case of false alarms, the distribution of the number of points $g_{FA}^{\boldsymbol{\theta}\,\mathrm{vol}}(\boldsymbol{\theta})$ has to be modeled

Table 1 — Parameters used in the SLAM simulations

Parameter	Value	Equation
$P_{\mathrm{D}}(\boldsymbol{m}_k^j \mid \hat{n}(\boldsymbol{m}_k^j, \boldsymbol{x}_k))$	See Figure 4	(22)
λ	0.06	(37)
$P_{\mathrm{FA}}(\boldsymbol{z}_k^l \mid \boldsymbol{x}_k)$	2.34%	(23)
μ	$[52.25\ 52.98]^{\mathsf{T}}$	(39)
Σ	$\begin{bmatrix} 602.10 & 628.97 \\ 628.97 & 664.65 \end{bmatrix}$	(40)
$\lambda_{n_{\mathrm{d}}}$	0.041	(42)

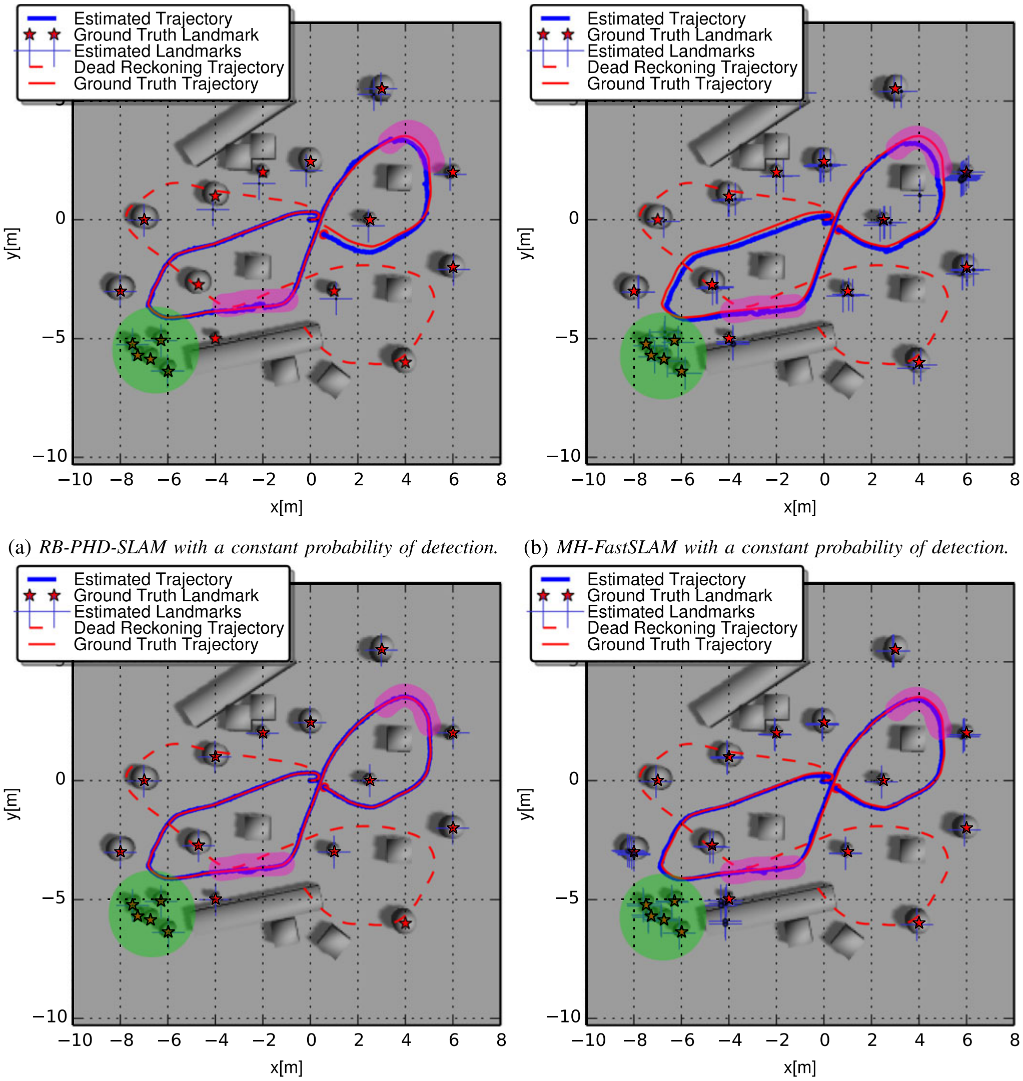
using known false alarms from the same simulated dataset. Figure 8 shows the histogram for the number of false alarms, along with an equivalent exponential distribution

$$g_{FA}^{\boldsymbol{\theta}_{\text{vol}}}(\boldsymbol{\theta}) = \lambda_{n_{\mathrm{d}}} \exp\left(-\lambda_{n_{\mathrm{d}}}(n_{\mathrm{d}} - N_{\min})\right), \qquad (42)$$

where, from Figure 8, $\lambda_{n_{\mathrm{d}}} = 0.041$ and $N_{\min} = 13$. The results of applying this descriptor in simulated SLAM trials will be shown in the Simulated SLAM Results section.

SIMULATED SLAM RESULTS

Table 1 shows the parameters used in the simulations. Figure 9 presents the results of running the algorithms with and without the proposed detection statistics and descriptor. The ground truth map features are displayed using red stars and the estimated map features using blue crosses. The ground



(a) *RB-PHD-SLAM with a constant probability of detection.*

(b) *MH-FastSLAM with a constant probability of detection.*

(c) *RB-PHD-SLAM with the learned probability of detection and descriptor information.*

(d) *MH-FastSLAM with the learned probability of detection and descriptor information.*

Fig. 9–SLAM results in a simulated environment. Highlighted in green: area where the detector performance significantly differs. Highlighted in pink: sections of trajectory where only a single feature was in the FoV of the robot's sensor. [Color figure can be viewed at wileyonlinelibrary.com and www.ion.org] truth trajectory is shown with a red line and the estimated trajectory with a continous blue line. The red dashed line shows the odometry input (dead reckoning). A satisfactory SLAM solution should estimate both the map and trajectory accurately; hence, ideally, the red stars and blue crosses and the red and blue lines should coincide. The background of each graph is a top-down view of the simulated environment.

Figure 9(a) and (b) shows the results of using a constant probability of detection of 0.7 within the sensors FoV for RB-PHD-SLAM and MH-FastSLAM, respectively.

Figure 9(c) and (d) includes both the proposed detection statistics from (22) and Figure 4, as well as the proposed descriptor likelihoods from (41) and (42) for RB-PHD-SLAM and MH-FastSLAM, respectively.

Including the proposed detection statistics and descriptor information improves the SLAM estimates for RB-PHD-SLAM and MH-FastSLAM. An area where the detector produced false measurements is highlighted in green in the figures. It can be seen that both algorithms, when based on a constant probability of detection, fail to map the area correctly, with RB-PHD-SLAM missing two features, and MH-FastSLAM including many false alarms. Only by adding both the descriptor and detection statistics is MH-FastSLAM able to map the area correctly. Sections of the robot's trajectory where only a single feature was visible are highlighted as a pink shaded area. In these sections, both algorithms provide improved trajectory estimation when using the descriptor information. From Figure 9(a) and (b), it can be seen that MH-FastSLAM has more trouble at the initial sharp 'U' turn (close to the origin). This can be explained by the fact that the PHD filter is quick to initialize new estimates, allowing it to deal with the initial sharp turn. Conversely, when a feature's probability of detection is nonzero, the PHD filter can remove features too quickly, if not sensed. This explains why two of the features in the green area are never correctly mapped at the end of each run.

Figures 10 and 11 show the trajectory and mapping errors, respectively. The errors show similar results as those observed in Figure 9. In particular, the inclusion of the detection statistics improves the results in the first part of the trajectory (up to approximately 1300[s]). After this time, the inclusion of the detection statistics without using the descriptor in MH-FastSLAM causes the algorithm to diverge more quickly than the algorithm using a constant probability of detection. Importantly, both algorithms have a significantly improved trajectory when including both the descriptor and the variable probability of detection. Table 2 shows the average trajectory error. The results are based on five Monte Carlo runs of each algorithm.

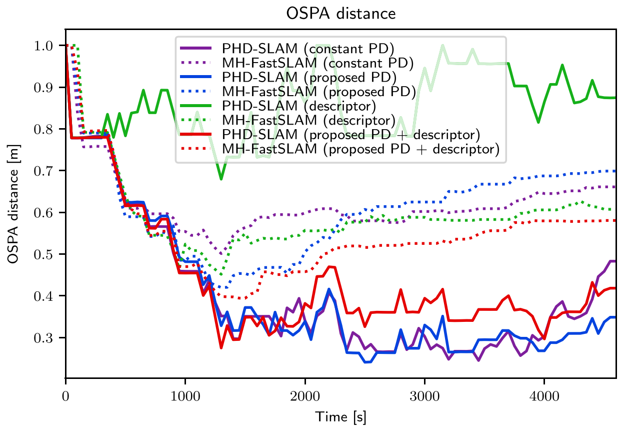


Fig. 10–Average OSPA errors between the ground truth map and the map estimates. Results are based on the detection statistics learned in the same environment (Figure 3(a)). Errors were averaged over five independent runs. OSPA parameters used were c=1 and p=1. [Color figure can be viewed at wileyonlinelibrary.com and www.ion.org]

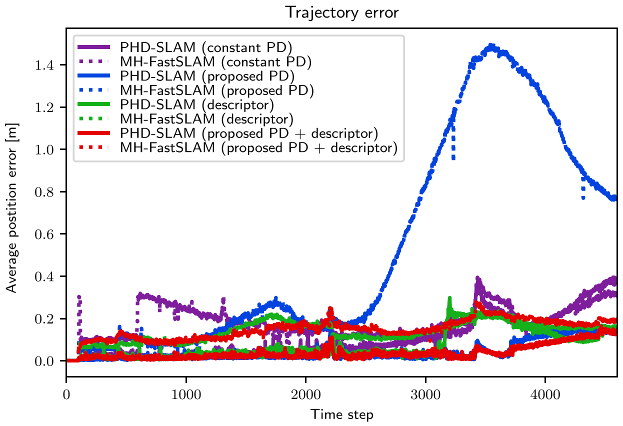


Fig. 11–Average position error over time. Results are based on the detection statistics learned in the same environment (Figure 3(a)). Errors were averaged over five independent runs. [Color figure can be viewed at wileyonlinelibrary.com and www.ion.org]

Using a different training dataset

To test the generality of using the proposed descriptor, a second training dataset has been used to produce different detection statistics. Figure 3 shows the environments used. These two environments differ in terms of the radii of the circles and their spacing. The spacing of the cylinders is important since, for example, closely located cylinders can cause the detector to generate fewer circles than are actually present, with false centers and radii. For example, the closely spaced cylinders in the lower left corner of Figure 3(a) result in the false alarms within the green circles in Figure 9(b). This experiment shows that the use of both detection statistics and descriptor information modeled from a second environment also improves the SLAM performance. From this, it can be concluded that the proposed detection statistics and descriptor are not

Table 2 — Average trajectory error of the simulated results, based on the detection statistics learned in the same environment (Figure 3(a))

Algorithm	Detection Statistics	Average Trajectory Error [m]	
		No descriptor	Using Descriptor
PHD-SLAM	Constant P_{D}	0.11	0.07
	Variable $P_{\mathrm{D}}\left(\boldsymbol{m}_k^j \mid \hat{n}\left(\boldsymbol{m}_k^j, \boldsymbol{x}_k\right)\right)$	0.04	0.04
MH-FastSLAM	Constant P_{D}	0.18	0.14
	Variable $P_{\mathrm{D}}\left(\boldsymbol{m}_k^j \mid \hat{n}\left(\boldsymbol{m}_k^j, \boldsymbol{x}_k\right)\right)$	0.54	0.15

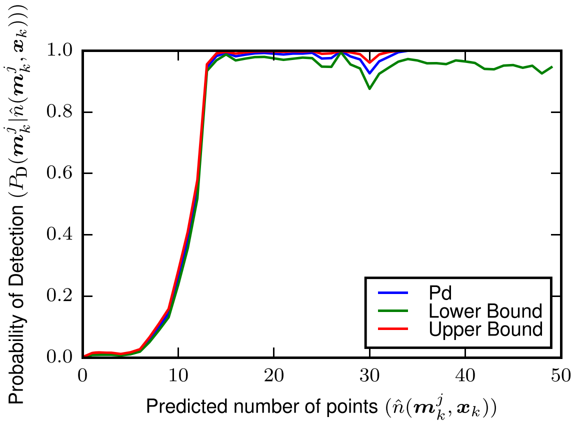


Fig. 12–The probability of detection as a function of the number of unoccluded points $\hat{n}\left(\boldsymbol{m}_k^j, \boldsymbol{x}_k\right)$. Based on the number of points available in the test dataset, a 99% confidence interval is calculated, the bounds of which are shown. [Color figure can be viewed at wileyonlinelibrary.com and www.ion.org]

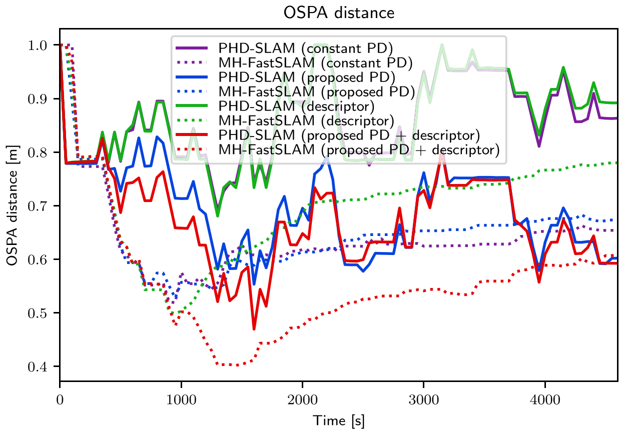


Fig. 13–Average OSPA errors between the ground truth map and the map estimates. Results are based on the detection statistics learned in a different environment (Figure 3(b)). Errors were averaged over five independent runs. OSPA parameters used were c=1 and p=1. [Color figure can be viewed at wileyonlinelibrary.com and www.ion.org]

merely ‘overfitting’ to the particulars of the specific environment.

Figure 12 shows the probability of detection obtained from the training dataset of Figure 3(b). As can be seen from the figure, the transition from close to zero probability of detection to a high probability of detection occurs at approximately the same value $\hat{n}(\boldsymbol{m}_k^j, \boldsymbol{x}_k) = 13$, but the values of the probability of detection are significantly different. The SLAM algorithms were then rerun based on the detection statistics shown in Figure 12.

As can be seen in Figure 13, the mapping errors of both algorithms improve when applying both descriptor information and detection statistics, as resulted when the statistics were calculated from the same environment. The performance of MH-FastSLAM does not improve when using either the detection statistics or the descriptor information individually. It does however improve when both are used. On the other hand, PHD-SLAM benefits mainly from the inclusion of the variable detection statistics, while the addition of the descriptor only produces a small improvement in the mapping accuracy.

Figure 14 shows the trajectory errors obtained using the new detection statistics. The trajectory error plots clearly display the problems PHD-SLAM has when lacking the detection statistics. The errors are reduced with the addition of the variable probability of detection, while in the case of the trajectory error, the further addition of the descriptor makes little difference. As with the mapping error, MH-FastSLAM benefits less than when using the environment-specific detection statistics. With the addition of both the descriptor and detection statistics, the pose error is still lowest for most of the trajectory, as was shown when using the environment specific detection statistics. Table 3 confirms these conclusions in terms of the average errors over the entire trajectory. The appendix shows the results of the same algorithms used to generate Figure 9 based on the new training environment in Figure A1. A single SLAM result is shown for each algorithm out of the five Monte Carlo runs used to generate Table 3.

LEARNING FROM A PARK ENVIRONMENT

This section applies the concepts described in the Learning from Simulated Range Data section to data obtained in a real park environment. The method explained is valid for any shape detector to

Table 3 — Average trajectory error of the simulated results, based on the detection statistics learned in a different environment (Figure 3(b))

Algorithm	Detection Statistics	Average Trajectory Error [m]	
		No descriptor	Using Descriptor
PHD-SLAM	Constant P_{D}	0.08	0.09
	Variable $P_{\mathrm{D}}(\boldsymbol{m}_k^j\,\vert\,\hat{n}(\boldsymbol{m}_k^j,\boldsymbol{x}_k))$	0.04	0.05
MH-FastSLAM	Constant P_{D}	0.18	0.21
	Variable $P_{\mathrm{D}}(\boldsymbol{m}_k^j\,\vert\,\hat{n}(\boldsymbol{m}_k^j,\boldsymbol{x}_k))$	0.20	0.15

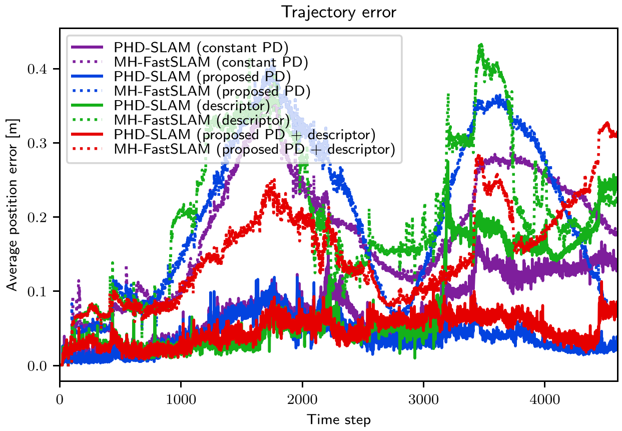


Fig. 14–Average position error over time. Results are based on the detection statistics learned a different environment (Figure 3(b)). Errors were averaged over five independent runs. [Color figure can be viewed at wileyonlinelibrary.com and www.ion.org]

which ray casting can be performed. A simple circular feature detector, based on the detector described by Guivant et al. in [34], is applied here. This detector first segments the laser scan into clusters, using a simple distance threshold and then fits a circle to each cluster, via mean squared error minimization. Further details of the detector can be found in [6]. The dataset used to learn $P_{\mathrm{D}}\left(\boldsymbol{m}_k^j\,|\,\hat{n}\left(\boldsymbol{m}_k^j,\boldsymbol{x}_k\right)\right)$ and $\kappa\left(\boldsymbol{z}_k^i\,|\,\boldsymbol{x}_k\right)$ was recorded in a park near the Universidad de Chile. The dataset consisted of 2,397 scans taken from 19 known vehicle poses. The area covered by the dataset contained 174 tree trunks and lamp posts, the number and locations of which were verified by independent means to form ground truth. Naturally, the generality of such an environment is questionable. In general, any semantic feature detection statistics can be determined in a manner similar to methods given in the following subsections but should be based on datasets from environments known to contain a significant number of the type of semantic feature sought.

Estimating Detection Probability $\mathbf{P}_{D}\left(\boldsymbol{m}_{\mathbf{k}}^{\mathbf{j}}|\hat{\mathbf{n}}\left(\boldsymbol{m}_{\mathbf{k}}^{\mathbf{j}},\boldsymbol{x}_{\mathbf{k}}\right)\right)$

Figure 15 shows the results of applying (22) to the dataset, where the probability of detecting a circular object (in this case a tree) is highly dependent on the number of unoccluded points. (Note that the values of the probability in Figure 15 differ significantly from those in Figure 4. This is because the feature detector used in the park can detect features using fewer range points than the RANSAC detector, which needs a relatively large number of inliers to differentiate detections from any other randomly generated circle. Further, the RANSAC detector was used in a simulated environment with a lower clutter rate.) The figure also shows a 99% confidence interval, based on the number of data points available for each value of $\hat{n}\left(\boldsymbol{m}_k^j,\boldsymbol{x}_k\right)$. It should be noted that there is less data corresponding to features that have a high number of unoccluded points. Therefore, for such features, the uncertainty in the estimated value of $P_{\mathrm{D}}\left(\boldsymbol{m}_k^j\,|\,\hat{n}\left(\boldsymbol{m}_k^j,\boldsymbol{x}_k\right)\right)$ is higher, as shown by the confidence interval.

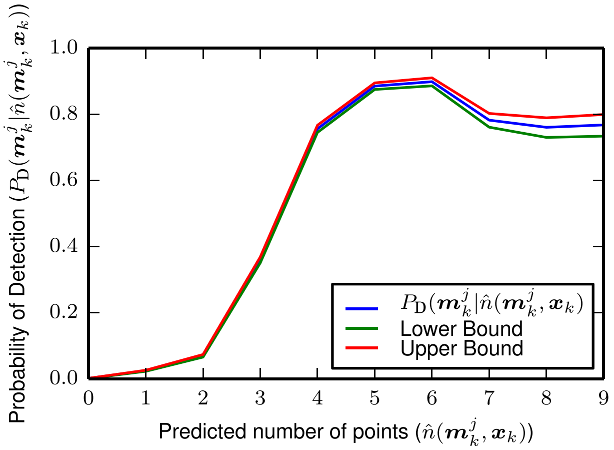


Fig. 15–The probability of detection as a function of the number of unoccluded points $\hat{n}\left(m_k^j,x_k\right)$. Based on the number of points available in the test dataset, a 99% confidence interval is calculated, the bounds of which are shown. [Color figure can be viewed at wileyonlinelibrary.com and www.ion.org]

Estimating $\mathbf{P}_{FA}\left(\boldsymbol{z}_{\mathbf{k}}^{\mathbf{l}}|\boldsymbol{x}_{\mathbf{k}}\right)$ and $\kappa\left(\boldsymbol{z}_{\mathbf{k}}^{\mathbf{i}}|\boldsymbol{x}_{\mathbf{k}}\right)$

Similar to the explanation in Estimating $P_{\mathrm{FA}}\left(\boldsymbol{z}_k^l\,|\,\boldsymbol{x}_k\right)$ and Intensity Function $\kappa\left(\boldsymbol{z}_k^i\,|\,\boldsymbol{x}_k\right)$ section, for the case of the park dataset, the his-

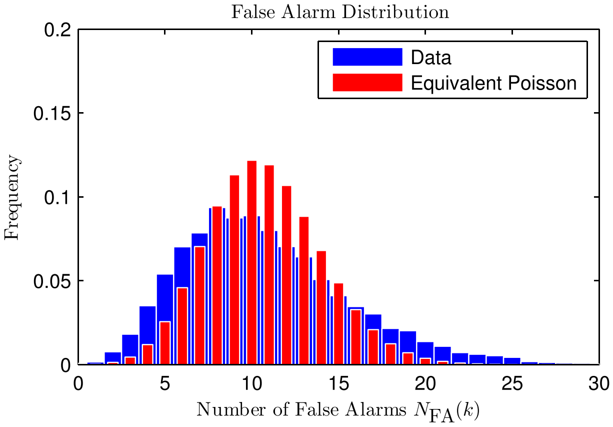


Fig. 16–A distribution of the number of false alarms per scan, obtained from the test dataset. The red graph shows the equivalent Poisson distribution. Note that the red bars are superimposed on to the blue bars, with half their width, to improve visibility. [Color figure can be viewed at wileyonlinelibrary.com and www.ion.org]

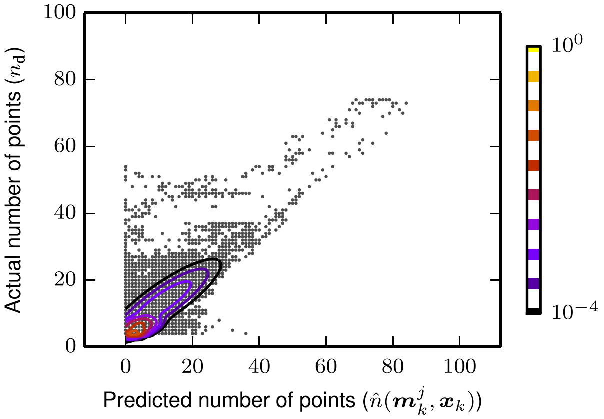


Fig. 17–Number of measured points n_d as a function of the predicted number of points $\hat{n}(\boldsymbol{m}_k^j, \boldsymbol{x}_k)$. The contours of an equivalent Gaussian mixture (composed of four Gaussian components) are also shown. [Color figure can be viewed at wileyonlinelibrary.com and www.ion.org]

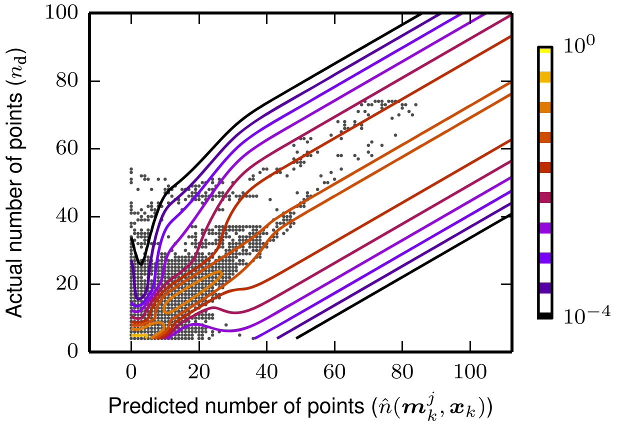


Fig. 18–The conditional distribution of n_d given $\hat{n}\left(\boldsymbol{m}_k^j, \boldsymbol{x}_k\right)$ calculated from the Gaussian mixture of Figure 17. [Color figure can be viewed at wileyonlinelibrary.com and www.ion.org]

togram of the number of false alarms per scan is shown in Figure 16, along with its equivalent theoretical Poisson distribution (with parameter $\lambda = 9.3$) plotted in red. $\kappa\left(\boldsymbol{z}_k^i \mid \boldsymbol{x}_k\right)$ was again determined from (37), while $P_{\mathrm{FA}}\left(\boldsymbol{z}_k^l \mid \boldsymbol{x}_k\right)$ was determined from (23) to be 47.19%.

Estimating Feature Likelihoods $g_{\boldsymbol{\theta}}^{\boldsymbol{\theta}\,vol}(\boldsymbol{\theta})$ *and* $g_{\mathrm{FA}}^{\boldsymbol{\theta}\,vol}(\boldsymbol{\theta})$

As in the Estimating Feature Likelihoods $g_{\boldsymbol{\theta}}^{\boldsymbol{\theta}\,\mathrm{vol}}(\boldsymbol{\theta})$ and $g_{FA}^{\boldsymbol{\theta}\,\mathrm{vol}}(\boldsymbol{\theta})$ section, to model the distribution of the number of range values used to generate a detection, given $p\left(n \mid \hat{n}\left(\boldsymbol{m}_k^j, \boldsymbol{x}_k\right)\right)$, first the joint probability distribution $p\left(n, \hat{n}\left(\boldsymbol{m}_k^j, \boldsymbol{x}_k]\right)\right)$ is modeled, shown in Figure 17.

In contrast to the simulated range data analyzed in Estimating Feature Likelihoods $g_{\boldsymbol{\theta}}^{\boldsymbol{\theta}\,\mathrm{vol}}(\boldsymbol{\theta})$ and $g_{FA}^{\boldsymbol{\theta}\,\mathrm{vol}}(\boldsymbol{\theta})$ section, a single Gaussian distribution would not accurately model the data in Figure 17; therefore, a Gaussian mixture

$$p\left(n, \hat{n}\left(\boldsymbol{m}_k^j, \boldsymbol{x}_k\right)\right) = \sum_i w_i \mathcal{N}\left(n, \hat{n}\left(\boldsymbol{m}_k^j, \boldsymbol{x}_k\right) \mid \mu_i, \Sigma_i\right) \tag{43}$$

is used. To estimate the Gaussian mixture parameters, the Expectation Maximization (EM) [35] algorithm was then applied, based on a predetermined (in this case selected to be four to avoid overfitting the data) number of Gaussian components. A contour plot of the result is also shown in Figure 17. Using the estimated Gaussian mixture, the conditional distribution can be obtained as

$$p\left(n \mid \hat{n}\left(\boldsymbol{m}_k^j, \boldsymbol{x}_k\right)\right) = \frac{p\left(n, \hat{n}\left(\boldsymbol{m}_k^j, \boldsymbol{x}_k\right)\right)}{p\left(\hat{n}\left(\boldsymbol{m}_k^j, \boldsymbol{x}_k\right)\right)} \tag{44}$$

$$= \frac{p\left(n, \hat{n}\left(\boldsymbol{m}_k^j, \boldsymbol{x}_k\right)\right)}{\int p\left(n, \hat{n}\left(\boldsymbol{m}_k^j, \boldsymbol{x}_k\right)\right) dn}. \tag{45}$$

By substituting (43) into (44), the conditional distribution becomes

$$\begin{aligned} g_{\boldsymbol{\theta}}^{\boldsymbol{\theta}\,\mathrm{vol}}(\boldsymbol{\theta}) &= p\left(n \mid \hat{n}\left(\boldsymbol{m}_k^j, \boldsymbol{x}_k\right)\right) \\ &= \frac{\sum_i w_i \mathcal{N}\left(n, \hat{n}\left(\boldsymbol{m}_k^j, \boldsymbol{x}_k\right) \mid \mu_i, \Sigma_i\right)}{\sum_i w_i \mathcal{N}\left(\hat{n}\left(\boldsymbol{m}_k^j, \boldsymbol{x}_k\right) \mid \mu_i(2), \Sigma_i(2,2)\right)}, \end{aligned} \tag{46}$$

where $\mu_i(2)$ and $\Sigma_i(2,2)$ are the elements of the mean and covariance matrix, respectively, which correspond to $\hat{n}\left(\boldsymbol{m}_k^j, \boldsymbol{x}_k\right)$. Figure 18 shows a con-

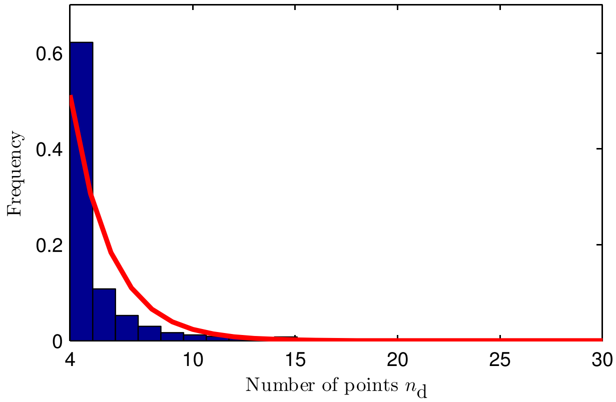


Fig. 19–Descriptor distribution for false alarms (blue) and its equivalent exponential distribution (red). [Color figure can be viewed at wileyonlinelibrary.com and www.ion.org]

tour plot of the conditional distribution calculated using (46) corresponding to the Gaussian mixture of Figure 17.

For the case of false alarms, the distribution of n_{d}, based on known false alarms from the park environment and shown in blue in Figure 19, was modeled with an exponential distribution

$$g_{FA}^{\boldsymbol{\theta}_{\mathrm{vol}}}(\boldsymbol{\theta}) = \lambda_{n_{\mathrm{d}}} \exp\left(-\lambda_{n_{\mathrm{d}}}\left(n_{\mathrm{d}} - N_{\mathrm{min}}\right)\right), \qquad (47)$$

with parameters $\lambda_{n_{\mathrm{d}}} = 0.5124$ and $N_{\mathrm{min}} = 4$ (red curve). The results of applying this descriptor in SLAM will be shown in the Experimental SLAM Results section.

EXPERIMENTAL SLAM RESULTS

The performance utilizing the probability of detection and descriptor information in Parque O'Higgins, Santiago and with the publicly available Victoria Park dataset [34] from Sydney, Australia, is evaluated in the Experiments in Parque O'Higgins, Santiago and Victoria Park dataset sections, respectively. Table 4 shows the parameters used, which are common to both datasets.

Experiments in Parque O'Higgins, Santiago

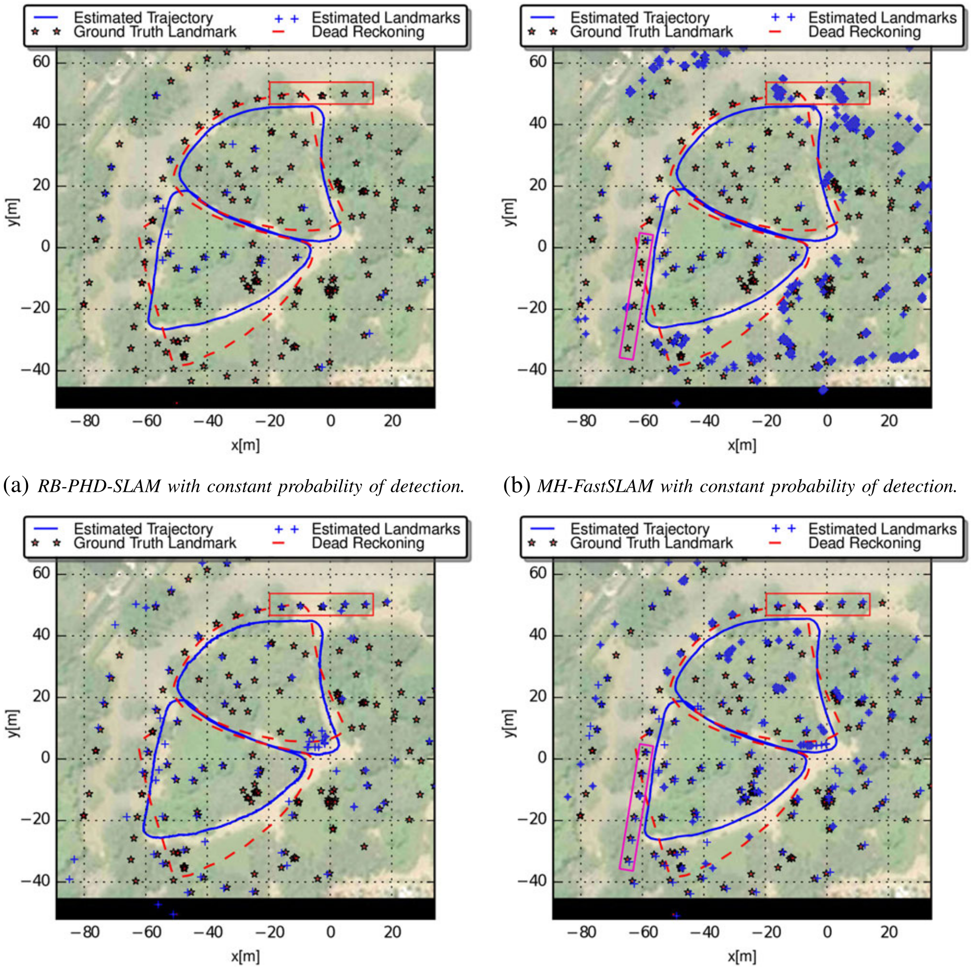
The robotic platform for collecting the experimental dataset was a Clearpath Husky A-200 robot equipped with a Sick LD-LRS-1000 laser range finder, with a reported maximum range of 80 m at 10% reflectivity. The Husky's wheel encoders provided odometry measurements, $\boldsymbol{u}_k$, used in the motion model in (1). The experiments were conducted in the same environment where the detection statistics were determined, albeit with a different dataset. Although not a very general result, this provides the best estimate of the real detection statistics that will be encountered in the experiments.

Table 4 — The parameters used in the SLAM experiments

Parameter	Value	Equation
$P_{\mathrm{D}}(\boldsymbol{m}_k^j \mid \hat{n}(\boldsymbol{m}_k^j, \boldsymbol{x}_k))$	See Figure 15	(22)
λ	9.3	(37)
$P_{\mathrm{FA}}(\boldsymbol{z}_k^l \mid \boldsymbol{x}_k)$	47.19%	(23)
ω_1	0.41	(46)
μ_1	$[2.75\ 4.32]^{\mathrm{T}}$	(46)
Σ_1	$\begin{bmatrix} 2.31 & 0.24 \\ 0.24 & 0.22 \end{bmatrix}$	(46)
ω_2	0.18	(46)
μ_2	$[10.96\ 13.38]^{\mathrm{T}}$	(46)
Σ_2	$\begin{bmatrix} 52.82 & 33.44 \\ 33.44 & 27.93 \end{bmatrix}$	(46)
ω_3	0.03	(46)
μ_3	$[16.01\ 22.42]^{\mathrm{T}}$	(46)
Σ_1	$\begin{bmatrix} 195.57 & 112.94 \\ 112.94 & 177.9 \end{bmatrix}$	(46)
ω_4	0.38	(46)
μ_4	$[3.89\ 6.23]^{\mathrm{T}}$	(46)
Σ_4	$\begin{bmatrix} 5.96 & 1.12 \\ 1.12 & 1.48 \end{bmatrix}$	(46)
$\lambda_{n_{\mathrm{d}}}$	0.5124	(42)

The robot ended its trajectory at approximately the same position as it started. The results for the constant probability of detection versions of RB-PHD-SLAM and MH-FastSLAM are shown in Figure 20(a) and (b), in which optimal performance in each case occurred for $P_{\mathrm{D}}(\boldsymbol{m}_k^j \mid \boldsymbol{x}_k) = 0.7$ and 0.5, respectively. The versions of RB-PHD-SLAM and MH-FastSLAM that use both the variable probability of detection (from (22) and Figure 15) and the feature descriptor information (from (46) and (47)) are shown in Figure 20(c) and (d), respectively.

In the case of RB-PHD-SLAM, visually, the map estimates of both the proposed algorithms appear to be better than the algorithm using a constant probability of detection, which removes most features from the map as soon as they stop being detected. This is because of the probable mismatch between the assumed constant probability of detection and the actual probability of detection. This effect can be observed particularly in the section of the maps highlighted with a red rectangle. In this rectangle, the algorithm which assumes a constant probability of detection, completely removes the estimates from its map. This occurs because the Gaussians that represent the estimates are multiplied many times by a factor of $(1 - P_{\mathrm{D}}(\boldsymbol{m}_k^j \mid \boldsymbol{x}_k))$, with a falsely high probability of detection (see the first term in the RHS of (14)). Even with an accurate probability of detection, the PHD filter is too quick to remove landmarks, which only accentuates the problem. However, by using the proposed detection statistics, the filter



(a) *RB-PHD-SLAM with constant probability of detection.*

(b) *MH-FastSLAM with constant probability of detection.*

(c) *RB-PHD-SLAM with learned probability of detection and descriptor information.*

(d) *MH-FastSLAM with learned probability of detection and descriptor information.*

Fig. 20–SLAM results in Parque O'Higgins, Santiago, Chile. Red rectangle: Area of the map that shows most improvement in RB-PHD-SLAM and MH-FastSLAM by including detection statistics. Pink rectangle: Area of the map that shows most improvement in MH-FastSLAM by using the detection statistics. [Color figure can be viewed at wileyonlinelibrary.com and www.ion.org]

Table 5 — Average error in the final position (averaged over five Monte Carlo runs)

	Average Final position error	
	PHD-SLAM (m)	MH-FastSLAM (m)
Constant $P_{\mathrm{D}}(\boldsymbol{m}_k^j \mid \boldsymbol{x}_k)$	1.36	1.07
Proposed $P_{\mathrm{D}}(\boldsymbol{m}_k^j \mid \boldsymbol{x}_k)$	1.33	0.68
Proposed $P_{\mathrm{D}}(\boldsymbol{m}_k^j \mid \boldsymbol{x}_k)$ + using descriptor	0.53	0.67

is able to maintain the estimates. The trajectory estimate of the constant probability RB-PHD-SLAM filter (Figure 20(a)) is also worse than its variable $P_{\mathrm{D}}(\boldsymbol{m}_k^j \mid \boldsymbol{x}_k)$ counterpart, as the pose estimate does not exactly return to its starting position, as shown in Table 5.

Figure 20(d), when compared with Figure 20(b), shows that with MH-FastSLAM, the proposed probability of detection and descriptor improves the map estimates, by allowing the filter to maintain the estimates of features when they are occluded. This is particularly noticeable in the row of trees located at approximately x = –60 m (pink rectangle), where the map estimates corresponding to these trees are maintained, while MH-FastSLAM based on a constant probability of detection failed to do so (Figure 20(b)). This could be explained by the fact that these estimates are occluded by other trees

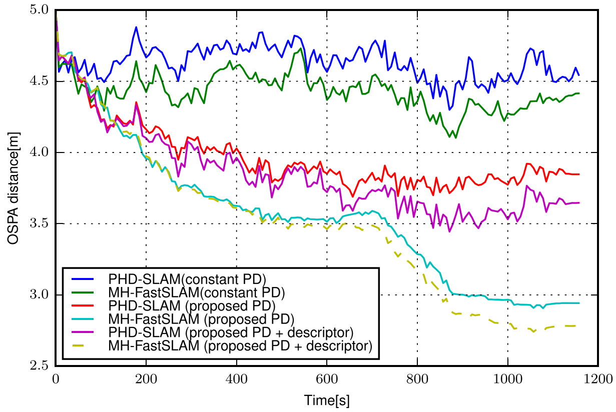


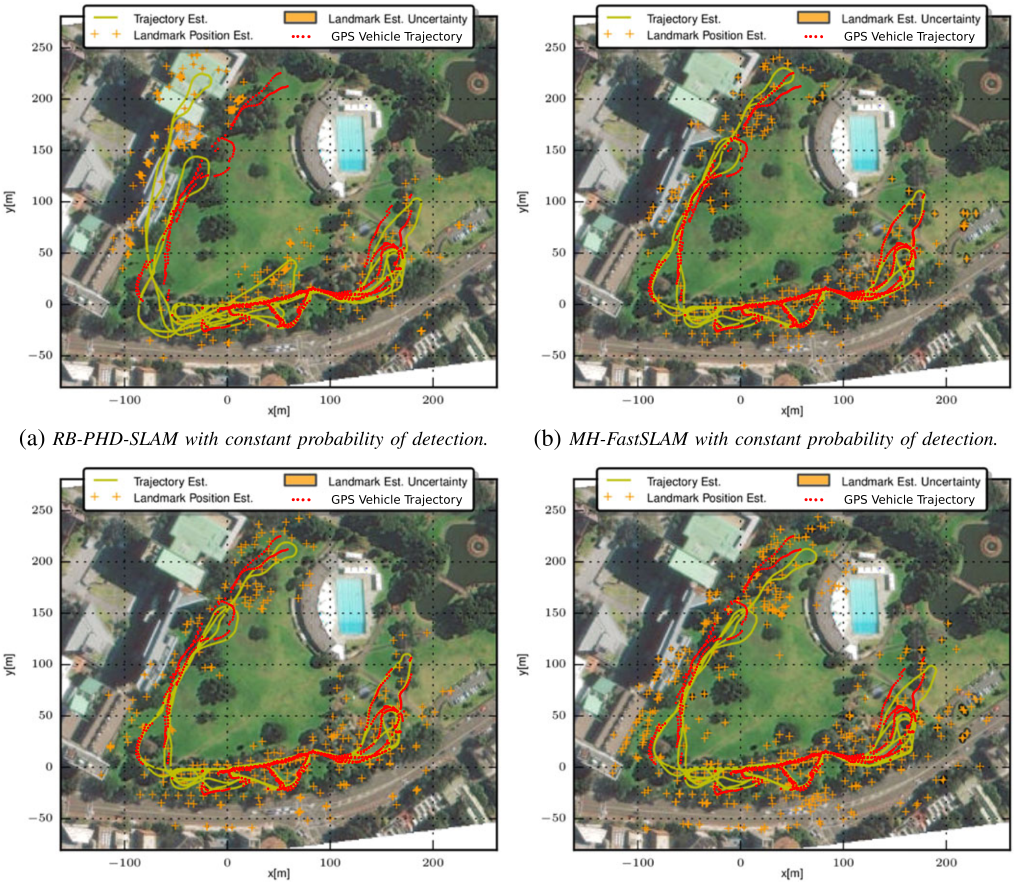
Fig. 21–Average OSPA errors between the ground truth map and the map estimates. Errors were averaged over five different runs. OSPA parameters used were c=5 and p=1. [Color figure can be viewed at wileyonlinelibrary.com and www.ion.org]

for a long time before they exit the theoretical FoV. The effect of the variable $P_{\mathrm{D}}\left(\boldsymbol{m}_k^j|\boldsymbol{x}_k\right)$ on the trajectory is less severe, but as can be seen in Table 5, the final point of the trajectory is closer to the starting point. Including descriptor information into MH-FastSLAM (Figure 20(d)) removes most of the clusters of false features and improves the trajectory estimate.

To quantify the mapping errors in each case, Figure 21 shows the OSPA errors between the estimated and ground truth maps, giving greater clarity of the mapping performance of each algorithm. In the case of both RB-PHD-SLAM and MH-FastSLAM, the metric confirms the higher map quality produced by employing the variable probability of detection and the further improved performance when using feature descriptor information. It should be noted that the focus here is to demonstrate the improvement in both RB-PHD-SLAM and MH-FastSLAM when feature descriptor information is used, and not to compare RB-PHD-SLAM and MH-FastSLAM. Although under the relatively low clutter conditions used in the SLAM experiments here, MH-FastSLAM out performs RB-PHD-SLAM, the converse is true under higher clutter conditions [36].

Victoria Park dataset

To confirm the results obtained using the dataset gathered in Parque O'Higgins, the Victoria Park



(a) *RB-PHD-SLAM with constant probability of detection.*

(b) *MH-FastSLAM with constant probability of detection.*

(c) *RB-PHD-SLAM with learned probability of detection and descriptor information.*

(d) *MH-FastSLAM with learned probability of detection and descriptor information.*

Fig. 22–SLAM results in Victoria Park, Sydney, Australia. [Color figure can be viewed at wileyonlinelibrary.com and www.ion.org] benchmark dataset was used [34]. The Victoria Park dataset is similar to the Parque O'Higgins dataset, in the sense that trees are the natural features of the environment. This dataset provides the opportunity to test the generality of the detection statistics and descriptor likelihoods learned based on data from Parque O'Higgins with a dataset taken by different researchers in a different location. Additionally, these statistics will be tested on the detector provided with the dataset [34], which was only modified to provide the number of points used in each detection. This detector, although it also detects circular cross-sectioned objects, is different from the detector used to gather the statistics in Parque O'Higgins.

Figure 22(a) and (b) shows the results of the standard RB-PHD-SLAM and MH-FastSLAM filters, with constant values $P_{\mathrm{D}}(\boldsymbol{m}_k^j|\boldsymbol{x}_k) = 0.4$ and 0.7, respectively, determined by trial and error, to optimize filter performance in each case. The 'forgetful' nature of RB-PHD-SLAM is evident, due to the probable mismatch in the estimated and actual probabilities of detection, causing it to diverge (Figure 22(a)).

Figure 22(c) and (d) shows the results of RB-PHD-SLAM and MH-FastSLAM, respectively, using both the values of $P_{\mathrm{D}}\left(\boldsymbol{m}_k^j|\boldsymbol{x}_k\right)$ descriptor likelihoods learned in Parque O'Higgins. The solution shown in Figure 22(c) converges to trajectories similar to other published SLAM solutions [9, 21]. Interestingly, both trajectories without the use the proposed methods have errors towards the building in the left part of Figure 22(a) and (b). The variable probability of detection enables the filters to use negative information properly and any errors in the trajectory place the vehicle in an accessible location (the park) rather than inside the building.

In the case of MH-FastSLAM, both versions converge to approximately the same trajectory, although using the learned detection statistics produced a map with many more features, while adding the descriptor removed a few of the added features. Due to the lack of map ground truth information and the partial absence of trajectory GPS information, it is difficult to judge the performance of the proposed modifications in this dataset. It should be noted, however, that in the satellite image, some of the additionally estimated features in Figure 22(d) correspond to real trees, specifically in the lower right corner of the image, across the street from the park location.

SUMMARY

The importance of detection as well as the usually considered spatial statistics upon SLAM performance was demonstrated in this article. Insights were derived from the target tracking community, in which the amplitude of radar data has been used as a target descriptor to provide detection statistics and likelihoods, to accompany radar range values. This concept was extended to give a generalized robotic map feature descriptor for range measuring sensors, based on the state dependent expected number of unoccluded points, available via ray casting. The difference between the actual and predicted number of range points received from a feature proved to be a useful approximation of a sufficient statistic to aid feature based SLAM, while accounting for map feature occlusions. SLAM simulations, with known map ground truth as well as outdoor SLAM experiments, verified the importance of including detection statistics into SLAM algorithms. It was particularly encouraging that the generation of detection statistics for SLAM maps based on tree locations, learned in one park location (Santiago, Chile), was able to provide improvements to SLAM algorithms using a dataset and a different detector in another location (Sydney, Australia). The SLAM algorithms tested were vector-based MH-FastSLAM in conjunction with a binary Bayes filter, which utilized the derived detection statistics, and RFS-based RB-GM-PHD-SLAM, which incorporates the statistics directly into its set state-based Bayesian recursion.

ACKNOWLEDGEMENTS

The authors acknowledge 'Becas Conicyt - Doctorado Nacional, 2014,' Conicyt - Fondecyt project 1150930, the Advanced Mining Technology Center (AMTC), and Clearpath Robotics, Canada. The authors also thank J. Guivant and E. Nebot for the publication of the Victoria Park dataset. This material is based upon work supported by the Air Force Office of Scientific Research under award number FA9550-17-1-0386.

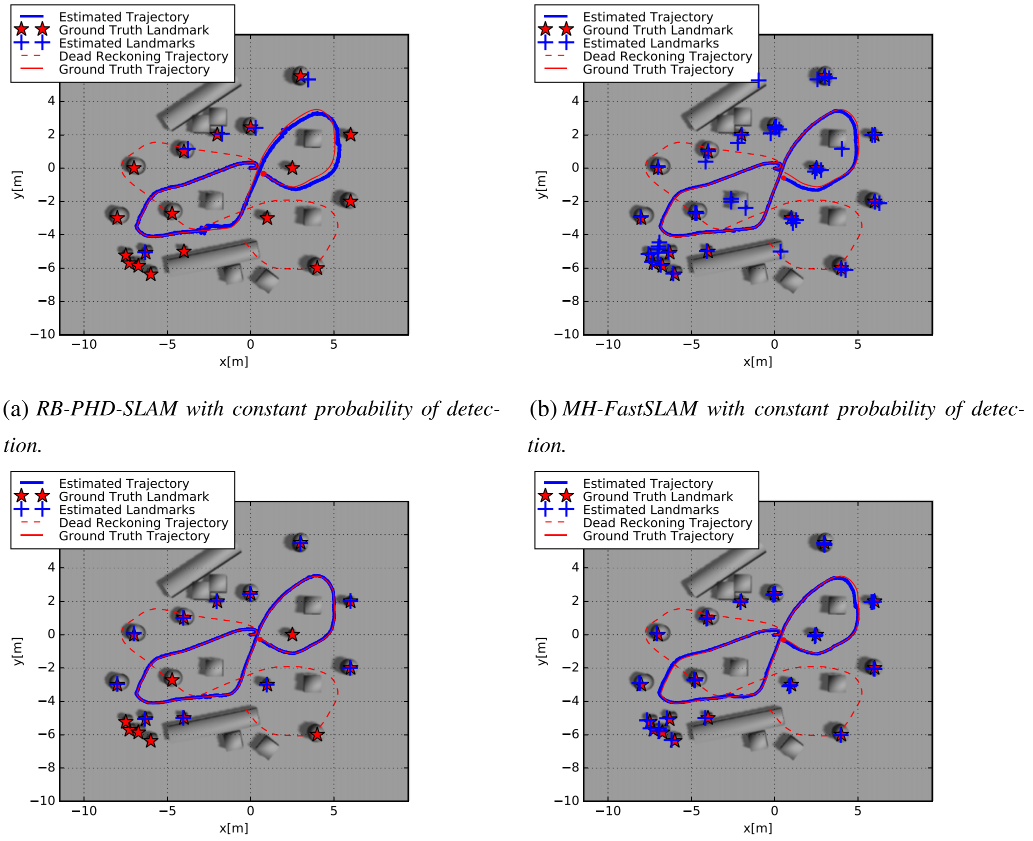
REFERENCES

1. Mahler, R., "A Survey of PHD Filter and CPHD Filter Implementations," *Proceedings of SPIE Defense & Security Symposium of Signal Processing, Sensor Fusion and Target Recognition XII*, 2007.
2. Thrun, S., Burgard, W., and Fox, D., *Probabilistic Robotics,* Vol. 1: MIT press Cambridge, 2005.
3. Durrant-Whyte, H. and Bailey, T., "Simultaneous Localization and Mapping: Part I," *IEEE Robotics & Automation Magazine*, Vol. 13, No. 2, 2006, pp. 99–110.
4. Leung, K., Inostroza, F., and Adams, M., "Generalizing Random-Vector SLAM with Random Finite Sets," *IEEE International Conference on Robotics and Automation (ICRA)*, 2015, pp. 4583–4588.
5. Barkat, M., *Signal Detection and Estimation*: Artech House Boston, 2005.
6. Inostroza, F., Leung, K., and Adams, M., "Semantic Feature Detection Statistics in Set Based Simultaneous Localization and Mapping," *IEEE 17th International Conference on Information Fusion*, 2014.
7. Inostroza, F., Leung, K., and Adams, M., "Incorporating Estimated Feature Descriptor Informa-

tion into Rao-Blackwellized-PHD-SLAM," *IEEE 18th International Conference Information Fusion*, 2015, pp. 1688–1695.

8. Fischler, M. A. and Bolles, R. C., "Random Sample Consensus: A Paradigm for Model Fitting with Applications to Image Analysis and Automated Cartography," *Communications of the ACM*, Vol. 24, No. 6, 1981, pp. 381–395.
9. Montemerlo, M., Thrun, S., Koller, D., and Wegbreit, B., "FastSLAM 2.0: An Improved Particle Filtering Algorithms for Simultaneous Localization and Mapping That Provably Converges," *Proceedings of the 18th International Joint Conference on Artificial Intelligence (IJCAI'03)*, 2003, pp. 1151–1156.
10. Mullane, J., Vo, B.-N., Adams, M. D., and Vo, B.-T., "A Random-Finite-Set Approach to Bayesian SLAM," *IEEE Transactions on Robotics*, Vol. 27, No. 2, 2011, pp. 268–282.
11. Lee, C. S., Clark, D., and Salvi, J., "SLAM with Dynamic Targets via Single-Cluster PHD Filtering," *IEEE Journal of Selected Topics in Signal Processing*, Vol. 7, No. 3, 2013, pp. 543–552.
12. Deusch, H., Reuter, S., and Dietmayer, K., "The Labeled Multi-Bernoulli SLAM Filter," *IEEE Signal Processing Letters*, Vol. 22, No. 10, 2015, pp. 1561–1565.
13. Vo, B.-T. and Vo, B.-N., "Labeled Random Finite Sets and Multi-Object Conjugate Priors," *IEEE Transactions on Signal Processing*, Vol. 61, No. 13, 2013, pp. 3460–3475.
14. Erdinc, O., Willett, P., and Bar-Shalom, Y., "The Bin-Occupancy Filter and its Connection to the PHD Filters," *IEEE Transactions on Signal Processing*, Vol. 57, No. 11, 2009, pp. 4232–4246.
15. Elfes, A., "Using Occupancy Grids for Mobile Robot Perception and Navigation," *Computer*, Vol. 22, No. 6, 1989, pp. 46–57.
16. Grisetti, G., Stachniss, C., and Burgard, W., "Improved Techniques for Grid Mapping with Rao-Blackwellized Particle Filter," *IEEE Transactions on Robotics*, Vol. 23, No. 1, 2007, pp. 34–46.
17. Moravec, H. P., "Sensor Fusion in Certainty Grids for Mobile Robots," *AI Magazine*, Vol. 9, No. 2, 1988, p. 61.
18. Mullane, J., Adams, M. D., and Wijesoma, W. S., "Robotic Mapping Using Measurement Likelihood Filtering," *The International Journal of Robotics Research*, Vol. 28, No. 2, 2009, pp. 172–190.
19. Thrun, S. and Montemerlo, M., "The GraphSLAM Algorithm with Applications to Large-Scale Mapping of Urban Structures," *The International Journal of Robotics Research*, Vol. 25, No. 5-6, 2006, pp. 403–429.
20. Kaess, M., Ranganathan, A., and Dellaert, F., "iSAM: Incremental Smoothing and Mapping," *IEEE Transactions on Robotics*, Vol. 24, No. 6, 2008, pp. 1365–1378.
21. Kaess, M., Johannsson, H., Roberts, R., Ila, V., Leonard, J. J., and Dellaert, F., "iSAM2: Incremental Smoothing and Mapping using the Bayes Tree," *The International Journal of Robotics Research*, Vol. 31, No. 2, 2011, pp. 216–235.
22. Sibley, D., Mei, C., Reid, I., and Newman, P., "Adaptive Relative Bundle Adjustment," *Robotics Science and Systems (RSS)*, Vol. 29, No. 8, 2009, pp. 958–980.
23. Agarwal, P., Tipaldi, G., Spinello, L., Stachniss, C., and Burgard, W., "Robust Map Optimization Using Dynamic Covariance Scaling," *IEEE International Conference Robotics and Automation (ICRA)*, 2013, pp. 62–69.
24. Granstrom, K. and Orguner, U., "A PHD Filter for Tracking Multiple Extended Targets Using Random Matrices," *IEEE Transactions on Signal Processing*, Vol. 60, No. 11, 2012, pp. 5657–5671.
25. Reuter, S. and Dietmayer, K., "Pedestrian Tracking using Random Finite Sets," *Proceedings of the IEEE 14th International Conference on Information Fusion (FUSION)*, 2011, pp. 1–8.
26. Clark, D., Ristic, B., Vo, B.-N., and Vo, B.-T., "Bayesian Multi-Object Filtering With Amplitude Feature Likelihood for Unknown Object SNR," *IEEE Transactions on Signal Processing*, Vol. 58, No. 1, 2010, pp. 26–37.
27. Leung, K., Inostroza, F., and Adams, M., "An Improved Weighting Strategy for Rao-Blackwellized Probability Hypothesis Density Simultaneous Localization and Mapping," *Proceedings of the International Conference on Control, Automation, and Information Sciences*, 2013.
28. Mahler, R. P. S., *Statistical Multisource-Multitarget Information Fusion,* Vol. 685: Artech House Boston, 2007.
29. Montemerlo, M., Thrun, S., Koller, D., and Wegbreit, B., "FastSLAM: A Factored Solution to the Simultaneous Localization and Mapping Problem," *Proceedings of the National Conference on Artificial Intelligence*, 2002, pp. 593–598.
30. Nieto, J., Guivant, J., Nebot, E., and Thrun, S., "Real Time Data Association for FastSLAM," *Proceedings of IEEE International Conference on Robotics and Automation (ICRA),* Vol. 1, 2003, pp. 412–418.
31. Adams, M., Vo, B.-N., Mahler, R., and Mullane, J., "New Concepts in Map Estimation: SLAM Gets a PHD," *IEEE Robotics and Automation Magazine*, Vol. 21, No. 2, 2014, pp. 26–37.
32. Skolnik, M., *Introduction to Radar Systems*, New York: McGraw Hill, 1982.
33. Koenig, N. and Howard, A., "Design and Use Paradigms for Gazebo, an Open-Source Multi-Robot Simulator," *IEEE International Conference Intelligent Robots and Systems*, 2004, pp. 2149–2154.
34. Guivant, J. and Nebot, E., "Simultaneous Localization and Map Building: Test Case for Outdoor Applications," *IEEE International Conference on Robotics and Automation (ICRA)*, 2002.
35. Dempster, A. P., Laird, N. M., and Rubin, D. B., "Maximum Likelihood from Incomplete Data via the EM Algorithm," *Journal of the Royal Statistical Society. Series B (Methodological)*, 1977, pp. 1–38.
36. Leung, K., Inostroza, F., and Adams, M., "Evaluating Set Measurement Likelihoods in Random-Finite-Set SLAM," *IEEE 17th International Conference Information Fusion*, 2014.

APPENDIX-SIMULATION RESULTS WITH OTHER TRAINING DATASET



(a) *RB-PHD-SLAM with constant probability of detection.*

(b) *MH-FastSLAM with constant probability of detection.*

(c) *RB-PHD-SLAM with learned probability of detection and descriptor information.*

(d) *MH-FastSLAM with learned probability of detection and descriptor information.*

Fig. A1–SLAM results in a simulated environment. [Color figure can be viewed at wileyonlinelibrary.com and www.ion.org]