Simultaneous Localization and Mapping with People

Philipp Robbel
MIT Media Laboratory
Cambridge, Massachusetts 02139
Email: robbel@mit.edu

David Demirdjian
Vecna Technologies, Inc.
Cambridge, Massachusetts, 02140
Email: demirdji@csail.mit.edu

Cynthia Breazeal
MIT Media Laboratory
Cambridge, Massachusetts 02139
Email: cynthiab@media.mit.edu

***Abstract*—This paper describes a novel approach for focusing the search for victims of a rescue robot in an unknown environment. Such a rescue operation under time constraints is driven by three opposing factors, namely the need to explore unknown territory, the requirement to maintain robot localization performance as well as map quality in the face of uncertainty, and the need to target the search to areas where victims are likely to be found.**

Key to our approach is to leverage knowledge from victims on the ground to focus and direct the search process. We assume that victims possess useful knowledge about where potentially other victims reside and integrate these directions with a principled exploration mechanism for particle filter-based simultaneous localization and mapping (SLAM). This paper includes a set of proposed evaluations in simulation and on a real robot that aim to validate whether gestural directions can optimize the search for victims when operating under time constraints.

We see this work as a first step towards enabling future applications in search and rescue where humans and robots collaborate in the same physical space to search for victims.

I. Introduction

In this paper we address the problem of *goal-directed* mapping of an unknown environment. Specifically, our focus is on rescue applications where a single ground robot is deployed in an unknown space and tasked to find as many victims as possible within a fixed time interval. The result of this operation should be a high-quality map of a subset of the environment with marked victim locations that can then be supplied to first responders entering the perimeter.

There are a number of challenges implicit in this task description that a successful algorithm needs to address. At the core lies a *simultaneous localization and mapping* (SLAM) problem: a quality map presupposes a good localization estimate but the latter in turn relies on the quality of the map hypothesis. This problem is one of the most studied in the robotics community and a number of landmark-based and occupancy grid-based SLAM algorithms have been demonstrated in recent years to address this interplay (see e.g. [9, 8, 10]). SLAM algorithms generally function as a passive recipient of sensor data and have no direct control over the trajectories that the robot is traversing in the environment. Research in *active exploration* for single and multiple robots has shown that the exploratory behavior affects the localization performance largely and seek to optimize it according to different metrics (e.g. [1], [13], [14], [15]). A well-known fact in this context is that loop-closing, i.e. revisiting already visited poses, can lead to a major reduction in the localization uncertainty, for example.

In the search and rescue community, autonomous victim finding is often treated as a by-product of good exploration of the entire unknown area and any implicit time constraints ignored (e.g. [17]). Given that our scenario calls for purposeful and time-constrained exploration, we seek to minimize the time the robot spends exploring areas where victims are unlikely to be found. At the same time we acknowledge that some time commitment, such as due to loop-closing, may still be required to sustain localization and map quality. An optimal result for our purposes would be a high-quality map of a subset of the entire unknown space that links all victims in the area and includes little needlessly explored space.

The key contribution of this paper is a method to utilize knowledge extracted from victims on the ground in a principled and directed exploration mechanism to focus the search for victims in the rescue domain. The main assumption is that in a rescue scenario people are likely to possess valuable knowledge of where potentially other victims are located. A human that is found in an office could, for example, direct the robot to another office down the hall so that a costly detour of the robot can be avoided. This work can be summarized as a SLAM problem with *human-in-the-loop* exploration and joins elements of classical robot mapping and planning with elements from human-robot interaction (HRI). In essence, our proposed solution embeds people detection and gesture recognition in the information-guided exploration process within a SLAM setting implemented as a Rao-Blackwellized particle filter (RBPF, see [8]). While human communication in the presented work is limited to gesture interpretation, the work extends to more sophisticated modes of interaction, such as language-based direction giving [19, 7].

Our proposed experiments in simulation and on a physical robot aim to quantify to what extent and under which conditions exploiting knowledge from victims on the ground can lead to speed-ups of the search process and increase the number of victims found in the allotted time.

The paper commences with a presentation of the adopted occupancy grid SLAM and information-guided exploration methods in Section II. We then present our approach of human-in-the-loop exploration in Section III and give a detailed outline of the algorithm and the implementation. In Sections IV we conclude with a proposed evaluation of our approach in simulation and on a physical robot.

II. Active Exploration in SLAM

We now briefly introduce the SLAM method that we utilize in our implementation and then turn to one active exploration scheme that we extend in the following section of the paper.

A. Occupancy Grid SLAM

The RBPF ([11, 8]) is an implementation of the full SLAM problem that estimates $p(x_{1:t}, m|z_{1:t}, u_{1:t})$, the joint distribution over the entire robot path and the map, given sensory observations (such as laser scans) up to time t, $z_{1:t}$, and control signals (such as odometry readings) $u_{1:t}$. Rao-Blackwellization uses the following factorization

$$p(x_{1:t}, m|z_{1:t}, u_{1:t}) = p(m|x_{1:t}, z_{1:t})p(x_{1:t}|z_{1:t}, u_{1:t})$$

which decouples the trajectory estimation problem (second factor) from the map computation (first factor).

Occupancy grid maps represent the map as an evenly spaced field of binary random variables corresponding to the occupancy at that location. Estimating the posterior over grid maps given the assumed known path $x_{1:t}$ corresponds to a *mapping with known poses* problem and can be solved by assuming independence between map cells m_i and by utilizing an inverse laser sensor model $p(m_i|z_t, x_t)$ to update the occupancy at each cell with incoming sensor readings ([16]). Rather than recording raw sensor readings, this probabilistic representation accounts for the inherent sensor noise and can filter out moving obstacles such as people.

For the path estimation problem in the second factor above, the RBPF uses a non-parametric implementation of the Bayes filter equations based on a set of M particles with associated weights, each representing a path hypothesis for the robot up to time t. In our planar setting, the robot path consists of a set of poses $x_t = (x, y, \theta)$ denoting 2D location and heading. The particle filter equations define a recursive relationship between the particle set for $p(x_{1:t-1}|z_{1:t-1}, u_{1:t-1})$ and $p(x_{1:t}|z_{1:t}, u_{1:t})$ consisting of a sampling from a proposal distribution (prediction step in the Bayes filter), particle weight updates (correction step), and an adaptive importance sampling step.

In summary, the RBPF maintains a set of path hypotheses each associated with a stochastic map representation stemming from that path. They represent a practical solution to the SLAM problem and have been applied to occupancy grid maps and landmark-based representations alike in recent years [8].

Left out in the description above is the exact procedure of how robot movement should be planned through an unknown environment. This is the topic of active exploration and existing approaches aim to reduce uncertainty in the robot pose [13], map [20], or both [15]. The method we adopt here is the combined measure of [1] which we now briefly introduce.

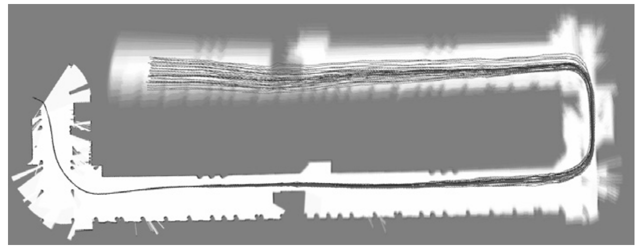

Fig. 1. The *expected map (EM)* in a RBPF with path hypotheses from each particle overlaid. Image is due to [1].

B. Active Exploration

Active exploration methods select robot actions that increase knowledge about the world as defined by some cost function that commonly trades off a measured benefit in knowledge gain with the incurred cost to achieve it [16]. A popular and principled approach is based on the entropy of the posterior path or map distributions and chooses actions to minimize this value [15].

For the work in this paper we use the *information* of the *expected map* (EMI) as an uncertainty measure of the RBPF which considers both robot path and map while avoiding some pitfalls that can occur when computing entropy of an occupancy grid map directly [1]. The *expected map* $p(EM|z_{1:t}, u_{1:t})$ of a RBPF is defined as the result of integrating out the path from the map posterior:

$$p(EM|z_{1:t}, u_{1:t}) = \int p(m|x_{1:t}, z_{1:t})p(x_{1:t}|z_{1:t}, u_{1:t})\, \mathrm{d}x_{1:t}$$

In the case of occupancy grid maps under the usual independence assumptions this corresponds to a weighted sum of the occupancy likelihoods at each grid cell across all particles (see Figure 1). This serves as a both intuitive and principled measure of *mutual consistency* between the individual map hypotheses in a RBPF.

The *information* of an occupancy grid map, I, is defined as the sum over all $I(m_i) := 1 - H(m_i)$ computed for each grid cell i. Here, $H(m_i)$ is the entropy (measured in bits) of the binary random variable denoting occupancy at cell i. Intuitively, I measures the information content of the map with certain occupied or empty cells contributing maximally to the sum while unexplored areas (with a prior probability of occupancy of 0.5) do not contribute at all. Applying the information measure I to the expected map of a RBPF above results in an uncertainty measure (EMI) for the entire RBPF with demonstrated advantages over other measures such as the effective sample size N_{eff} [1].

With the definition above there are two ways of gaining information in the EM of a RBPF, namely to reduce the map "blurriness" by closing a loop or by extending the mapped area into previously unexplored space. The EMI assigns a value to both of those options but favors loop-closing actions

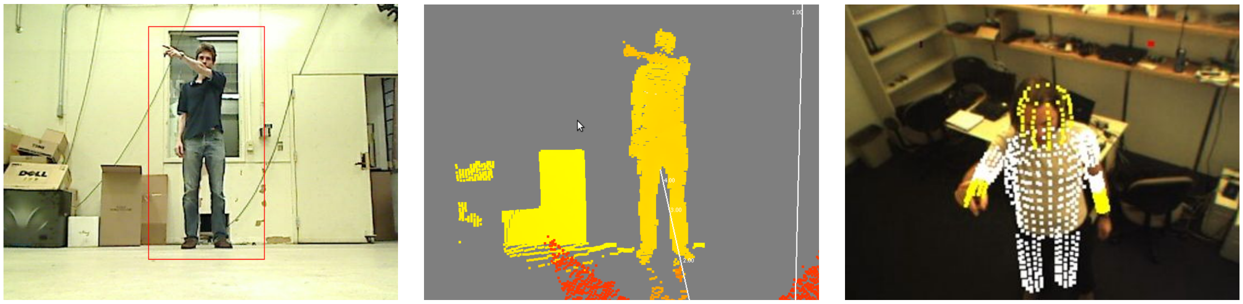

Fig. 2. People detection and gesture recognition from ground robot. From left to right: person recognition, depth image extraction and 3D model fitting. The direction vector is mapped into the robot's coordinate frame as detailed in Section III.

when the pose estimate is uncertain.

Embedded into an exploration framework this method allows to assess different candidate actions of the robot. Previous work generates a set of random sample locations around the robot and chooses the one that maximizes the expected gain in EMI as the next target for exploration. Our contribution instead gives humans an active role in the exploration process, allowing gestural directions to steer the robot towards salient locations in the unexplored space. Our method is detailed in the following section.

III. Human-in-the-loop Exploration

In the scenario we are considering, a rescue robot's core task is to find as many victims as possible in the allotted time and to report back with a quality map that indicates a path to the recorded victims. Underlying the approach presented in this paper is the assumption that knowledge from humans in the same physical space can be leveraged to focus the search for victims in the unknown environment. The question we seek to address in a principled way is how and when the robot should utilize human input and when other factors, such as maintaining good localization, should take precedence during search.

The usefulness of victim interaction during search is directly dependent on two factors, namely the capability of the robot to extract useful information from humans as well as the appropriate usage thereof. The first is best described as a human-robot interaction (HRI) problem [2]. In our current implementation detailed below we limit the interaction to interpretation of gestural directions from the victims. We note, however, that a vast amount of literature on intention recognition (e.g. [7]) or direction understanding (e.g. [19]) applies in principle.

To address the second factor, we embed human directions in a principled exploration mechanism that utilizes the EMI measure introduced in the previous section. Candidate exploratory actions are sampled along human direction vectors but traded off against "safe" loop closing actions according to their EMI value. Rather than following directions blindly, the robot maintains enough autonomy to guard itself against wandering off into space where localization performance may not be maintainable.

We now outline our approach for recognizing gestural directions before describing our exploration algorithm in detail.

A. Gestural direction understanding

Our recognition pipeline for gestural directions consists of two parts. First a pointing gesture is extracted from a human victim which is then translated into a direction vector in the robot's coordinate frame. Second, a probabilistic interpretation of direction is overlaid on the currently built EM of the RBPF to define a search space derived from the observed gesture.

1) Gesture recognition: Gesture recognition is part of an interaction that is initiated whenever a human is detected in the ground robot's field of view. For human detections with a monocular robot-mounted camera we use the histogram of oriented gradients (HOG) descriptor and a linear SVM classifier [3]. We currently employ the real-time GPU implementation and trained SVM classifier of [12] successfully with standing and sitting humans.

Humans are assumed stationary for the interaction that is now initiated. Through an on-board speaker the robot asks the human to point in the direction of more victims if known. At the same time, an on-board 3D time-of-flight camera records depth information in the field of view of the robot. Based on optical flow in the camera image we forward point clouds corresponding to "stationary" images to a 3D model fitting algorithm to overlay a 3D model on the depth data. The optical flow criterion reduces the number of poses to evaluate but assumes that a directional gesture is held for a short amount of time.

We define a 3D articulated model (see Figure 2) by a set of limbs connected to each other by rotational joints. We parameterize pose as $\theta = (C, \phi_1, ...\phi_M)$, where C is the 3D torso location and ϕ_1, ..., ϕ_M is the set of articulated angles. The dimension of θ is 20.

Similar to [4, 5], we estimate articulated pose by minimizing a fitting error function $E(\theta)$ based on the distance between a 3D human model and the recorded 3D points. Minimization is performed using a variant of the Iterative Closest Point (ICP) algorithm to register articulated objects.

To achieve robust estimation, we employ a multi-hypotheses framework similar to [6] and reduce the search space through

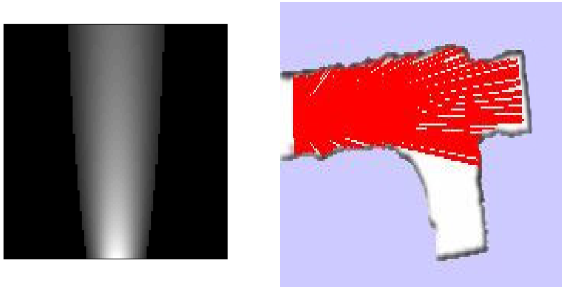

Fig. 3. Human direction likelihood field modeling the distribution over victim locations given a vertical direction vector (left). Likelihood field overlaid on current map (with distance cutoff) (right).

constraints on the poses θ reachable by the human body. Model fitting stops when $E(\theta)$ is below a threshold or a fixed number of ICP iterations has been reached. To speed up the model matching process at each ICP iteration, we have implemented different algorithms including kd-trees and exhaustive search. The overhead due to the construction of kd-trees makes them too slow in practice and we achieve highest performance with a GPU-implementation of exhaustive search.

2) Direction interpretation: The model fitting process yields a direction vector that can be mapped directly into the robot's reference frame based on the current position and heading estimate. To account for inaccuracies in human directions (especially in an emergency scenario) we seek a probabilistic interpretation of gesture if a directional vector was obtained in the previous step. Specifically, we are interested in $p(victim|x_t, \vec{d}, m_t)$, the distribution over victim locations given x_t, the position of the direction giver, $\vec{d}$, the indicated direction, and m_t, the map built so far during SLAM.

Our model is inspired by other sensor forward models (such as for laser or sonar scanners) from the robot localization and mapping literature, specifically the likelihood field described in [16]. We define the density as

$$p(victim|x_t, \vec{d}, m_t) = \epsilon_{\sigma(dist)}$$

where ϵ denotes a Gaussian centered on the direction vector with growing variance away from the direction giver. Unlike the sensor model for laser and sonar scanners there is no obvious "cut-off" because we do not obtain a distance estimate from the direction giver.

Note that the algorithm we describe in the following section only requires us to sample from this distribution in order to select candidate locations to explore. Figure 3 shows the direction model both individually and superimposed on a partially explored map. In the left image, the lighter a location appears, the larger $p(victim|x_t, \vec{d}, m_t)$. In the exploration algorithm we introduce next, our probabilistic interpretation of direction is overlaid on the currently built EM of the RBPF to define a search region from the observed gesture.

B. Integrated exploration algorithm

In this section we describe how gestural directions come together with the previously outlined information measure to form an integrated framework for human-directed exploration in a RBPF. In our approach we follow the basic framework of [15]:

1. Generate a set of sample target locations around the robot using the most likely map of the RBPF.
2. Predict the observations along the path to each target and integrate them into a copy of the original RBPF.
3. Determine the information gain between each candidate and the original RBPF.
4. Choose the target location which results in the highest expected gain in information.

The final target that the robot navigates to trades off path length with expected information gain. In our scenario, the process above iterates until a fixed time has elapsed.

Our version of the exploration scheme is outlined in Algorithm 1 below. The key difference is the role that human directions play in the search. In line 6, a set of candidate targets is sampled from $p(victim|x_t, \vec{d}, m_t)$ projected on the current EM of the RBPF. We draw from this posterior by first sampling a distance $dist$ along the direction vector and then a displacement from the Gaussian $\epsilon_{\sigma(dist)}$ ensuring that it falls into the boundaries of the EM.

Line 11 computes the Utility of a candidate target. Utility is defined as expected information gain minus path cost. The latter is proportional to the path length but weighted by it's likelihood under $p(victim|x_t, \vec{d}, m_t)$. The likelihood field is decayed towards a uniform distribution in line 14 every time

Algorithm 1: Human-in-the-loop exploration

Input: $RBPF_t$ the current RBPF at time t
1 initialize likelihood field to uniform random across map;
2 **while** *search time has not elapsed* **do**
3   update $RBPF_t$ with current sensor readings;
4   **if** *robot is not navigating* **then**
5   get most likely particle and grid map;
6   sample k potential targets from likelihood field;
7   **foreach** *target i* **do**
8       $RBPF(i) \leftarrow RBPF_t$;
9   update $RBPF(i)$ with simulated observations along path to target;
10   compute $EMI(i)$, the EMI of $RBPF(i)$;
11   compute $Utility(i)$;
12   **end**
13   begin navigating to target with highest utility;
14   decay likelihood field to uniform random;
15   **else**
16   proceed to current target;
17   **end**
18   **if** *human is detected in field of view* **then**
19   stop navigating;
20   update likelihood field from direction vector;
21   **end**
22 **end**

a new target location is drawn to avoid the robot following false or obsolete directions for too long.

Note that in our current implementation the robot keeps track of previously recorded victim locations. Querying for directions only takes place when a victim at a particular location is first encountered.

IV. Proposed Evaluation

In this section we outline the proposed experiments for the human-in-the-loop exploration mechanism described above.

The efficiency of the search for victims with our algorithm is largely affected by two factors, namely *i)* the accuracy of the human gestures, and *ii)* the accuracy of the direction extraction, which is in turn based on the human detection and gesture recognition performance. In the following, we suggest to evaluate our approach in a simulation setup as well as on a physical robot. The first allows us to abstract away the complexities associated with human detection and direction extraction and to assess the performance of the algorithm under varying accuracy of the human direction vectors. The second evaluation concerns the integrated algorithm and compares the performance with other exploration schemes in a more realistic setting.

Of particular interest to us is what the minimum accuracy of human directions must be in order for the algorithm to perform better than simpler exploration mechanisms that do not rely on human input.

A. Experimental setup

The implementation of our algorithm extends an existing and openly available implementation of the RBPF SLAM algorithm, *GMapping*[1]. Enabling GMapping for active exploration required us to add computation of the EM and EMI (for target destination selection), a laser raycasting module (to simulate experience along the path to each candidate destination), and a navigation module (to drive to the selected destination points). In our implementation, navigation is handled by the freely available Carmen toolkit[2].

For the evaluation on the physical robot we also integrated the HOG-based person detector of [3] and the gesture recognition method of [6] into our robot control software.

In the following two sections, we contrast our algorithm against two other popular exploration methods that do not include a human in the loop, namely a frontier-based [18] and the original EMI-based [1] exploration methods.

B. Simulation setup

The first evaluation will be carried out in the USARSim simulation environment. The key assumption to our work is that victims possess useful knowledge about where potentially other victims reside so that the exploration process can be targeted accordingly. In this experiment, we evaluate our algorithm's reliance on the accuracy of human directions.

[1]See `http://www.openslam.org/gmapping.html`.
[2]See `http://carmen.sourceforge.net`.

Fig. 4. Used map for simulation: 2006 RoboCup Rescue office environment.

Our procedure is as follows: we introduce a single parameter k that summarizes the correctness of victim directions as well as the robot's ability to extract the direction vector. Whenever the robot reaches a victim (whose locations in the simulation are known exactly), the likelihood that the correct direction vector is made available to the robot is k whereas a random direction vector is returned with a probability of $1-k$. By varying k, we will find the minimum required accuracy for our algorithm to outperform frontier exploration and the default EMI-based method. Performance is assessed based on the number of victims found in a fixed time interval.

The map we use during this part of the evaluation is the office environment from the 2006 RoboCup Rescue competition and is shown in Figure 4. Repeated measurements will be taken and victim positions varied from one run to the next.

C. Application to real-world data

The evaluation of the algorithm on physical hardware is carried out on a mobile robot in our lab (see Figure 5). The robot's task is to find five human victims (either standing or sitting on the ground) in a planar in-door environment as quickly as possible.

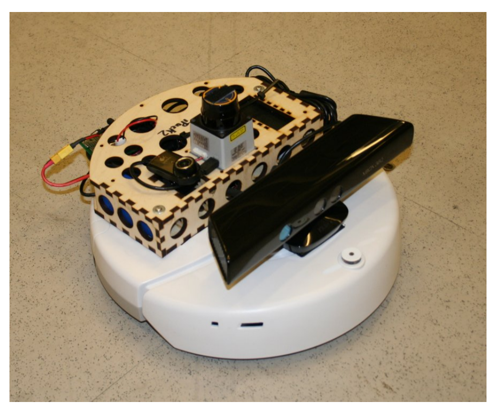

Fig. 5. The "iPuck" robot with laser scanner, Microsoft Kinect depth camera, and on-board Intel Atom PC.

Different from the evaluation in simulation above, performance of the algorithm is now also dependent on the human detection and gesture recognition performance. For this part of the evaluation we will first collect a dataset of representative human configurations (standing, sitting, and pointing in different directions) and evaluate the performance of each classifier in isolation. Second, the integrated algorithm will be executed and compared to frontier- and EMI-based exploration according to the "number of victims found in a fixed interval" criterion.

Acknowledgments

P. Robbel and C. Breazeal were supported by the Office of Naval Research. Their support is gratefully acknowledged.

References

[1] J. L. Blanco, J. A. Fernández-Madrigal, and J. Gonzalez. A novel measure of uncertainty for mobile robot slam with rao-blackwellized particle filters. *Int. Journal of Robotics Research*, 27(1):73–89, 2008.

[2] Cynthia Breazeal. Toward sociable robots. *Robotics and Autonomous Systems*, 42(3-4):167–175, 2003.

[3] Navneet Dalal and Bill Triggs. Histograms of oriented gradients for human detection. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 886–893, 2005.

[4] Q. Delamarre and O. D. Faugeras. 3D articulated models and multi-view tracking with silhouettes. In *Proceedings of ICCV'99*, pages 716–721, 1999.

[5] D. Demirdjian and T. Darrell. 3D articulated pose tracking for untethered deictic reference. In *Proc. ICMI'02, Pittsburgh, PA, USA*, 2002.

[6] D. Demirdjian, L. Taycher, G. Shakhnarovich, K. Grauman, and T. Darrell. Avoiding the streetlight effect: Tracking by exploring likelihood modes. In *IEEE International Conference on Computer Vision*, pages 357–364, 2005.

[7] F. Doshi and N. Roy. Efficient model learning for dialog management. In *Proc. ACM/IEEE International Conference on Human-robot Interaction (HRI)*, pages 65–72, New York, NY, USA, 2007. ACM.

[8] G. Grisetti, C. Stachniss, and W. Burgard. Improved Techniques for Grid mapping with Rao-Blackwellized Particle Filters. *IEEE Transactions on Robotics*, 23(1): 34–46, 2007.

[9] M. Kaess, A. Ranganathan, and F. Dellaert. iSAM: Incremental smoothing and mapping. *IEEE Trans. on Robotics, TRO*, 24(6):1365–1378, Dec 2008.

[10] M. Montemerlo, S. Thrun, D. Koller, and B. Wegbreit. FastSLAM 2.0: An improved particle filtering algorithm for simultaneous localization and mapping that provably converges. In *Proceedings of the Sixteenth International Joint Conference on Artificial Intelligence (IJCAI)*, Acapulco, Mexico, 2003. IJCAI.

[11] K. P. Murphy. Bayesian map learning in dynamic environments. In *Proc. Advances in Neural Information Processing Systems (NIPS)*, pages 1015–1021, Denver, CO, USA, 1999.

[12] Victor Prisacariu and Ian Reid. fasthog - a real-time gpu implementation of hog. Technical Report 2310/09, Department of Engineering Science, Oxford University, 2009.

[13] N. Roy and S. Thrun. Coastal navigation with mobile robots. In *Proc. Advances in Neural Processing Systems (NIPS)*, pages 1043–1049, 1999.

[14] C. Stachniss. *Robotic Mapping and Exploration*, volume 55 of *STAR Springer tracts in advanced robotics*. Springer, 2009. ISBN 978-3-642-01096-5.

[15] C. Stachniss, G. Grisetti, and W. Burgard. Information gain-based exploration using rao-blackwellized particle filters. In *RSS*, pages 65–72, Cambridge, MA, USA, 2005.

[16] S. Thrun, W. Burgard, and D. Fox. *Probabilistic Robotics*. The MIT Press, September 2005. ISBN 0262201623.

[17] A. Visser and B. Slamet. Balancing the information gain against the movement cost for multi-robot frontier exploration. In *Proc. Second European Robotics Symposium (EUROS)*, pages 43–52, Prague, Czech Republic, 2008.

[18] Arnoud Visser, Xingrui-Ji, Merlijn Ittersum, Luis A. González Jaime, and Laurenţiu A. Stancu. Beyond frontier exploration. *RoboCup 2007: Robot Soccer World Cup XI*, pages 113–123, 2008.

[19] Y. Wei, E. Brunskill, T. Kollar, and N. Roy. Where to go: Interpreting natural directions using global inference. In *Proc. IEEE International Conference on Robotics and Automation (ICRA)*, pages 3761–3767, Kobe, Japan, 2009.

[20] Brian Yamauchi. A frontier-based approach for autonomous exploration. In *In Proceedings of the IEEE International Symposium on Computational Intelligence, Robotics and Automation*, pages 146–151, 1997.