Object Localization using Bearing Only Visual Detection

Kristoffer SJÖ [a,1], Chandana PAUL [a] and Patric JENSFELT [a]
[a] *Royal Institute of Technology (KTH), Centre for Autonomous Systems, SE-100 44 Stockholm, Sweden*

Abstract.
This work demonstrates how an autonomous robotic platform can use intrinsically noisy, coarse-scale visual methods lacking range information to produce good estimates of the location of objects, by using a map-space representation for weighting together multiple observations from different vantage points. As the robot moves through the environment it acquires visual images which are processed by means of a fast but noisy visual detection algorithm that gives bearing only information. The results from the detection are projected from image space into map space, where data from multiple viewpoints can intrinsically combine to yield an increasingly accurate picture of the location of objects. This method has been implemented and shown to work for object localization on a real robot. It has also been tested extensively in simulation, with systematically varied false positive and false negative detection rates. The results demonstrate that this is a viable method for object localization, even under a wide range of sensor uncertainties.

Keywords. Accumulator Grid, Object Detection, Object Localization

Introduction

One of the major thrusts of robotics has been the development of robots which can provide domestic assistance in household tasks. Although this ambitious dream has now come closer to reality, there are still many challenging areas which remain to be addressed. One of the main areas is the development of appropriate representations of the environment which enable the robot to be cognizant of its surroundings and interact in a way that is appropriate to human-centered requirements.

One of the ways in which robots have been made more aware of their surroundings is through the development of spatial maps for navigation. There has been much work in this area, and the field of SLAM, Simultaneous Localization and Mapping has matured significantly. There are now an abundance of approaches to address the problem. For an overview see for example [1].

However, pure navigation maps such as those developed with SLAM, containing no extra structure of the environment, are not sufficient for the robot to deal with complex tasks requiring interaction with objects. For this, more complex representations need to be developed which contain both spatial and semantic information.

[1]Corresponding Author: Kristoffer Sjö, Royal Institute of Technology (KTH), Centre for Autonomous Systems, SE-100 44 Stockholm, Sweden; E-mail: krsj@nada.kth.se

There has been relatively little work which deals with the development of such semantic maps. Kuipers was one of the true pioneers in this area in robotics, with the so called Spatial Semantic Hierarchy [8,9]. Nonetheless, despite the early interest in this topic, the area is just beginning to gain momentum. Galindo *et. al.* have recently developed a hierarchical spatial representation in which simple objects have been detected and added [6]. Vasudevan *et. al.* have developed a representation which combines a topological representation of space with so called object graphs which model the spatial relation between objects within an area [16]. The recognition of objects is based on SIFT features [12]. Ranganathan *et al* have used three different detectors to detect objects: Harris corners, Maximally Stable Extremal Regions (MSER) and Canny edges and using a SIFT descriptor, inserted the features into a constellation graph model to generate a representation of a topological location [14].

Most of this work has dealt with the issue of an ideal structure for the representation of the environment, detached from the requirements of creating such a representation. We bring to light, however, that there is relationship between the accuracy of a representation and the amount of resources used to create that representation. Thus, if the robot performs a detailed inspection of the environment, visually processing every part of the environment from a close distance, it can build an accurate representation. However, if the robot performs a quick perfunctory inspection of the environment, with a fast visual processing algorithm, it can acquire data to build an approximate representation, consisting of hypotheses for potential object locations.

We further bring to light that both these methods can be used in conjunction, leading to a representational structure with two conceptual levels. The first level is an approximate representation of the location of objects in the environment. This representation, which takes very little resources to build, can be used to guide the construction of the second level, which is the accurate spatial and semantic representation of the environment. This multi-tiered approach leads to a much more directed and efficient use of sensory and computational resources in the creation of a semantic representation.

In previous work [13] we developed an accurate semantic representation which consists of four layers. At the first layer, it includes a metric representation created using SLAM methods, at the second a navigation graph of nodes indicating navigable free space and their connectivity, at the third a topological map dividing space into rooms and corridors, and at the fourth a conceptual map of semantic entities corresponding to places and objects and their relationships. In [11] we presented a method for visually recognizing and localizing these objects and inserting them into the map. Such visual procedures are expensive, and require much processing. In this paper, we have focused on the development of a layer of approximate representation regarding the location of objects using fast processing, which can then allow the robot to direct its more expensive visual and computational resources.

Our method of creating an approximate representation from uncertain visual processing techniques has been inspired by the creation of occupancy grids [4]. In an occupancy grid the world is divided into a grid of cells and each cell is assigned a value to reflect the certainty of that cell being occupied. A large number of methods for updating such grids have been proposed, based on Bayesian theory, fuzzy logic, etc. See [15,17] for an overview and comparison of sonar based occupancy grid methods. The sonar sensor provides relatively accurate distance estimates but the angular information is rather vague. Integrating many measurements is therefore necessary to get a good estimate of

the structure of the environment. For obstacle avoidance using sonar the Histogram in Motion Mapping approach [2] is sometimes used as it provides a fast way of updating the grid in a time-critical application. In it a simplified update of the occupancy grid is used where only the cells along the acoustic axis of the sensor are updated, by adding and subtracting fixed integer values, thus ignoring the angular uncertainty of the sensor. This is in contrast to the full Bayesian formulation, where every cell will be affected by every observation. Accuracy in the model is instead gained by frequent updates.

However, this work is different from the previous work on occupancy grids, in that fast visual detection is much more uncertain than previous sensing techniques. With a laser or sonar sensor, there is a fairly high likelihood that if a positive reading arises then an obstacle exists within the range of the sensor. However, with a fast visual detection algorithm, many false positives can arise, and thus locating an object with a certain measure of confidence presents a much greater challenge. Furthermore the sensors used for occupancy grids usually provide good range information and moderately precise bearing information. The fast visual detection algorithm used here on the other hand provides no range information, only bearing, which makes it a slightly different problem. Finally, previous techniques have been object non-specific. Thus, every entity in the environment has been considered to be of the same type. In this work, a higher degree of semantic specificity is applied, and only objects recognized by a trained visual algorithm are considered. Thus the entities considered are much sparser in the environment, and the localization problem is different, as detections from the same location are likely to arise from the same object, as opposed to two closely adjacent objects.

Methods akin to the occupancy grid have also been used for global localization of a mobile robot [5], where the grid represents a discretization of the probability density function for the pose of the robot. In this work, however, the robot's position is considered known and objects are what is being localized.

This work is also similar in purpose to the mapping aspect of bearing-only SLAM; however, such methods [3,10] are not designed to cope with the high levels of false positives and false negatives that are present in the visual detection algorithms; rather, they deal with uncertain bearing measurements in addition to missing range data.

In the next section, the accumulator grid algorithm is presented. Following this, in Section 2 a proof-of-concept implementation on a mobile robot is described, and results are demonstrated. Section 3 introduces a simulation environment in which a thorough performance evaluation of the algorithm under varying conditions of false positive and false negative detection rates is performed. The results are presented, followed by discussion and conclusions.

1. Accumulator Grid

In order to create and maintain a distribution across space of the confidence the robot has in the presence of objects, an *accumulator grid* has been developed. This is a grid overlaid on the robot's navigation map, where each grid cell is associated with a confidence value representing the certainty of an object being in the cell. One set of such values is maintained for every distinct object of interest. In the following description, only one object is considered for simplicity.

As the robot moves through the environment and acquires visual data, the accumulator grid cells are updated according to the output of the visual object detection algo-

rithm. As data accumulate from different locations, the confidence values reinforce each other in a way similar to the functioning of the Hough transform [7], embodying an increasingly accurate representation of the true locations of objects in the environment. The robot can then utilize these to direct its motion, or store them for later use. Although the information in the grid is 2-dimensional, it can still be used to guide a directed 3D object search, considering all vertical locations corresponding to the objects 2D location in the map.

1.1. Initialization

The accumulator grid is set up with a specific extent in the X and Y dimensions as well as a cell size. Appropriate values for these parameters are highly dependent on the environment the robot operates in. In general, its extent should equal the size of the operating area and the cell size should be roughly equal to the size of sought objects. If objects of different sizes are to be represented, the smallest size can be used or, alternatively, several grid sizes may be used in parallel.

Initially, the confidence values are all set to 0 if the robot does not possess any prior information about object locations. It would be possible to represent prior knowledge of the likely and unlikely locations of objects during initialization using non-zero values.

1.2. Image processing

The algorithm is designed to work with Receptive Field Cooccurrence Histograms (RFCH). RFCH is an image processing method for object detection based on bulk image properties, as opposed to feature-based methods such as SIFT typically used for recognition. It is used here because it is fast and produces a scalar degree-of-match as output given any part of an image, large or small. RFCH are object-specific and so this algorithm, based on them, will be as well. However, any object detection algorithm with similar properties to RFCH matching could in principle be substituted.

Prior to engaging in the object search, the robot's vision system is trained on each object of interest by performing feature value clustering and histogram extraction on training images of the objects of interest. The clusters and the histogram for the object are stored for the purpose of object detection on the images subsequently acquired.

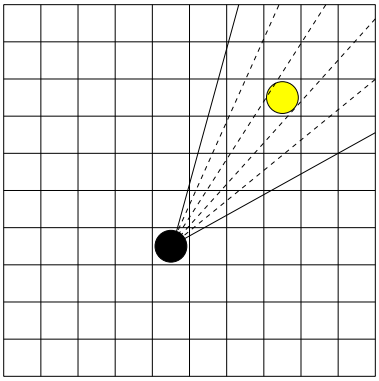
1.3. Update

As the robot proceeds through its surroundings, it acquires camera images, which are subdivided into small patches for which RFCH are extracted. The resulting histograms are compared to the histogram from the training image of the object being sought, producing a match value for each patch.

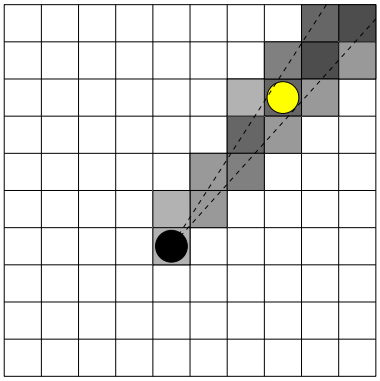
It is assumed that the pose of the robot is known at all times. If this is not the case, it will become necessary to take steps similar to those needed for occupancy grids built during mapping under uncertain odometry; such considerations are however beyond the scope of this paper.

The robot's camera is kept horizontal during exploration. As a result, each column of image patches corresponds to a specific interval of bearings, which is projected onto the 2D map from the location of the robot. This produces a "sensor wedge" as shown in Figure 1(a).

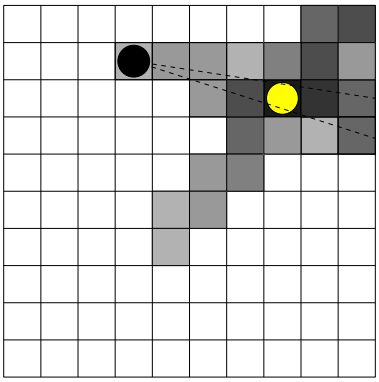
The accumulator grid is then updated according to the following principle: Each cell covered by a sensor wedge is incremented by the maximum magnitude of all the RFCH match values associated with it. This value represents the likelihood of the object's presence within that wedge; see Figure 1(b). Note that we do not make any assumptions about range information from the visual detection algorithm, but only bearing information. As the robot continues to acquire images from different vantage points, the wedges intersecting objects will reinforce the grid cells, whereas cells that are only incidentally part of a sensor wedge will not get reinforced; see Figure 1(c).



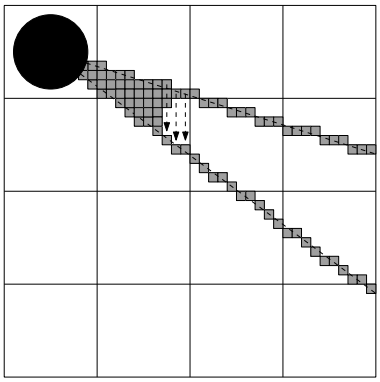
(a) The robot's field of view is subdivided into wedges



(b) Each wedge is updated by its associated sensor response. Note the smoothing effect of supersampling



(c) As views are acquired from different vantage points, cells containing objects will be reinforced



(d) Cells in each wedge are incremented, using supersampling to counter aliasing effects

Figure 1. Principles of the accumulator grid update

In practical terms, the update of the cells in each sensor wedge is carried out by means of line rasterization, performed in parallel for the left- and right-hand rays of the slice, respectively, and by filling in the intermediate cells in rows or columns as appropriate. However, given the potentially very large granularity of the accumulator grid and the aliasing effects this causes, a supersampling scheme is adopted in which the resolution of the grid is augmented by a factor of 10 during the update. In effect, each

cell is updated in proportion to how well it is covered; see Figure 1(d). This alleviates aliasing problems without incurring any extra storage costs.

2. Implementation and Experiments

The algorithm described in the previous section was implemented and tested on a Performance Peoplebot mobile robot platform. The robot is approximately 1.2m in height. It is equipped with a Canon VCC4 pan-tilt-zoom camera, a SICK Laser range finder, and ultrasonic sensors. The camera is mounted at a height of 1m above the floor, and has a horizontal field of view of 45 degrees. The camera was used to acquire images at a resolution of 320×240 pixels. In order for the robot to be able to detect objects in a wide field of view as it explores the environment, the camera's zoom was not used. The laser scanner was mounted at 30 cm above the floor, and used to acquire information about the structure of the environment. Features extracted from laser information were then used in a standard EKF based SLAM algorithm, to obtain a metric map of the environment and the position of the robot within it.

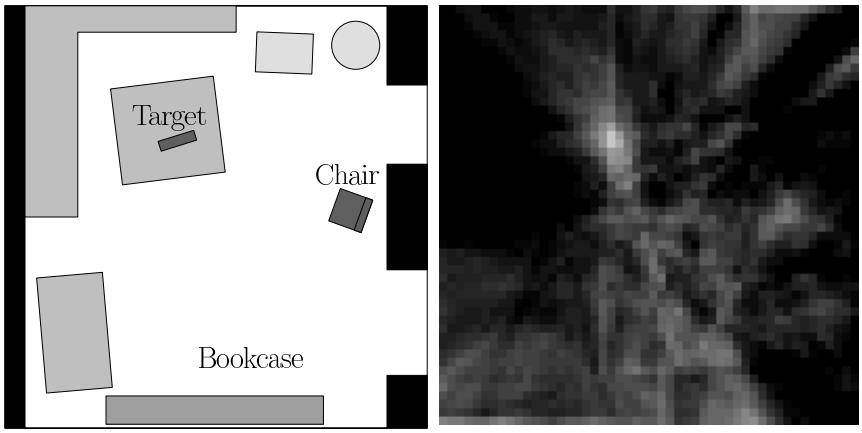
The test of object localization was performed in a mockup living room of approximate dimensions 4.5m $\times$ 6m, shown schematically in Figure 2(a). The living room consisted of a sofa, a coffee table, a chair, a table, and a bookcase. Other miscellaneous items were also present. The target which was a multi-colored rice box was placed on the coffee table towards one corner of the room. An accumulator grid of 50×50 cells was created at a resolution of 0.1m to represent the environment, and all values were initially set to zero. The robot was programmed to visit 5 predefined locations in the room and take pictures in all directions (8 views at a field-of-view of 45°) from each location. RFCH detection was carried out from each viewpoint, and an accumulator grid was updated according to Section 1.3.

After visiting each location and processing the images, the result was the accumulator grid shown in Figure 2(b). The actual location of the sought object, a packet of rice, is plainly visible as the brightest spot in the upper left quarter of the grid. On the center right a chair of somewhat similar color to the object can be made out, and towards the bottom of the grid a bookcase containing many items of varying appearance causes a low-level blur; still, these false positives are clearly less prominent than the true location. A view planning algorithm can use this information to create priorities for regions to investigate, which would lead it to begin with the area that actually contains the object in this case.

3. Simulation

Although the above experiment demonstrates the feasibility of the approach, there are many factors that can affect the reliability of the visual detection on which it is based. These include lighting conditions, the structure of the environment, the saliency of the object and the parameters of the detection algorithm.

In order to perform a broader investigation of how these factors influence the accumulator grid algorithm, the performance of the system was evaluated by means of Monte Carlo simulation in an abstract scenario, in which the false positive and false negative detection rates of the object detection can be varied freely.



(a) Schematic representation of the room used

(b) Result of grid accumulation. The true object location is at the bright spot on the upper left

Figure 2. Experimental results

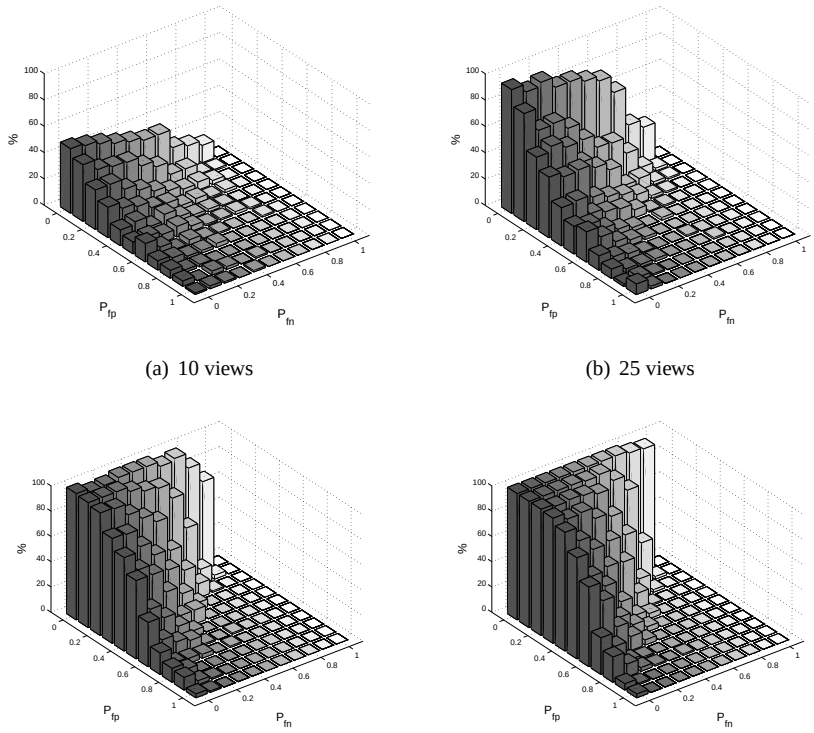
The environment is represented as a 20×20 grid. The object is positioned randomly in the environment (excepting the outer edges), and occupies exactly one grid cell. The robot can move to any part of the environment except for the cell containing the object. The robot can also have any orientation in space. In the real world, the robot would follow some continuous trajectory through space, sensing as it went. It would thus see the object from several distinct orientations and views. Here, in order to simulate this fact, while avoiding any bias introduced by a manually selected trajectory, the robot is given a random new orientation and position for every view that is acquired. The robot is assumed to know its pose with perfect accuracy.

For a given robot pose, a field of view is simulated by an angular slice of the map, with its apex at the robot's position, as was shown in Figure 1(a). The field of view is divided into wedges corresponding to the image patch columns described in Section 1.3. For each wedge, the cells it covers are incremented if the object intersects the wedge.

After a set number of random views, the search is terminated and the result is evaluated by locating the maximally-valued cell and comparing its position in the map to the known position of an object. The test is considered successful if the cell with the maximum value in the accumulator grid is the one containing the object or adjacent to it.

The performance of a noisy visual detection algorithm such as RFCH matching was simulated as a detector which gives a binary response on the location of the object, with non-zero false positive and false negative rates P_{fp} and P_{fn} respectively. The performance of object localization with the accumulator grid, under such noisy visual detection conditions, were evaluated in a series of tests. The false positive and false negative rates were varied in intervals of 0.1 between 0 and 1. For each combination of values, tests were performed evaluating the localization with 10, 25, 75 and 150 views. 100 such tests were performed for each parameter setting.

Figure 3 shows the outcomes of the tests. Obviously, a low view count does not provide enough data for a good estimate, but with 10 random views a good guess can be made approximately 40% of the time if the false positive and false negative rates are low. With 25, results are reasonably reliable in the absence of noise. The real-world test



(a) 10 views (b) 25 views

(c) 75 views (d) 150 views

Figure 3. Performance of object localization for various probabilities of false positive and false negative rates. The Z axis denotes the percentage correct estimations, and the X and Y axes the rate of false positives and negatives, respectively.

in Section 2 with 40 well-planned views compares to a situation with 75 or 150 random views. At this level the algorithm is able to deal with a large number of false negatives if the false positive rate is low. The converse also holds, though to a somewhat lesser extent. This is because a false negative does not alter the accumulator grid, whereas a false positive introduces flawed data.

Typically most visual processing algorithms are associated with a *Receiver Operating Characteristic* or ROC curve, describing how the false positive rate relates to the false negative rate for that algorithm when varying some discrimination threshold. For RFCH, for instance, a threshold on the degree of match will determine these error rates. Figure 3 shows that good results are achieved whenever either P_{fp} or P_{fn} can be made small, which is the case with many algorithms. Thus, with knowledge of the ROC curve, it is possible to optimize the performance of the accumulator grid for any given algorithm.

4. Discussion

The simulated and real results presented in this paper are promising, and suggest that this algorithm would be a viable method for object localization under various conditions

of sensing uncertainty. However, there are several issues which should be considered further.

Robot localization

In the simulation, it was assumed that the robot knew its pose with perfect accuracy. However, in the real world there can be some discrepancy between the robot's real pose and estimated pose. The accumulator grid is dependent upon a good estimate of the position of the robot. If the estimate drifts over time, the grid may be affected through the blurring, offset, or duplication of maxima. Nevertheless, contemporary SLAM methods are typically sufficiently robust for this not to be a problem.

Viewpoint bias correction

In simulation, the robot was given a new position and orientation at every time step. In the real world the robot doesn't move randomly through space, and subsequent locations and views of the object are not entirely independent of previous locations. This is not always a critical problem, as is seen in the successful localization in Figure 2(b), but nonetheless bears consideration.

Extrinsically, the problem can be solved by planning for the robot to move and obtain visual data in a way that provides a good distribution of viewpoints. If this is not possible, intrinsic solutions involving changes to the algorithm itself might be made. Possibilities include weighting of the grid increment based on the amount of sensor data gathered from the current direction, or the creation of multiple grid layers for different viewing directions. Either way would require increasing the amount of data stored.

RFCH

The use of RFCH in this work was due to its ability to produce degrees of match across the entire image, as well as its relatively good object specificity. However, though relatively fast, current implementations still leave something to be desired in terms of speed and especially scalability: at present processing a 320×240 image requires on the order of one second on our platform, and this grows linearly with the number of objects. Faster implementations and alternative methods, especially hierarchical ones, would be a valuable modification.

Furthermore, RFCH doesn't respond equally well to different views of the same object. An easy solution to this is to represent multiple views as multiple objects, but this is costly. Clustering of similar views can help in this regard, as could a hierarchical object detector structure. A hierarchical detection algorithm could also help reduce the memory requirements of the accumulator grid itself, which currently maintains one layer of cells per distinct object.

5. Conclusion

This paper presents a novel method for consolidating noisy bearing-only visual information to achieve object localization. The method uses an accumulator grid to update

confidence information through an algorithm that transforms the data from visual into map space. Results are presented of feasibility testing in a realistic scenario as well as an extensive evaluation in simulation. The results indicate that the method performs well in the face of noisy measurements and promises to be useful as a way of obtaining and representing the approximate location of objects in a robot's environment.

Acknowledgements

This work was supported by the EU FP6 IST Cognitive Systems Integrated Project "CoSy" FP6-004250-IP. Kristoffer Sjö was supported in part by the Swedish Research Council, contract 621-2006-4520.

References

[1] T. Bailey and H. Durrant-Whyte. Simultaneous localization and mapping (SLAM): Part II state of the art. *IEEE Robotics & Automation Magazine*, 13(3):108–117, 2006.

[2] J. Borenstein and Y. Koren. Histogram in-motion mapping for mobile robot obstacle avoidance. *IEEE Transactions on Robotics and Automation*, 7(4):535–539, August 1991.

[3] A. J. Davison. Real-time simultaneous localisation and mapping with a single camera. *Computer Vision, 2003. Proceedings. Ninth IEEE International Conference on*, pages 1403–1410 vol.2, 13-16 Oct. 2003.

[4] A. Elfes. Using occupancy grids for mobile robot perception and navigation. *Computer*, 22(6):46–57, 1989.

[5] D. Fox, W. Burgard, and S. Thrun. Markov localization for mobile robots in dynamic environments. *Journal of Artificial Intelligence Research*, 11:391–427, 1999.

[6] C. Galindo, A. Saffiotti, S. Coradeschi, P. Buschka, J.A. Fernández-Madrigal, and J. Gonález. Multi-hierarchical semantic maps for mobile robotics. In *Proc. of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS'05)*, pages 3492–3497, 2005.

[7] J. Illingworth. and J. Kittler. A survey of the hough transform. *Computer Vision, Graphics, and Image Processing*, 1988.

[8] B. J. Kuipers. Modeling spatial knowledge. *Cognitive Science*, 2:129–153, 1978.

[9] B. J. Kuipers. The spatial semantic hierarchy. *Artificial Intelligence*, 119:191–233, 2000.

[10] T. Lemaire, S. Lacroix, and J. Sola. A practical 3d bearing-only slam algorithm. *Intelligent Robots and Systems, 2005. (IROS 2005). 2005 IEEE/RSJ International Conference on*, pages 2449–2454, 2-6 Aug. 2005.

[11] D. Galvéz Lopez, K. Sjöö, C. Paul, and P. Jensfelt. Hybrid laser and vision based object search and localization. In *Proc. of the IEEE International Conference on Robotics and Automation (ICRA'08)*, 2008.

[12] D. G. Lowe. Object recognition from local scale-invariant features. In *Proc. of the International Conference on Computer Vision (ICCV 1999)*, pages 1150–57, Corfu, Greece, September 1999.

[13] O. Martínez Mozos, P. Jensfelt, H. Zender, G.-J. Kruijff, and W. Burgard. From labels to semantics: An integrated system for conceptual spatial representations of indoor environments for mobile robots. In *Proc. of the Workshop "Semantic information in robotics" at the IEEE International Conference on Robotics and Automation (ICRA'07)*, Rome,Italy, April 2007.

[14] A. Ranganathan and F. Dellaert. Semantic modeling of places using objects. In *Proc. of Robotics: Science and Systems Conference (RSS06)*, 2003.

[15] M. Ribo and A. Pinz. A comparison of three uncertainty calculi for building sonar-based occupancy grids. In *SIRS*, Coimbra, Portugal, July 1999. A revised version will appear in Journal of Robotics.

[16] S. Vasudevan, S. Gächter, V. Nguyen, and R. Siegwart. Cognitive maps for mobile robots – an object based approach. *Robotics and Autonomous Systems*, 55:359–371, 2007.

[17] O. Wijk. *Triangulation Based Fusion of Sonar Data with Application in Mobile Robot Mapping and Localization*. PhD thesis, Royal Institue of Technology, Stockholm, Sweden, April 2001.