Towards an Informed Search Behavior for Domestic Robots

Leon Ziegler[1], Frederic Siepmann[1,2], Marco Kortkamp[1], Sven Wachsmuth[1,2]

[1]Applied Informatics, Faculty of Technology, Bielefeld University
[2]Central Lab Facilities, CITEC
Universitätstraße 25, 33615 Bielefeld, Germany
{lziegler,fsiepman,mkortkam,swachsmu}@TechFak.Uni-Bielefeld.DE

Abstract. In this paper we present an object search behavior for a mobile domestic robot that reduces the search space by applying a novel kind of spatial attention system. Different visual cues are mapped in a SLAM-like manner in order to identify hypotheses for possible object locations. These locations are scanned for known objects using a recognition method consisting of two complementary pathways – a detector measuring color distributions and a classifier using a SVM with a Pyramid Matching Kernel. We show the usefulness of the proposed approach by conducting an evaluation in a real world apartment scenario.

1 Introduction

Service robotics is a growing field of interest. To become feasible robots need to be able to communicate and interact with humans, but also need to autonomously perform tasks in regular domestic home environments. A basic task for such robots is "fetch and carry". The robot is instructed by a human to fetch a known object from another room and to deliver it to her. Variants of this task can be found in actual robot competitions, like the "mobile manipulation challenge" at ICRA 2010 (concentrating on grasping), the "semantic robot vision challenge" (concentrating on object finding), or the "robocup@home" competition. Despite the fact that the single skills needed to perform the task successfully, are well established, robots frequently fail if they need to show them in a realistic scenario. The glue to efficiently combine them in a coherent manner is typically underestimated.

In this paper, we concentrate on the first part of finding the object requested in a complex room environment. Here the performance of the object recognizer crucially depends on the navigation component and the related exploration behavior. Object recognition techniques have been highly optimized to image datasets that are acquired by human photographers, e.g. flickr datasets. Here, typically the human already solved the sub-task of centering the object at a proper resolution. Thus, the search space of the robot to find an appropriate view on the target object is huge and uninformed search will not succeed in a limited amount of time. Tsotsos et. al [16] already showed that the problem of visual matching is NP complete, the visual search task in a complex 3D environment is even worse. In [11], Shubina and Tsotsos argue for using attentive cues

Proceedings of SIMPAR 2010 Workshops
Intl. Conf. on SIMULATION, MODELING and PROGRAMMING for AUTONOMOUS ROBOTS
Darmstadt (Germany) November 15-16, 2010
ISBN 978-3-00-032863-3
pp. 241-250

that optimize the search process of a robot. They propose a greedy algorithm that considers the cost and effect of different actions. Various kinds of a priori knowledge are utilized, like objects in spatial proximity (also proposed by [6,18]), saliency knowledge (see e.g. [5]), or spatiotemporal constraints (see also [15,2]). The STAIR (Stanford Artificial Intelligent Robot) system includes an approach for peripheral-foveal vision, an integrated view-planning strategy is presented by [12], *Curious George* was developed for the semantic robot challenge [9]. It combines a geometric map for view and path planning and an attentive system based on 3D structure and visual appearance. A probabilistic approach utilizing semantic knowledge about the environment to find persons in a mobile robot's surrounding is proposed by [13].

In the following, we propose a target-directed search strategy for known objects in unknown rooms. It combines (i) a two-step object recognition approach that applies a color-based top-down attention filter as a first step, (ii) the exploitation of scene geometry for the extraction of appropriate places for objects in a room, (iii) a SLAM approach that dynamically builds a semantically annotated map of the room. The whole approach is evaluated in a real apartment and contrasted with a pure open-space exploration strategy.

2 Grid-Mapping Extension and Information Fusion

The basis for the semantically annotated map is an occupancy grid representing the spatial structure of the environment generated by a SLAM implementation [10]. This map contains only physical obstacles that can be detected by the laser range finder, such as walls and furniture. The information from this map is sufficient for planning simple navigation tasks and for measuring the robot's current position, but is not valuable for the selection of a promising *viewpoint* from which the possible object location can be inspected. The work of [12] assumes implicitly that the probability of each object's location is uniformly distributed over the entire known environment, so their proposed view planning approach tries to cover the whole reachable space with a complete object recognition scan. This behavior is very time consuming because areas containing no relevant visual information are analyzed using the computationally expensive recognition algorithm. The method proposed in this paper aims to reduce the number of actual scene analyzes through enrichment of the spatial map with peripheral visual

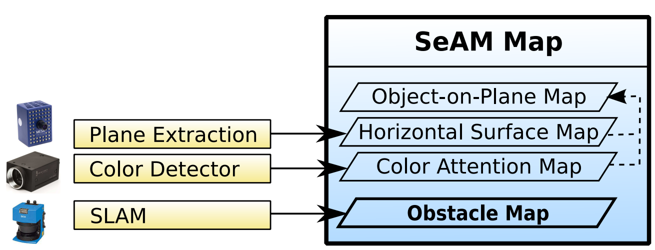


Fig. 1. Layout of the SeAM map.

Proceedings of SIMPAR 2010 Workshops
Intl. Conf. on SIMULATION, MODELING and PROGRAMMING for AUTONOMOUS ROBOTS
Darmstadt (Germany) November 15-16, 2010
ISBN 978-3-00-032863-3
pp. 241-250

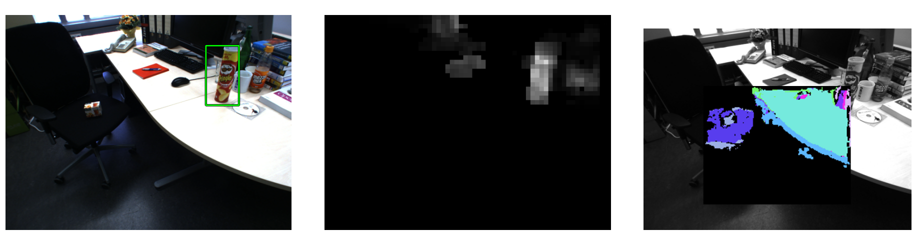
information. Additional grid map layers on top of the SLAM obstacle map are introduced by our "Semantic Annotation Mapping" approach (SeAM) to encode the low-level visual cues calculated while the robot explores its environment (see Fig. 1). These overlays are used for a more detailed analysis later on. Hence, the combination of these information can be considered as a mechanism for mapping spatial attention that constantly runs as a subconscious background process.

2.1 Vision Components

The object recognition capabilities of the robot are implemented as two complementary pathways. At first, potential object positions are detected within the robots visual field by using simple and computationally efficient visual features. E.g., it makes more sense to look for a red chips box in a cupboard with red stuff in it than to search for it on a green wall. After the robots moved to an interesting spot, it tries to classify whether a *target object* is present or not by employing more complex visual representations. To focus the search on horizontal surfaces, e.g. table tops, a component exists that extracts such surfaces from the scene. An example of the outcome of these components is depicted in Fig. 2.

Horizontal Surface Extraction. Similar to [9], we use the fact that objects are most likely placed on horizontal surfaces. Technically, the information about these surfaces in the current visual field analyzes a 3D point cloud received from a SwissRanger camera [14] (see Fig. 2(c)).

Color Distribution Detection. Suppose the robot searches for a known red box of chips, as seen in Fig. 2(a). The system loads the corresponding model from the memory and during the whole search process, it executes a fast detection component. The purpose of this detector is to identify potential locations of the chips within the robot's visual field by employing the known appearance of the target object. In this work, we use a search for the target color distribution quite similar to [4]. This detection could be interpreted as a kind of top-down or model-driven saliency. It is also possible to use other cues like shape and texture or to combine them, as in the work of [4]. Important requirements for a potential detector are its low computational complexity and applicability for low-pixel images of the object and changes in lighting, pose, scale, deformation, or occlusion.



(a) Original scene (b) Color matching (c) Planes in 3D space

Fig. 2. Results of the input sources for the attention mapping mechanism.

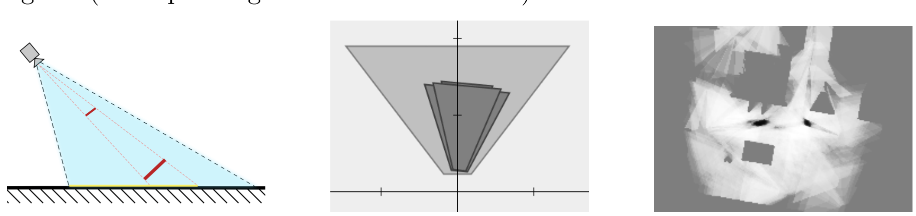
Proceedings of SIMPAR 2010 Workshops
Intl. Conf. on SIMULATION, MODELING and PROGRAMMING for AUTONOMOUS ROBOTS
Darmstadt (Germany) November 15-16, 2010
ISBN 978-3-00-032863-3
pp. 241-250

Classification. This component is not intended to be an input for the mapping, but to make a decision whether an image region that has been found by the detector after approaching a viewpoint actually contains the target object. By using different pathways and search states, our framework intrinsically supports the idea of applying computationally expensive processes very selectively. Hence, a classification is only invoked in valuable situations (i.e. the presence of positive detection results). We use a Support Vector Machine with a customized version of the Pyramid Matching Kernel [7] based on Speeded Up Robust Features [1] for the classification. Further, successive classification results are combined over time to enhance the stability.

2.2 Spatial Mapping

In order to register information-rich regions into the grid maps, the visual information need to be spatially estimated relatively to the robot's current position. The 3D plane description can be easily transformed into a 2D aerial view representation. In case of the color distribution cue, the direction of the detected location can be calculated using several facts about the camera's properties like FoV and resolution, as well as how it is mounted on the robot (see Fig. 3(a)). The size of the actual object in the real scene can be estimated by the size of the bounding box. But as the object's size in the image depends on its distance to the camera, the possible locations form a cone originating from the robot's position and pointing to the calculated relative direction of the seen cue (see Fig. 3(b)). The length of the cone in the direction of the detected cue can be estimated by the physical and visual limitations of the system. The cone begins at the nearest distance from where an object could be detected and ends at the distance where the resolution of the camera prevents a reliable detection. Additionally, we assume that the greatest possible distance of an object represented by a certain bounding box is reached, when it is lying on the floor. So the cone's maximum size is additionally limited by the corresponding floor distance of the center pixel in the detected bounding box.

The actual mapping of the found regions is done by raising or lowering the cell values of the corresponding layer in the SeAM map. If a cell is covered by the recognized cone its value is lowered, but is raised for cells which are covered by the robot's field of view and not the detected cone. This encoding is similar to the representation of the SLAM results. While values near 0.5 mean unknown area, higher values mean free space and lower values mean detected attention regions (corresponding to obstacles in SLAM).



(a) Visual cue (b) Robot's viewport (c) Color detector map

Fig. 3. Steps when mapping visual cues from color detector.

Proceedings of SIMPAR 2010 Workshops
Intl. Conf. on SIMULATION, MODELING and PROGRAMMING for AUTONOMOUS ROBOTS
Darmstadt (Germany) November 15-16, 2010
ISBN 978-3-00-032863-3
pp. 241-250

When an object is seen for the first time, the whole cone of possible locations will be registered in the grid map. However, while the robot moves, it may detect the same object from different angles. The probability for the object will be raised where the cones overlap. Probabilities for cells that were previously detected as possible object locations, will be lowered if the new results do not vote for those cells. Eventually, only the true location of the detected object will remain in the map (see Fig. 3(c)).

Because of the layer structure of the grid maps representing the same spatial area, information from multiple layers can be fused to generate more sophisticated data. When registering 2D color distribution results in the grid map, we assume that the corresponding object is not behind a wall or another tall obstacle. So, cells that correspond to obstacle cells in the SLAM layer or are positioned behind those cells in respect to the robot's viewpoint, are **not** altered and remain unchanged (see Fig. 3(c)). This leads to a problem when detected objects are placed on furniture like low cupboards or shelves that are detected as obstacles in the laser level. In this case detected cues would be ignored. To solve this problem, we introduce an additional grid map layer that fuses information from the color detector and the horizontal surface detector. Semantically this map represents object hypotheses on horizontal surfaces above the floor (*object-on-plane* map). The probabilities are only raised if both detectors vote for the same cell. More details can be found in [22].

3 Modeling of Logic and Behaviors

As mentioned before, the mapping of attention is implemented as a constantly running background process, so when the robot is initialized, it has no knowledge about its environment. Hence, it has to begin with a visual exploration of the scene to locate interesting objects or places. In order to find those locations the attention maps are analyzed for valuable regions and corresponding viewpoints.

Map Acquisition through Exploration. To gain initial information, the implemented behavior begins by exploring the area. The first step is a frontier-based exploration strategy as proposed by [21] using the SLAM map. It finds locations on the existing SLAM map, where the known free space fades to the unknown area. These frontiers are assumed to provide new information, so the robot just has to move to the nearest frontier repeatedly, to explore its environment.

Using this strategy exclusively is problematic, because the laser range is more far-reaching than the camera's view port. Important places may be missed by the cameras when the described approach will consider a certain space as already explored. However, the necessary information of areas covered by the camera's view port are encoded in a grid map similar to the SLAM map. This information can be used by applying the exploration algorithm to one of the camera's attention maps to perform a visual exploration (see Fig. 4(a)).

Viewpoint Computation. Viewpoints close to the actual objects are desired to receive enough pixels for the recognition component, as well as views from different angles to confirm the recognition result (see Fig. 4(c)). As a first step

Proceedings of SIMPAR 2010 Workshops
Intl. Conf. on SIMULATION, MODELING and PROGRAMMING for AUTONOMOUS ROBOTS
Darmstadt (Germany) November 15-16, 2010
ISBN 978-3-00-032863-3
pp. 241-250

(a) Visual exploration (b) Color detector map (c) Viewpoints

Fig. 4. Exploration goals and viewpoint calculation.

the map is binarized to distinguish attention cells which have a value less than 0.5 from unknown and clear regions. The resulting map is processed by a region growing algorithm. Next, we dilate each continuous region by an amount of cells that corresponds to roughly one meter in the real world and apply a sobel filter to reduce the found regions only to their edge cells. We assume that these cells have a reasonable distance to the detected location, but not all of them are appropriate as navigation goals. The obstacle map is used to delete all cells which are not reachable, do not have a minimum distance to obstacles, or are located in an unknown area. The remaining cells are clustered using the k-means algorithm [8]. The centroids of the resulting clusters are treated as viewpoints. Small clusters result in many viewpoints that can be used to analyze the attention region from many different angles, while large clusters result in a small number of viewpoints.

Behavior Implementation. The implemented behavior executes several states to perform the search task. It begins with a laser-based exploration state that switches to a camera-based visual exploration, when the current room is sufficiently explored by the laser. In regular intervals the current attention maps are analyzed for viewpoints. The analysis distinguishes strong viewpoint suggestions derived from the *object-on-plane* map and weak suggestions derived from a map representing only one of the visual cues. This is because we assume the combined map to be more valuable and more robust than the others. When a strong suggestion was found, the exploration is suspended and the robot approaches the viewpoint. When reached, the robot starts the classification described in Sec. 2.1. The result and its location is stored and the behavior continues analyzing the attention maps ignoring already visited locations. If no strong suggestion remains, the robot carries on with weak viewpoints. When all viewpoints were visited, the robot continues exploring until no further frontiers are found.

Architectural Embedding. The software architecture of our robot BIRON consists of many different components, each providing a certain functionality such as speech recognition or face detection. All components follow the concept of Information-Driven-Integration (IDI) [19] by sharing their data via an Active Memory [20] within the system. As it has been pointed out by Brugali [3] the *configuration* of the system, namely the connections between all components at runtime, is crucial for component-based systems such as the BIRON system. We have developed a tool called *BonSAI* to flexibly model the robot behavior on

Proceedings of SIMPAR 2010 Workshops
Intl. Conf. on SIMULATION, MODELING and PROGRAMMING for AUTONOMOUS ROBOTS
Darmstadt (Germany) November 15-16, 2010
ISBN 978-3-00-032863-3
pp. 241-250

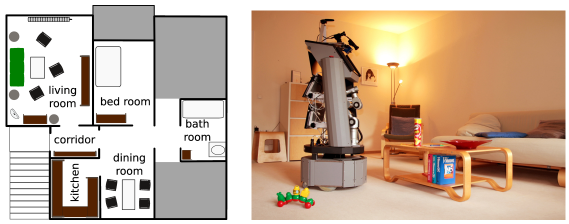


Fig. 5. The BIRON apartment (left: Overview – right: BIRON in the living room).

the one hand and also encapsulate the necessary system configuration on the other hand.

BonSAI behavior modeling. *BonSAI* is a domain-specific library that builds up on the concept of *sensors* and *actuators* which allow the linking of perception to action. The sensors and actuators encapsulate rather complex perception-action-linking processes, but take care of the system configuration and provide a simple interface. BonSAI facilitates the design of applications that focus on the modeling of the robot's behavior as e.g. described in Sec. 3. The behavioral states should be modelled locally, which means that e.g. all strategies for the specific behavior are included and are not spread over many states. The behaviors should be minimal, e.g. modeling one specific functionality of the robot.

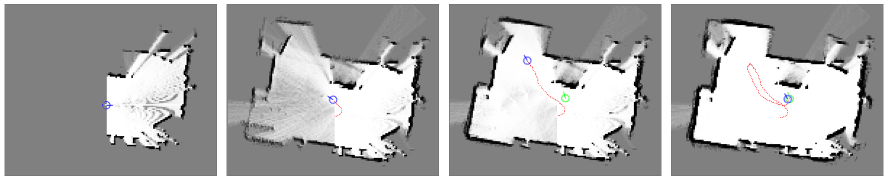
4 Evaluation

For evaluation we used a real world living room and placed three known objects on varying spots on the furniture. We compared the performance with an uninformed search behavior by measuring the number of correctly found objects and false detection results. When performing this competitive behavior, the robot's movement is not based on any assumptions concerning possible object positions, but is triggered by an exclusively reactive behavior led by the currently visible obstacles in the perception of the laser range finder. Every few seconds the robot turns to the nearest obstacle in order to search for a known object using the recognition component. As the uninformed behavior was expected to visit significantly more viewpoints in the same time period, the results were additionally normalized with the total of visited viewpoints which serves as a measurement for the effectiveness of the viewpoint choice.

4.1 Real World Experiments: Setup and Scenario

The *BIRON* hardware platform (see Fig. 5(b)) we use is based on the research platform *GuiaBot*™ by MobileRobots customized and equipped with sensors that allow analysis of the current situation in a human-robot interaction. A more detailed description of the hardware platform can be found in [17].

Proceedings of SIMPAR 2010 Workshops
Intl. Conf. on SIMULATION, MODELING and PROGRAMMING for AUTONOMOUS ROBOTS
Darmstadt (Germany) November 15-16, 2010
ISBN 978-3-00-032863-3
pp. 241-250

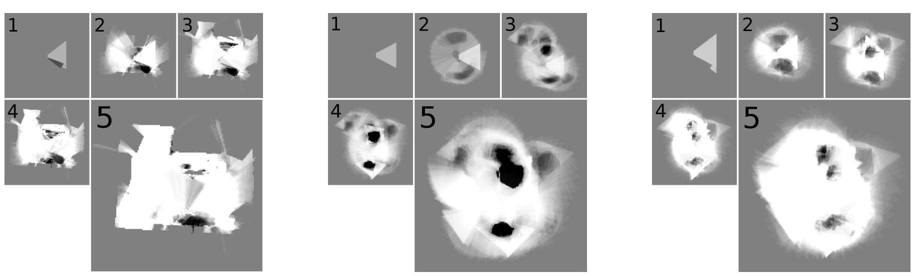


(a) (b) (c) (d)

Fig. 6. An example of the robot's movement. Depicted is the SLAM map at different stages of development. Blue circle: robot – Green circle: viewpoint.

A Real World Apartment For evaluating our system under real world conditions and carrying out user studies we have permanently rented an off-campus apartment in Bielefeld. One of our goals is to establish an evaluation-development cycle to iteratively improve the capabilities of our robots. One focus of our research here in particular is on the interaction design and adaptive interaction skills enabling naive users to successfully interact with the robot. The work presented here was evaluated in the apartment. The search task, as described in the next part, was carried out in the living room as depicted in Fig. 5(a).

An Example of the Search Behavior Fig. 6(a-d) shows the first few seconds of the robot's movement recorded at an exemplary evaluation run. The system begins with an exploration behavior that makes the robot turn in place and subsequently start to move towards the unknown region behind the furniture in the upper left area of the map (Fig. 6(b)). Due to the fact that the room is quite small, the robot decides after a short time that its exploration strategy roughly covered the whole area already. As a result the attention maps are analyzed. The third image of each progress picture in Fig. 7 shows the developed map at this point in time. A strong suggestion to take a look at the couch table from the viewpoint represented by the green mark in Fig. 6(c) is generated. Then the



(a) Color attention map (b) Horizontal plane map (c) Object-on-plane map

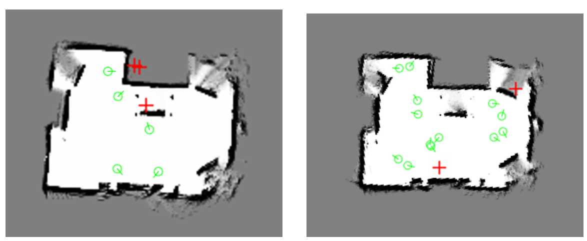
Fig. 7. Temporal progress of the attention maps from the upper left to the lower right image in each subfigure (exemplary evaluation run). Notice that the color attention regions shrink over time, when seen from different angles.

Proceedings of SIMPAR 2010 Workshops
Intl. Conf. on SIMULATION, MODELING and PROGRAMMING for AUTONOMOUS ROBOTS
Darmstadt (Germany) November 15-16, 2010
ISBN 978-3-00-032863-3
pp. 241-250

robot navigates to the computed viewpoint and starts the object recognition (Fig. 6(d)). The result will be saved and the search continues by analyzing the attention maps until no further viewpoints are found.

4.2 Results

We made the robot search the living room 15 times and for comparison we started the system 7 additional times with the previously described uninformed behavior. It turned out that the implemented search behavior identified correctly on average 1.40 objects per search and detected 1.36 false positives, while the uninformed approach only identified 0.86 correct objects per run and detected in contrast 2.14 false positives. These results already show an advantage for the implemented search behavior, but considering that the uninformed behavior did not have to struggle with navigation problems, we did an additional comparison based on the number of approached viewpoints. The uninformed behavior visited on average more than twice the number of viewpoints in the same period of time, compared to the implemented search behavior (1.61 `VP/sec` to 0.80 `VP/sec`). So we also measured the amount of *Correct Identifications per Viewpoint* (`C/VP`). The result of 0.22 `C/VP` when using the implemented search behavior compared to 0.06 `C/VP` by the uninformed behavior shows that the proposed approach provides a very effective strategy for searching objects.



(a) Search behavior (b) Uninformed behavior

Fig. 8. Exemplary results of the evaluation. The implemented search behavior (a) produced more reasonable viewpoints, than the uninformed behavior (b). The proposed searching approach found one correct object on the couch table and two times the item on the couch. The uninformed behavior found the object on the armchair in the upper right corner and accidentally found a wrong object in the shelf.

5 Conclusion

We presented a solution for the autonomous object finding part of the "fetch and carry" task by concentrating on an attention mechanism. We could show that the proposed mapping approach reasonably reduces the search space for the robot. In a set of experiments we evaluated the implemented system and could prove the applicability of the approach for object search in real world indoor environments.

Proceedings of SIMPAR 2010 Workshops
Intl. Conf. on SIMULATION, MODELING and PROGRAMMING for AUTONOMOUS ROBOTS
Darmstadt (Germany) November 15-16, 2010
ISBN 978-3-00-032863-3
pp. 241-250

References

1. Bay, H., Tuytelaars, T., Van Gool, L.: SURF: Speeded up robust features. In: ECCV (2006)
2. Bourgault, F., Furukawa, T., Durrant-Whyte, H.F.: Coordinated decentralized search for a lost target in a bayesian world. In: IROS. pp. 48–53 (2003)
3. Brugali, D., Scandurra, P.: Component-based robotic engineering (Part I). Robotics Automation Magazine 16(4), 84–96 (2009)
4. Ekvall, S., Kragic, D., Jensfelt, P.: Object detection and mapping for service robot tasks. Robotica 25(2), 175–187 (2007)
5. Frintrop, S.: VOCUS: A Visual Attention System for Object Detection and Goal-Directed Search, vol. 3899. Springer LNAI (2006)
6. Garvey, T.D.: Perceptual strategies for purposive vision. Tech. Rep. 117, SRI International (1976)
7. Grauman, K., Darrell, T.: Approximate correspondences in high dimensions. In: NIPS (2006)
8. MacQueen, J.B.: Some methods for classification and analysis of multivariate observations. In: Cam, L.M.L., Neyman, J. (eds.) Berkeley Symposium on Mathematical Statistics and Probability. vol. 1, pp. 281–297 (1967)
9. Meger, D., Gupta, A., Little, J.J.: Viewpoint detection models for sequential embodied object category recognition. In: ICRA (2010)
10. Montemerlo, M., Thrun, S., Koller, D., Wegbreit, B.: FastSLAM 2.0: an improved particle filtering algorithm for simultaneous localization and mapping that provably converges. In: IJCAI. pp. 1151–1156 (2003)
11. Shubina, K., Tsotsos, J.: Visual search for an object in a 3d environment using a mobile robot. Computer Vision and Image Understanding 114(5), 535–547 (2010)
12. Sjö, K., López, D.G., Paul, A., Jensfelt, P., Kragic, D.: Object search and localization for an indoor mobile robot. Journal of Computing and Information Technology 17(1), 67–80 (2009)
13. Stückler, J., Behnke, S.: Improving people awareness of service robots by semantic scene knowledge. In: International RoboCup Symposium, Singapore (2010)
14. Swadzba, A., Wachsmuth, S.: Categorizing perceptions of indoor rooms using 3d features. In: Workshop on Structural and Syntactic Pattern Recognition and Statistical Pattern Recognition. pp. 744–754 (2008)
15. Tovar, B., LaValle, S., Murrieta, R.: Optimal navigation and object finding without geometric maps or localization. In: ICRA. pp. 464–470 (2003)
16. Tsotsos, J.: The complexity of perceptual search tasks. In: IJCAI. pp. 1571–1577 (1989)
17. Wachsmuth, S., Siepmann, F., Schulze, D., Swadzba, A.: ToBI - Team of Bielefeld: The Human-Robot Interaction System for RoboCup@Home 2010. Tech. rep., RoboCup Singapore 2010 (2010)
18. Wixson, L., Ballard, D.: Using intermediate object to improve efficiency of visual search. Int. J. Comput. Vis. 18(3), 209–230 (1994)
19. Wrede, S.: An Information-Driven Architecture for Cognitive Systems Research. Ph.D. thesis, Bielefeld University (2008)
20. Wrede, S., Hanheide, M., Wachsmuth, S., Sagerer, G.: Integration and coordination in a cognitive vision system. In: ICVS (2006)
21. Yamauchi, B.: A frontier-based approach for autonomous exploration. In: ICRA (1997)
22. Ziegler, L.: Developing a Vision-Based Object Search Behavior for a Mobile Robot. Master's thesis, Bielefeld University (2010)

Proceedings of SIMPAR 2010 Workshops
Intl. Conf. on SIMULATION, MODELING and PROGRAMMING for AUTONOMOUS ROBOTS
Darmstadt (Germany) November 15-16, 2010
ISBN 978-3-00-032863-3
pp. 241-250