ToBI - Team of Bielefeld: The Human-Robot Interaction System for RoboCup@Home 2011

Sven Wachsmuth, Frederic Siepmann, Leon Ziegler, Florian Lier

Faculty of Technology, Bielefeld University,
Universitätstraße 25, 33615 Bielefeld, Germany

Abstract. The Team of Bielefeld (ToBI) was founded in 2009. The robocup activities are embedded in a long-term research history towards human-robot interaction with laypersons in regular home environments. The robocup@home competition is an important benchmark and milestone for the overall research goal. For robocup 2011, the team concentrates on mixed-initiative scenarios, sophisticated scene understanding methods including semantically annotated maps, and an easy to use programming environment.

1 Introduction

The Robocup@Home competition aims at bringing robotic platforms to use in regular home environments. Thus, the robot needs to deal with unprepared domestic environments, perform tasks in them, autonomously, and interact with laypersons. ToBI (Team of Bielefeld) has been founded in 2009 and successfully participated in the German Open 2009 and 2010 as well as the Robocup 2009 in Graz and RoboCup 2010 in Singapore. The robotic platform and software environment has been developed based on a long history of research in human-robot interaction [1, 2]. The overall research goal is to provide a robot with capabilities that enable the interactive teaching of skills and tasks through natural communication in previously unknown environments.

The challenge is two-fold. On the one hand, we need to understand the communicative cues of humans and how they interpret robotic behavior [3]. On the other hand, we need to provide technology that is able to perceive the environment, detect and recognize humans, navigate in changing environments, localize and manipulate objects, initiate and understand a spoken dialog. Thus, it is important to go beyond typical command-style interaction and to support mixed-initiative learning tasks. In the ToBI system this is managed by a sophisticated dialog model that enables flexible dialog structures [4, 5].

In this year's competition, we extend the scene understanding of our robot. Most robotic systems build a 2D map of the environment by using laser scans and associate semantic labels to certain places that are known beforehand or are interactively tought to the system like in the *walk-and-talk* task. However, there is no understanding of a table, desk, or sideboard where objects are typically placed on. In this year's Robocup@Home competition, we will make a first step towards this goal by integrating a 3D scene analysis component that is based on a 3D depth sensor. During exploration tasks, the system is continuously mapping the environment and is accumulating semantic information in an additional annotation layer.

Another focus of the system is to provide an easy to use programming environment for experimentation. An abstract sensor- and actuator interface (BonSAI) encapsulates the sensors, components, and behavior strategies of the system. Providing an easy to use Java-API, it allows to a fast modeling and iterative change of the robot behavior during experimental trials. The abstraction also allows to formulate, re-use, and compare behavior strategies, e.g. searching an environment, that are independent of specific tasks. As the student team members are changing every year, a steep learning curve can be observed using the BonSAI-API and associated tools.

2 The ToBI Platform

The robot platform *ToBI* is based on the research platform *GuiaBot*$^{\text{TM}}$ by MobileRobots[1] customized and equipped with sensors that allow for an analysis of the current situation. ToBI is a consequent advancement of the *BIRON* (**BI**lefeld **R**obot compani**ON**) platform in the RoboCup@Home context. It can be rooted to a continuous development originating in 2001 until now. It comprises two piggyback laptops to provide the computational power and to achieve a system running autonomously and in real-time for HRI. The robot base is a PatrolBot$^{\text{TM}}$ which is 59cm in length, 48cm in width, weighs approx. 45 kilograms with batteries. It is maneuverable with 1.7 meters per second maximum translation and 300+ degrees rotation per second. The drive is a two-wheel differential drive with two passive rear casters for balance. Inside the base there is a 180 degree laser range finder with a scanning height of 30cm above the floor (SICK LMS, see Fig.1 bottom right). In contrast to most other PatrolBot bases, ToBI does not use an additional internal computer. The piggyback

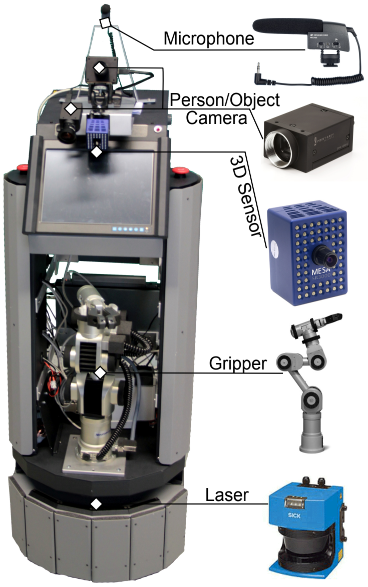


Fig. 1. The robot ToBI with it's components shown on the right. From top right: microphone, cameras for object/face detection, Swissranger 3D sensor, KATANA arm and laser range finder.

[1] www.mobilerobots.com

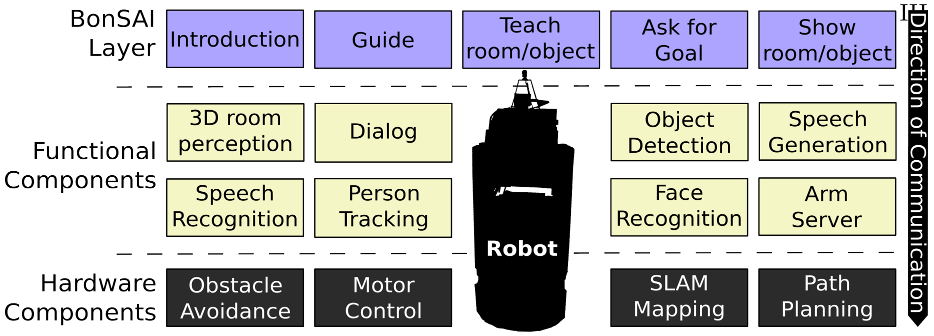


Fig. 2. Software components of the ToBI system and their level of abstraction from the hardware.

laptops are Core2Duo © processors with 2GB main memory and are running Ubuntu Linux. The cameras that are used for person and object detection/recognition are 2MP CCD firewire cameras (Point Grey Grashopper, see Fig.1). One is facing down for object detection/recognition, the second camera is facing up for face detection/recognition. For room classification and providing 3D object positions, ToBI is equipped with an optical imaging system for real time 3D image data acquisition.

Additionally the robot is equipped with a Katana IPR 5 degrees-of-freedom (DOF) arm (see Fig.1 second from bottom on the right); a small and lightweight manipulator driven by 6 DC-Motors with integrated digital position encoders. The end-effector is a sensor-gripper with distance and touch sensors (6 inside, 4 outside) allowing to grasp and manipulate objects up to 400 grams throughout the arm's envelope of operation. The upper part of the robot houses a touch screen ($\approx 15in$) as well as the system speaker. The on board microphone has a hyper-cardioid polar pattern and is mounted on top of the upper part of the robot. The overall height is approximately 140cm.

3 ToBI's Software Architecture

The software architecture of the ToBI system consists of many different components, each of which is a piece of software providing functionality, e.g. speech recognition, to the system. Figure 2 shows the different components that are used for the RoboCup 2011. The different colors from bottom to top refer to the level of abstraction from the hardware. All components follow the concept of Information-Driven-Integration (IDI) [6] by sharing their data via an active memory [7] within the system. Components in the black level at the bottom are either depending on direct sensory input or have a direct connection to the hardware. The *obstacle avoidance* for instance needs to get the input of the laser sensor (see bottom right on Fig. 1) at high frequencies to be able to detect obstacles while the robot is moving whereas the motor control represents a direct connection to the motors of the robot base to actually move the robot. Components of the light yellow level in the middle are not as depending on the robot's hardware and can facilitate information and/or functions provided by components of the layer below or of the same layer as indicated by the communication direction in Fig. 2. The *person tracking* e.g. fuses information from the lowest layer (Laser) as well as information from the face recognition. The upper layer in Figure 2 comprises components that facilitate the *BonSAI* library to implement certain skills of the robot, which are used to solve the various challenges of the RoboCup@HOME tasks.

As it has been pointed out by Brugali [8] the *configuration* of the system, the connections between components at runtime, is crucial for component-based systems such as the ToBI system. This configuration defines to a great extend what the robot is able to do at a certain point of time. This implies that the configuration needs to be dynamic to enable the system to react to changes in the environment. Our approach makes functionality of the robot feasible and explicitly models the desired robot behavior. With different scenarios and more complex tasks for a robot to interact in, this focus on designing the behavior of the robot and its interaction capabilities improves the system performance in complex environments. This is what we aim to achieve with BonSAI.

3.1 Modeling HRI

Sensors	**Actuators**
Laser	Navigation
Camera	Camera
Speech	Speech
Odometry	Arm
Position	Screen
Map	
Speed	
Object	
Person	

Table 1. The sensors and actuators available in BonSAI.

BonSAI is a domain-specific library that builds up on the concept of *sensors* and *actuators* that allow the linking of perception to action. The sensors and actuators that are provided by BonSAI are listed in table 3.1. These *sensors* and *actuators* reach beyond simple hardware abstraction by encapsulating complex perception-action-linking processes. The ability of BonSAI to configure the system allows to have simple interfaces, e.g. the *Person* sensor, which trigger a complex sequence of actions in the *Functional Component* level (see Fig. 2). One of the benefits of this approach is that a system configuration as described earlier now is linked to a specific skill of the robot. This is e.g. the *Follow Me* skill (see top level Fig. 2) that employs among others the *Person* sensor and the *Navigation* actuator. To give you an example: Within the *Follow Me*, the robot obviously needs to move, therefor the Navigation actuator is used which will trigger processes from the levels below (see Fig. 2) to navigate the robot to the desired position.

Calling the actuator automatically takes care of configuring the system and all necessary components. This behavior-oriented design (BOD) as e.g. proposed by Bryson [9] enables the developer to model the behavior and the interaction instead of the system configuration, which increases the flexibility of the interaction and strongly supports iterative system design. With BonSAI it is possible to model a robot behavior in such a way that the skills of the robot, e.g. following a person, and strategies, e.g. for recovering or searching, are modeled together to form more flexible and interactive behaviors.

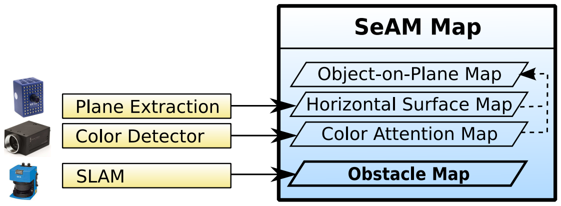


Fig. 3. Layout of the SeAM map.

The paradigm for our iterative design approach is to model the robot behavior locally, which means that strategies as described above are part of the behavior and are not spread over many components or rules. The behaviors also should be minimal, e.g. modeling one functionality of the robot such as *Follow Me*, and should be modular to enable combination of many behaviors for a specific scenario such as the RoboCup. This also eases up the design process of a robot behavior for developers, because they don't have to design rules for the different component configurations, which would require a detailed knowledge of the whole system. Additional experience gained over the years via user studies and the RoboCup can be implemented into individual behaviors. This makes improvements of the interaction measurable for each behavior in a quantitative manner and can improve the overall robot performance.

4 Semantic Map Annotation and Information Fusion

In order to improve the effectiveness of search tasks, the robot performs a scene analysis of its environment and builds up a 2D representation of the possibly most interesting regions. 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. 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. 3). 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.

4.1 Vision Components

In the case of *lost-and-found* tasks, the annotation component relies on two low-level visual cues to establish the attention map. At first, potential object positions are detected within the robot's 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. Additionally we detect horizontal surfaces in the perceived environment. An example of the outcome of these components is depicted in Fig. 4.

Horizontal Surface Extraction. Similar to [11], 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 [12] (see Fig. 4(c)).

Color Distribution Detection. Suppose the robot searches for a known red box of chips, as seen in Fig. 4(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 [13]. 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.

4.2 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. 5(a)).

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 cue region its value is lowered, but is raised for cells which are covered by the robot's field of view and not the detected region. 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).

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. 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-*

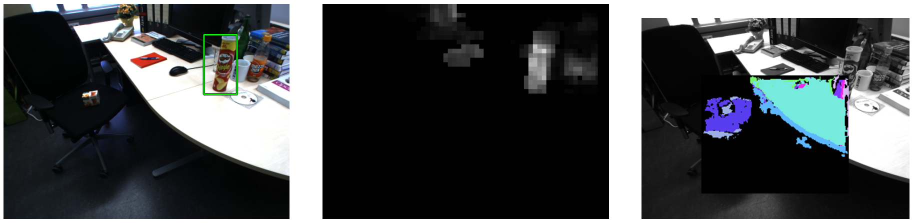
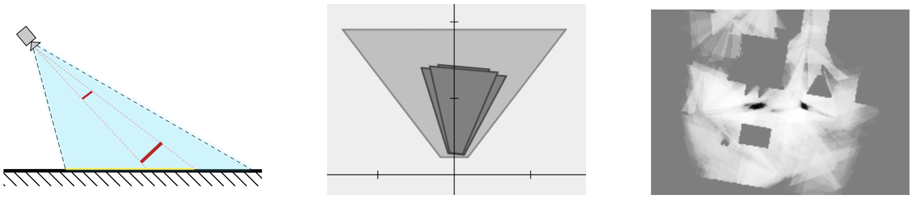

(a) Original scene (b) Color matching (c) Planes in 3D space

Fig. 4. Results of the input sources for the attention mapping mechanism.



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

Fig. 5. Steps when mapping visual cues from color detector.

on-plane map). The probabilities are only raised if both detectors vote for the same cell. More details can be found in [14].

4.3 Map Acquisition through Exploration

To gain initial information about the environment we usually use a frontier-based exploration strategy as proposed by [15] using the SLAM map. In order to perform a visual exploration using the robot's camera, the necessary information of areas covered by the camera's view port are encoded in the SeAM map which is 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.

When performing an actual search, the software can provide viewpoints that provide a reasonable view on the interesting areas. 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.

5 Conclusion

We have described the main features of the ToBI system for Robocup 2011 including sophisticated approaches for person detection and 3D scene analysis. BonSAI represents a flexible rapid prototyping environment, providing capabilities of robotic systems by defining a set of essential functions for such systems.

The RoboCup@HOME competition in 2009 served as an initial benchmark of the newly adapted platform. The Team of Bielefeld (ToBI) finished 8th place, starting with the new hardware and no experience in competitions like RoboCup. The determined tasks had to be designed from scratch because there where no such demands for our platform prior to the RoboCup competition. BonSAI with its abstraction of the system functionality proved to be very effective for designing determined tasks, e.g. the Who-is-Who task where the robot has to autonomously find three persons in the arena and re-identify them at the entrance door of the arena in a given time. This scenario is well defined for a script-like component as the number of people in the scene is known in advance and also what actions the robot should take. Additionally the runtime of the task can be used as ultimate trigger for the robot's behavior. In contrast, other tasks like the open challenges or General Purpose Service Robot task have no determined set of goals. Here a flexible human-robot interaction is much more in focus, the robot needs to deal much more flexible with its capabilities and an enriched understanding of the environment is essential. In this paper we presented some avenues towards this goal.

References

1. Wrede, B., Kleinehagenbrock, M., Fritsch, J.: Towards an integrated robotic system for interactive learning in a social context. In: Proc. IEEE/RSJ Int. Conf. on Intelligent Robots and Systems - IROS 2006, Bejing (2006)
2. Hanheide, M., Sagerer, G.: Active memory-based interaction strategies for learning-enabling behaviors. In: International Symposium on Robot and Human Interactive Communication (RO-MAN), Munich (01/08/2008 2008)
3. Lohse, M., Hanheide, M., Rohlfing, K., Sagerer, G.: Systemic Interaction Analysis (SInA) in HRI. In: Conference on Human-Robot Interaction (HRI), San Diego, CA, USA, IEEE, IEEE (11/03/2009 2009)
4. Peltason, J., Wrede, B.: Pamini: A framework for assembling mixed-initiative human-robot interaction from generic interaction patterns. In: SIGDIAL 2010 Conference, Tokyo, Japan, Association for Computational Linguistics, Association for Computational Linguistics (24/09/10 2010)
5. Li, S., Wrede, B., Sagerer, G.: A computational model of multi-modal grounding. In: Proc. ACL SIGdial workshop on discourse and dialog, in conjunction with COLING/ACL 2006, ACL Press, ACL Press (2006) 153–160
6. Wrede, S.: An Information-Driven Architecture for Cognitive Systems Research. PhD thesis, Bielefeld University (2008)
7. Wrede, S., Hanheide, M., Wachsmuth, S., Sagerer, G.: Integration and coordination in a cognitive vision system, St. Johns University, Manhattan, New York City, USA, IEEE, IEEE (2006)
8. Brugali, D., Shakhimardanov, A.: Component-based robotic engineering (part ii). Robotics Automation Magazine, IEEE **17**(1) (mar. 2010) 100 –112
9. Bryson, J.: The Behavior-Oriented Design of Modular Agent Intelligence. In Carbonell, J., Siekmann, J., Kowalczyk, R., Müller, J., Tianfield, H., Unland, R., eds.: Agent Technologies, Infrastructures, Tools, and Applications for E-Services. Volume 2592 of Lecture Notes in Computer Science. Springer Berlin / Heidelberg (2010) 61–76
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: Int. Joint Conf. on Artificial Intelligence. (2003) 1151–1156
11. Meger, D., Gupta, A., Little, J.J.: Viewpoint detection models for sequential embodied object category recognition. In: Robotics and Automation. (2010)
12. Swadzba, A., Wachsmuth, S.: Categorizing perceptions of indoor rooms using 3d features. In: Lecture Notes in Computer Science: Structural, Syntactic, and Statistical Pattern Recognition. Volume 5342. (2008) 744–754
13. Ekvall, S., Kragic, D., Jensfelt, P.: Object detection and mapping for service robot tasks. Robotica **25**(2) (2007) 175–187
14. Ziegler, L., Siepmann, F., Kortkamp, M., Wachsmuth, S.: Towards an informed search behavior for domestic robots. In: Domestic Service Robots in the Real World. (2010)
15. Yamauchi, B.: A frontier-based approach for autonomous exploration. In: Robotics and Automation. (1997)