# homer@UniKoblenz

Viktor Seib, Raphael Memmesheimer, Florian Polster, Ivanna Mykhalchyshyna, Dietrich Paulus
Active Vision Group (AGAS)
University of Koblenz and Landau
Universitätsstr. 1, 56070 Koblenz, Germany
vseib@uni-koblenz.de, http://homer.uni-koblenz.de

***Abstract*—This paper describes the robot hardware and software used by team homer@UniKoblenz of the University of Koblenz and Landau, Germany, for the participation in the RoCKIn@Home competition 2015 in Lisbon. A special focus is put on detailed hardware and software description and novel scientific achievements. For navigation and mapping we use well-established SLAM techniques based on particle filter and grid maps. Object recognition is achieved by clustering of local invariant features. Three-dimensional scans of the environment are acquired using a RGB-D sensor mounted on a pan-tilt unit. In order to manipulate objects a 6 DOF industrial-grade robotic arm is employed. An abstraction layer in our software architecture allows for task planning and general purpose task execution. For the improvement of human-robot interaction we developed a generic face model that is synchronized to speech and can show seven different face expressions. This robot face is available as a ROS-node for other teams. Some of the novelties of this year is a new robotic platform and a complete change to ROS that now replaces our previously used software architecture. Further, our object recognition, mapping and navigation algorithms were published as ROS package. Furthermore, improvements were made in speech recognition, object recognition, manipulation as well as 3D mapping.**

## I. Introduction

Our team homer@UniKoblenz was formed to participate in the RoboCup@Home competition. Our team already participated successfully as finalist in Suzhou, China (2008), Graz, Austria (2009) in Singapur (2010), where it was honored with the RoboCup@Home Innovation Award, in Mexico-City, Mexico (2012), where it was awarded the RoboCup@Home Technical Challenge Award and in Eindhoven (2013). Further, we participated in stage 2 at the RoboCup@Home World Championship in Istanbul, Turkey (2011). In 2015 we were placed second in the German Open and won the overall first place at the RoboCup@Home track in Hefei, China.

Besides RoboCup@Home we competed in the RoboCup Rescue league with our robot *Robbie*, where our team won the Interleague Mapping Challenge award (2010) and became German Champion in Rescue Autonomy for the 5th time in 2011.

A part of our current team participated in the RoCKIn@Home camp 2014 were we won the best final presentation. In 2014 we also participated in the RoCKIn@Home competition were we won the overall first price and were placed second in the object recognition benchmark.

In 2015 we plan to attend the RoCKIn@Home competition in Lisbon with our robot *Lisa* (Fig. 1). Our team will be presented in the next section. Section III is devoted to the hardware used for Lisa. In Section IV our software architecture, autonomous navigation and human-robot interaction will be described. Finally, Section V will summarize this paper.


Fig. 1. Current setup of our robot Lisa.

## II. About our Team

The members of team *homer@UniKoblenz* are mostly undergraduate and Master students from the University of Koblenz and Landau, Germany. They develop and improve our robot in practical courses offered by the Active Vision Group (AGAS). Head of the group is Prof. Dr.-Ing. Dietrich Paulus.

Since most practical courses last one semester only, it is natural that most of the team members are replaced on a regular basis. Each practical course is supervised by PhD students who act as scientific supervisors, but also incorporate their research into the project.

### A. Team Members and their Contributions

This year, team *homer@UniKoblenz* consists of the following members, listed with their contributions:
Viktor Seib: team leader, scientific supervisor, object recognition

Raphael Memmesheimer: system integration, speech recognition
Florian Polster: hardware integration, grid mapping and navigation
Ivanna Mykhalchyshyna: person recognition

### B. Focus of Research

The current focus of research is object recognition with visual and shape features. Novel approaches related to Implicit Shape Models (ISM) [1], as well as approaches for affordance detection are currently tested and integrated into the software collection of Lisa.

Additionally, with large member fluctuations in the team, as is natural for a student project, comes a necessity for an architecture that is easy to learn, teach and use. We thus migrated from our classic architecture *Robbie* [2] to the Robot Operating System (ROS) [3]. We developed an easy to use general purpose framework based on the ROS action library that allows us to create new behaviors in a short time.

## III. Hardware

### A. Robot Platform

As a mobile robotic platform we use the CU-2WD-Center, manufactured by UlrichC[1]. It is equipped with odometry sensors and has a 2 wheel drive in contrast to the 4 wheel drive of our previous platform. It allows the robot to turn on the spot with significantly less friction and thus, saves battery power and preserves the motors. On top of the platform, we have installed a prototype framework, which was designed and built by the Center of Excellence of the Chamber of Crafts in Koblenz and is able to carry additional sensors and a notebook running the control software.

### B. Sensors and additional Hardware

*1) Notebook:* The software of the robot runs on a Lenovo W520 notebook equipped with an Intel Core i7-2670QM processor and 12 GB of RAM using Ubuntu Linux 14.04 as operating system.

*2) SICK LMS100 laser range finder:* The SICK LMS100 is mounted at the bottom and generates 180° scans. It has an adjustable angular resolution, while its maximal measured distance is 20 m. It is used for mapping, localization and people tracking.

*3) DirectedPerception PTU-D46 pan-tilt unit:* The DirectedPerception PTU-D46 is mounted on top of the robot's neck. It is able to rotate $159°$ in each direction and to tilt from $+31°$ to $-47°$ out of a horizontal position. The angular resolution is $0.012857°$. Further sensors are attached on top of this unit.

*4) Rode VideoMic Pro Microphone:* The microphone Rode VideoMic Pro is used for speech input. It is a light-weight microphone and is mounted on the pan-tilt unit of our robot.

*5) IDS UI-5580CP-C-HQ camera:* The new sponsored IDS camera[2] is used to provide high resolution images to enhance object recognition. It is connected over ethernet and provides images with a resolution of 5 megapixels at 14 Hz.

*6) ASUS Xtion Pro live:* We replaced the Microsoft Kinect by an ASUS Xtion Pro live. Though both sensors provide the same data (depth and color images of size 640×480 pixels at 30 Hz), the Xtion camera does not need a separate power supply. Further, its new firmware is able to register color and depth images on the sensor, whereas the Kinect camera uses a software driver and thus increases the CPU payload.

*7) Neuronics Katana 400HD:* The Katana 400HD is a 6 DOF industrial-grade robot arm. It is attached to our robot's body plate and used to manipulate objects on tables and other furniture of similar height. With an accuracy of 1 mm and a length of 90 cm, it enables us to perform delicate manipulation tasks on light-weight objects.
The end effector is a custom setup and consists of 4 Festo Finray-fingers. It is safe for interaction with humans.

## IV. Technology and Scientific Contribution

In this section we will describe the architecture as well as the software components we developed. In the upcoming Springer Book on Robot Operating System most of our software will be described more detailed and includes a step by step guide how to set up the ROS nodes.

### A. System Core Architecture

In 2014, we completely migrated all of our software to ROS to benefit from a homogeneous architecture. Further, using this standard middle ware allows us to use code already available for ROS, but also to share some of our components with the community. For abstraction we make strong usage of actions which encapsulate basic functionality for straight forward reuse.

### B. Graphical Interface

We also migrated our GUI to ROS. It can be run directly on the robot or on a different computer via WLAN. The user interface is realized using Qt4 and OpenGL, further, with the migration to ROS we included the rviz plugin. A visualization example is given in Fig. 2.

### C. Simultaneous Localization and Mapping

To enable users without technical knowledge to use the robot and to ease the setup procedure, it has to be able to create a map without the help of a human. For this purpose, our robot continuously generates and updates a 2D map of its environment based on laser scans. Figure 3 shows an example of such a map.
To detect obstacles below and above the LRF-plane we use Kinect data to augment the occupancy map.

We recently published our mapping and navigation algorithms as ROS packages.

[1] Manufacturer of our new robotic platform: http://www.ulrichc.de

[2] IDS imaging: http://www.ids-imaging.com

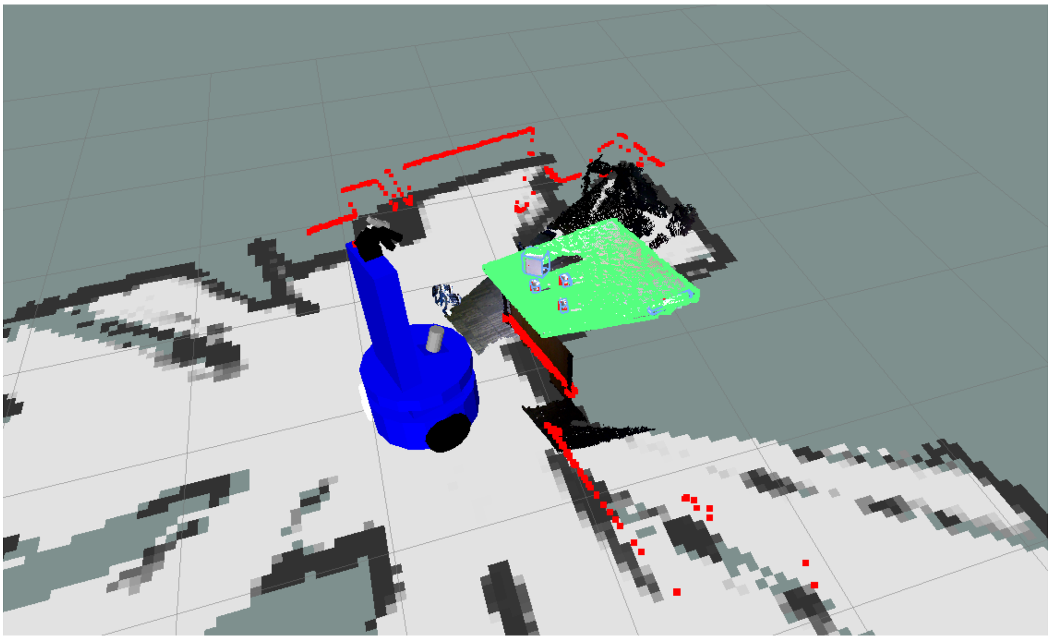

Fig. 2. Visualization of RGB-D and laser sensor (red) data when the robot is standing in front of a table. Also the grippable objects (light blue box) on the table and the segmented table planes (green) are visualized.

Fig. 3. Real-time maps of the RoboCup 2008 (left) and 2009 (right) @Home arena.

### *D. Navigation in Dynamic Environments*

In real-life situations, a static grid map is not sufficient for navigating through an everyday environment, as due to the movement of persons and other dynamic obstacles, an occupancy map that only changes slowly in time does not provide sufficient information.

Thus, our navigation system, which is based on Zelinsky's path transform [4], [5], always merges the current laser range scan as a frontier into the occupancy map. A once calculated path is then checked against obstacles in small intervals during navigation, which can be done at very little computational expense. If an object blocks the path for a given interval, the path is re-calculated. This approach allows the robot to efficiently navigate and avoid obstacles in highly dynamic environments.

### *E. Human-Robot Interface*

The robot is equipped with speakers and a microphone which enables communication via speech interface. In addition, it has a small screen that is used to display facial expressions and state information. For speech synthesis we use festival[3]. For speech recognition we use a grammar based solution by Nuance VoCon [4].

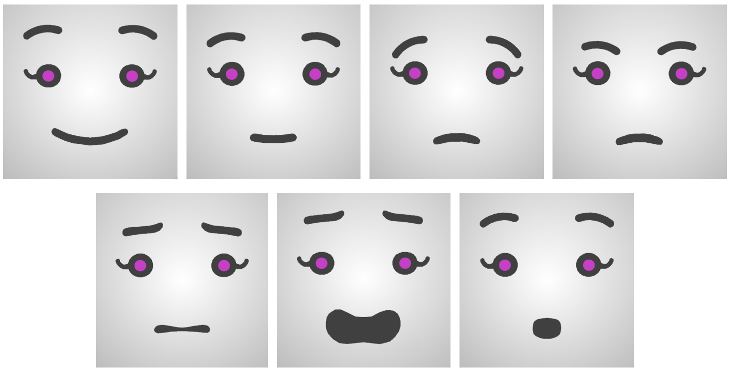

Fig. 4. Animated face of our service robot Lisa. The depicted face expressions are (from left to right): happy, neutral, sad, angry, disgusted, frightened, and surprised.

We have designed a concept of a talking robot face that is synchronized to speech via mouth movements. For this feature we are using the open source library Ogre3D[5] for visualisation. Furthermore, the face is modelled with Blender[6] and exported via the Ogre Mesh Exporter[7] for the use with Ogre.

[3] Speech synthesis system festival http://www.cstr.ed.ac.uk/projects/festival/
[4] http://www.nuance.com/for-business/speech-recognition-solutions/vocon-hybrid/
[5] Open-source graphics rendering engine Ogre3D http://www.ogre3d.org/
[6] Free open source 3D content creation suite Blender http://www.blender.org/
[7] Ogre Mesh Exporter http://www.ogre3d.org/tikiwiki/Blender+Exporter

To include the robot face into our software, we have created a ROS node for this application. We extended the robot face to show seven different face expressions (Figure 4). Further, we provide two similar face meshes, a female and a male one. The colors and the voice (female or male) can be configured via a configuration file without recompiling the application. The robot face has been released as an open source package for ROS[8]. Thus, it is possible to apply it for any robot. Furthermore, we conducted a broad user study to test how people perceive the shown emotions. The results can be found in [6].

To further enhance communication with humans a robot has to detect their emotions. One step towards this goal is the recognition of facial expressions. We use an approach based on the facial action coding system and action units. In the first step features are extracted from the detected face and classified into action units. The subsequent step analyzes the obtained action units and assigns a facial expression. Both classification steps use a support vector machine. Our system is able to recognize the same 7 face expressions that Lisa can exhibit. This approach works without prior calibration with a neutral face expression of the detected person.

### F. Object detection

Objects for mobile manipulation are detected by first segmenting horizontal planes as table hypotheses. Subsequently, all points above the plane are clustered and the resulting clusters considered as objects.

Transparent objects (in our case drinking glasses) are detected by making use of one fault of the Kinect sensor. The structured light emitted by the Kinect is scattered in transparent objects providing no valid depth information. We segment areas with no depth data and compare them to holes in detected planes to extract contours and match them with drinking glass contour templates. Since the supporting table plane around the transparent objects has valid depth information, a size and location estimation of the transparent objects is obtained and used for grasping.

### G. Object and Face Recognition

The object recognition algorithm we use is based on Speeded Up Robust Features (SURF) [7], which are local scale-invariant features of gray images. First, features are matched between the trained image and the current camera image based on their euclidean distance. A threshold on the ratio of the two nearest neighbours is used to filter unlikely matches. Then, matches are clustered in hough space using a four dimensional histogram using their position, scale and rotation. This way, sets of consistent matches are obtained. The result is further optimized by calculating a homography between the matched images and discarding outliers. Our system was evaluated in [8] and shown as suitable for fast training and robust object recognition.

With this object recognition approach we won the Technical Challenge 2012. Our object recognition is available as open source software in ROS[9]. A detailed description of the approach is provided on the website and in [9] as well.

### H. 3D Object Recognition

For 3D object recognition we use a continuous Hough-space voting scheme related to Implicit Shape Models (ISM). In our approach [10], SHOT features [11] from segmented objects are learned. Contrary to the ISM formulation, we do not cluster the features. Instead, to generalize from learned shape descriptors, we match each detected feature with the $k$ nearest learned features in the detection step. Each matched feature casts a vote into a continuous Hough-space. Maxima for object hypotheses are detected with the Mean Shift Mode Estimation algorithm [12].

### I. People Detection

People are detected by the combination of three sensors. The laser range finder is used to detect legs. The RGB image of the Kinect camera provides data for face detection. We use the face detection algorithm implemented in the OpenCV library. Finally, the Kinect depth camera allows to detect silhouettes of persons. For a person to be detected, not every sensor has to see the person. However, the more sensors see a person the higher the probability to really encounter a person at the position in question.

The silhouette detection works in three steps. First, candidate silhouettes are found. In order to do this, the depth data is split in overlapping slices along the forward axis. Each slice is then tested for point clusters roughly matching a human silhouette (size and round shape of the head). In a second step low-frequency fourier features are extracted from each candidate silhouette. In the last step a support vector machine classifies the feature vector as a non-human object, a lateral silhouette or as frontal silhouette.

### J. People Tracking

For the perception of people we combine the three methods described in IV-I. In case that these sensors yield similar points, the output is merged. Thus, it is assumed that the points belong to the same person. The obtained person positions are used as input for the particle filter which estimates the movement and position of the people.

### K. Object manipulation

Our system detects planes in the acquired 3D point cloud.This information is used to find euclidean point clusters on top of planes which fit into the gripper and are thus regarded as candidates for grasping.

If a specific object has to be grasped, one of the color cameras is adjusted to face the object cluster. The object detection algorithm is executed on the region of interest defined by the object cluster's bounding box.

The movement planning for our robotic arm is performed using an approach operating directly in working space. Chaining motion primitives, our path planner builds a graph from the

[8]Robot Face: http://www.ros.org/wiki/agas-ros-pkg

[9]Object Recognition software and detailed description: http://www.ros.org/wiki/agas-ros-pkg

starting position to the goal. The planning can be optimized towards specific objectives. These are performing a smooth path or keeping a maximum distance from obstacles using heuristic and cost functions [13].

### L. *Affordance Detection*

Affordances have gained much popularity for object classification and scene analysis. Our current research focuses on sitting affordances to analyze scenes regarding sitting possibilities for an anthropomorphic agent. Recently, we introduced the concept of fine-grained affordances [14]. It allows to distinguish affordances on a fine-grained scale (e.g. sitting without backrest, sitting with backrest, sitting with armrests) and thus facilitates the object classification process. Additionally, our approach estimates the sitting pose with regard to the detected object (Figure 5).

Fig. 5. Detected sitting affordances and the corresponding sitting pose of an anthropomorphic agent.

## V. Conclusion

In this paper, we have given an overview of the approach used by team homer@UniKoblenz for the RoCKIn@Home competition. We presented a combination of out-of-the box hardware and sensors and a custom-built robot framework. We explained the use of well-established techniques like SLAM based on a particle filter and a grid map. Further, we explained our approach for object recognition using matching and clustering of local invariant features and the ability to detect and manipulate objects with a 6 DOF robotic arm. Finally, we presented our software contribution to the community. So far, this is the animated robot face, the object recognition algorithm based on SURF, grid based mapping and navigation. These packages can be downloaded at http://wiki.ros.org/agas-ros-pkg.

## References


[1] B. Leibe, A. Leonardis, and B. Schiele, "Combined object categorization and segmentation with an implicit shape model," in *ECCV' 04 Workshop on Statistical Learning in Computer Vision*, 2004, pp. 17–32.

[2] S. Thierfelder, V. Seib, D. Lang, M. Häselich, J. Pellenz, and D. Paulus, "Robbie: A message-based robot architecture for autonomous mobile systems," *INFORMATIK 2011-Informatik schafft Communities*, 2011.

[3] M. Quigley, K. Conley, B. P. Gerkey, J. Faust, T. Foote, J. Leibs, R. Wheeler, and A. Y. Ng, "Ros: an open-source robot operating system," in *ICRA Workshop on Open Source Software*, 2009.

[4] A. Zelinsky, "Robot navigation with learning," *Australian Computer Journal*, vol. 20, no. 2, pp. 85–93, 5 1988.

[5] A. Zelinsky, "Environment exploration and path planning algorithms for a mobile robot using sonar," Ph.D. dissertation, Wollongong University, Australia, 1991.

[6] V. Seib, J. Giesen, D. Grüntjens, and D. Paulus, "Enhancing human-robot interaction by a robot face with facial expressions and synchronized lip movements," in *21st International Conference in Central Europe on Computer Graphics, Visualization and Computer Vision*, V. Skala, Ed., 2013. [Online]. Available: http://www.uni-koblenz.de/~agas/Documents/Seib2013EHI.pdf

[7] H. Bay, T. Tuytelaars, and L. Van Gool, "SURF: Speeded up robust features," *ECCV*, pp. 404–417, 2006. [Online]. Available: http://www.vision.ee.ethz.ch/~surf/eccv06.pdf

[8] P. Decker, S. Thierfelder, D. Paulus, and M. Grzegorzek, "Dense Statistic Versus Sparse Feature-Based Approach for 3D Object Recognition," in *10th International Conference on Pattern Recognition and Image Analysis: New Information Technologies*, vol. 1. Moscow: Springer MAIK Nauka/Interperiodica, 12 2010, pp. 181–184.

[9] V. Seib, M. Kusenbach, S. Thierfelder, and D. Paulus, "Object recognition using hough-transform clustering of surf features (report on the technical challenge of the robocup@home 2012, mexico-city)," 2013. [Online]. Available: http://www.uni-koblenz.de/~agas/Documents/Seib2013ORU.pdf

[10] V. Seib, N. Link, and D. Paulus, "Implicit shape models for 3d shape classification with a continuous voting space," in *Proceedings of International Joint Conference on Computer Vision, Imaging and Computer Graphics Theory and Applications*, 2015, to appear.

[11] F. Tombari, S. Salti, and L. Di Stefano, "Unique signatures of histograms for local surface description," in *Proc. of the European conference on computer vision (ECCV)*, ser. ECCV'10. Berlin, Heidelberg: Springer-Verlag, 2010, pp. 356–369.

[12] Y. Cheng, "Mean shift, mode seeking, and clustering," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 17, no. 8, pp. 790–799, 1995.

[13] B. Cohen, S. Chitta, and M. Likhachev, "Search-based planning for manipulation with motion primitives," Anchorage, Alaska, 2010.

[14] V. Seib, N. Wojke, M. Knauf, and D. Paulus, "Detecting fine-grained affordances with an anthropomorphic agent model," in *Computer Vision - Workshops of ECCV 2014*, D. Fleet, T. Pajdla, B. Schiele, and T. Tuytelaars, Eds. Springer, 2014.