RoboCup 2014 - homer@UniKoblenz (Germany)

Viktor Seib, Raphael Memmesheimer, Tatjana Jakowlewa, David Nagel, Niklas Gard, Denise Dünnebier, Anatoli Eckert, Eva Kreckel, Alexander Kreutz, Ivanna Mykhalchyshyna, Greta Rettler, Florian Sattler, Alruna Veith, Malte Knauf, Arne Peters, Dietrich Paulus

Active Vision Group
University of Koblenz and Landau
Universitätsstr. 1
56070 Koblenz, Germany
`vseib@uni-koblenz.de`
`http://userpages.uni-koblenz.de/~robbie`

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 at the RoboCup@Home 2014 in João Pessoa. A special focus is put on novel scientific achievements and newly developed features with respect to last year's competition. 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 algorithm was published a ROS package.

1 Introduction

Our team homer@UniKoblenz has 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 Instanbul, Turkey (2011). A part of our team also participated in the RoCKIn@Home 2014 challenge were we won the best final presentation award.

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.

In 2014 we will attend the RoboCup@Home in Brazil with our robot *Lisa* (Fig. 1). Our team will be presented in the next section. Section 3 describes the hardware used for Lisa. In Section 4 our software architecture, autonomous navigation and human-robot interaction will be discussed. Finally, Section 5 will summarize this paper.

2 About our Team

The members of team *homer@UniKoblenz* are students from the University of Koblenz and Landau, Germany. They develop and improve our robot in practical courses.

2.1 Team Members and their Contributions

Team *homer@UniKoblenz* consists of the following members and their contributions.

Fig. 1. Current setup of our robot Lisa.

Viktor Seib:	team leader *homer@UniKoblenz*, scientific supervisor
Raphael Memmesheimer:	team leader assistant, programming
David Nagel:	technical chief designer, programming
Denise Dünnebier:	team building representative, programming
Niklas Gard:	quality assurance, programming
Anatoli Eckert:	website, programming
Eva Kreckel:	flexible (wo)manpower, programming
Alexander Kreutz:	hardware, programming
Ivanna Mykhalchyshyna:	media, programming
Greta Rettler:	infrastructure, programming
Florian Sattler:	hardware, programming
Alruna Veith:	public relations, programming
Tatjana Jakowlewa:	assistant helper
Arne Peters:	bachelor's thesis (General Purpose)
Malte Knauf:	student research project (Mapping and Navigation using ROS)

2.2 Focus of Research

Our interests in research are sensor fusion for person tracking, people detection, object, face and gesture recognition as well as object manipulation using a manipulator.

3 Hardware

3.1 Robot Platform

(New in 2014) Currently, we are migrating to a new robotic platform, 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.

---

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

3.2 Sensors and additional Hardware

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 13.04 as operating system.

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.

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 sensores are attached on top of this unit.

Rode VideoMic Pro Microphone The Microfon 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. We use the Jack audio server[2] to capture speech and pass it either directly to the system or a noise filter. For filtering, the Quantile-based noise estimator [SFB00] is used. This audio setup and speech filtering was recommended by and tested in cooperation with team Golem from the UNAM, Mexico.

Canon PowerShot SX100 IS In earlier experiments we have found out that because of their low resolution the color images provided by the Kinect are not suitable for reliable object recognition. Thus, this off-the-shelf digital camera is mounted on top of the pan-tilt unit to capture high resolution (8 megapixels) images for object recognition.

Microsoft Kinect The Microsoft Kinect ist attached to the pan-tilt unit and provides depth and color images of size 640x480 pixels at 30Hz. The depth sensor operation range lies between 0.8 m and 3.5 m at a horizontal field of view of 58° and a vertical field of view of 45°. It also has two microphones and a tilt motor for sensor adjustment. However, the robot is equipped with another microphone for speech recognition. Its tilt motor is not used because of the pan-tilt unit that the Kinect is mounted on.

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.
(New in 2014) The end effector is a custom setup and consists of 4 Festo Finray-fingers. It is safe for interaction with humans.

4 Technology and Scientific Contribution

4.1 System Core Architecture

(New in 2014) In the last years we used a hybrid software architecture that consisted of a self developed architecture [TSL+11] and ROS[3]. For the first time this year, we completely migrated all of our software to ROS to benefit from a homegeneous architecture.

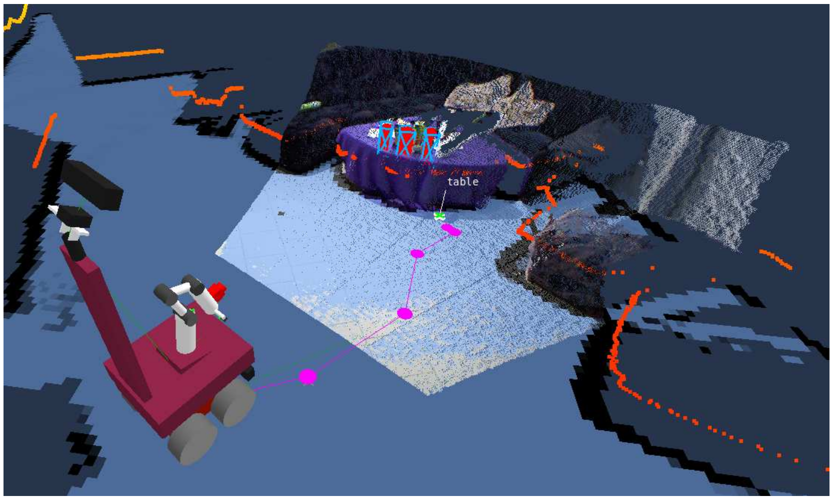

Fig. 2. Visualization of RGBD and laser sensor data. Further, a planned path to the point of interest “table” as well as grippable objects on the table are visualized.

4.2 Graphical Interface

(New in 2014) 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 with included the rviz plugin. A visualization example is given in Fig. 2.

4.3 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.

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

---

[2] Jack audio server: `http://jackaudio.org/`

[3] Documentation of ROS `http://www.ros.org/`

4.4 Navigation in Dynamic Environments

In real-life situations, the approach described above 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 [Zel88,Zel91], 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 efficently navigate in highly dynamic environments.

4.5 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. On the software side, we decided to use two open source libraries: pocketsphinx[4] for speech recognition and festival[5] for speech synthesis.

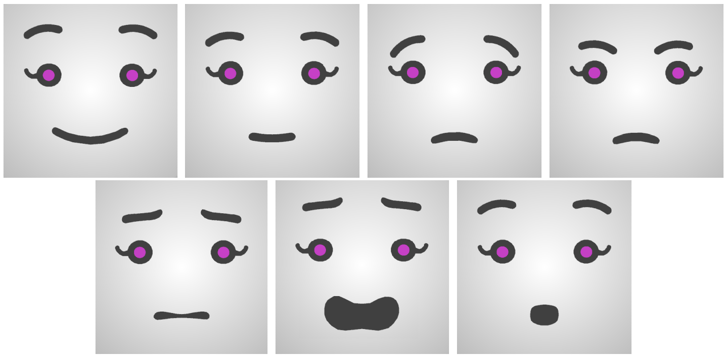

Fig. 4. Animated face of our service robot Lisa. The depicted face expressions are (from left to right): happy, neutral, sad, angry, disgusted, frightened, and surprised.

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[6] for visualisation. Furthermore, the face is modelled with Blender[7] and exported via the Ogre Mesh Exporter[8] for the use with Ogre.

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[9] so it is possible to apply it for any robot.
(New in 2014) Furthermore, we conducted a broad user study to test how people perceive the shown emotions. The results can be found in [SGGP13].

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

---

[4] Speech recognition system pocketsphinx `http://www.speech.cs.cmu.edu/pocketsphinx/`

[5] Speech synthesis system festival `http://www.cstr.ed.ac.uk/projects/festival/`

[6] Open-source graphics rendering engine Ogre3D `http://www.ogre3d.org/`

[7] Free open source 3D content creation suite Blender `http://www.blender.org/`

[8] Ogre Mesh Exporter `http://www.ogre3d.org/tikiwiki/Blender+Exporter`

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

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.

4.6 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.

4.7 Object and Face Recognition

The object recognition algorithm we use is based on Speeded Up Robust Features (SURF) [BTVG06], 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 [DTPG10] and shown as suitable for fast training and robust object recognition.

(New in 2014) With this object recognition approach we won the Technical Challenge 2012. Our object recognition is available as open source software in ROS[10]. A detailed description of the approach is provided on the website as well.

4.8 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.

4.9 People Tracking

For the perception of people we combine the three methodes described in 4.8. 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.

---

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

4.10 Object manipulation

Our system detects planes in the acquired 3D point cloud using RANSAC. 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. Depending on the position of the object, camera and manipulation device are selected. These are either the bottom camera with the 2 DOF gripper or the Kinect RGB-camera combined with the 6 DOF robotic arm. Additionally, a laser range finder on top of the 6 DOF arm help for precise gripping.

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 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 [CCL10].

5 Conclusion

In this paper, we have given an overview of the approach used by team homer@UniKoblenz for the RoboCup@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. Furthermore, 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. Based on the existing system from last year's competition, effort was put into enhancing the detection of people and the human-robot interaction abilities of our robot. 3D scans of the environment are aquired using a RGB-D sensor.

References

BTVG06. Herbert Bay, Tinne Tuytelaars, and Luc Van Gool. SURF: Speeded up robust features. *ECCV*, pages 404–417, 2006. 6

CCL10. Benjamin Cohen, Sachin Chitta, and Maxim Likhachev. Search-based planning for manipulation with motion primitives. Anchorage, Alaska, 2010. 7

DTPG10. Peter Decker, Susanne Thierfelder, Dietrich Paulus, and Marcin 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*, volume 1, pages 181–184, Moscow, 12 2010. Springer MAIK Nauka/Interperiodica. 6

SFB00. V. Stahl, A. Fischer, and R. Bippus. Quantile based noise estimation for spectral subtraction and wiener filtering. In *Acoustics, Speech, and Signal Processing, 2000. ICASSP'00. Proceedings. 2000 IEEE International Conference on*, volume 3, pages 1875–1878. IEEE, 2000. 3

SGGP13. Viktor Seib, Julian Giesen, Dominik Grüntjens, and Dietrich Paulus. Enhancing human-robot interaction by a robot face with facial expressions and synchronized lip movements. In Vaclav Skala, editor, *21st International Conference in Central Europe on Computer Graphics, Visualization and Computer Vision*, 2013. 5

TSL$^+$11. 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

Zel88. Alexander Zelinsky. Robot navigation with learning. *Australian Computer Journal*, 20(2):85–93, 5 1988. 5

Zel91. Alexander Zelinsky. *Environment Exploration and Path Planning Algorithms for a Mobile Robot using Sonar.* PhD thesis, Wollongong University, Australia, 1991. 5

Document date: 2014-02-08