RoboCup 2009 - homer@UniKoblenz (Germany)

David Gossow, Marc Arends, Susanne Thierfelder

Active Vision Group
University of Koblenz-Landau
Universitätsstr. 1
56070 Koblenz, Germany
dgossow@uni-koblenz.de
http://www.uni-koblenz.de/~agas/robbie12

Abstract. This paper gives a description of the robot hardware and software used by team homer@UniKoblenz, which participates in RoboCup@home 2009. A Special Focus is put on novel scientific achievements and newly developed features with respect to last year's competition.

Introduction

The *homer@UniKoblenz* is a team of researchers and students from the University of Koblenz-Landau, Germany. Together with its sister team *resko@UniKoblenz*, team *homer@UniKoblenz* will participate in the RoboCup@home league. Our mobile system called "Robbie 12" was developed during the last four years. Six practical courses were held in which the robot hardware was assembled and the software developed. The robot was first used to guide persons to various places within the university, then as a transport vehicle. Finally, it started participating in the RoboCup Rescue league and, since RoboCup German Open 2008, in the @home league.

1 About our team

1.1 Participation in two leagues

Before "Robbie X", our project was competing in the Rescue league. Then we decided to enhance our software platform to support a wider range of tasks so we could participate in the @home league as well.

Our experience at the RoboCup German Open and World Cup in 2008 proved this concept to be successfull, since the new functionality could be integrated into our architecture. Thus, we decided to continue developing one software for parallel participation in both leagues and hope that others will follow.

1.2 Team members and their contributions

- Members of team *homer@UniKoblenz*
  - David Gossow: team leader
  - Marc Arends: head of project team home, person tracking
  - Sönke Greve: hardware assistant
  - Viktor Seib: open contest, second robot
  - Susanne Thierfelder: public relations, face and object recognition
  - Christian Winkens: object detection
- The homer team is supported by the team *resko@UniKoblenz* with the members:
  - Johannes Pellenz: team leader, scientific advisor
  - Peter Decker: scientific advisor
  - Bernhard Reinert: head of project team rescue
  - David Beckmann: infrastructure, thermal camera

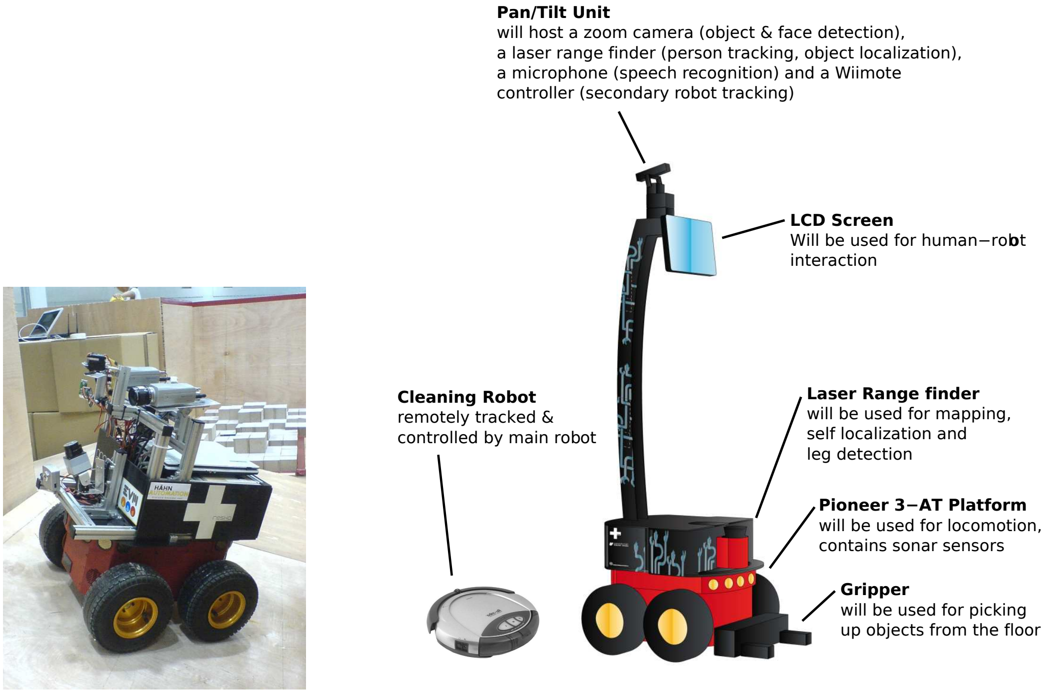

Fig. 1. Left: Our Pioneer 3 AT robot "Robbie X" used for the 2008 Rescue and @home competition, Right: Mockup of the modified Robot platform which will be used for the @home competition in 2009.

- Denis Dillenberger: stepfield detection
- Christian Fuchs: quality management, GUI
- Susanne Maur: scan matching
- Kevin Read: technical chief designer, mapping

Present at the RoboCup competition in Graz will be David Gossow, Marc Arends, Sönke Greve, Viktor Seib, Susanne Thierfelder and Christian Winkens. The @home team is supported by the Centre of Excellence of the Chamber of Crafts in Koblenz[1] (see section 2.1).

1.3 Focus of research

Our interests in research are grid-based 2d mapping, localization and navigation, system architecture for automobile systems, real-time visualization of sensor data and system states, laser-based person tracking, object and face recognition and object retrievement using a manipulator.

2 Hardware

2.1 Robot platforms

Pioneer3-AT As platform an ActivRobots Pioneer3-AT [2] is used. It provides sonar and odometry sensors as described below. It is equipped with four air-filled tires having a diameter of 21.5cm which can be controlled individually, allowing the robot to turn on the spot while maintaining a high degree of stability. **(New in 2009)** Attached to the platform is a 2-DOF gripper, which will be used to pick up objects from the floor.

[1] `http://www.hwk-kompetenzzentrum.de`
[2] `http://www.activerobots.com`

Home Robot Framework (New in 2009) On top of the P3-AT, we will install a prototype framework, which was designed and built by the Centre of Excellence of the Chamber of Crafts in Koblenz and will carry additional sensors and a notebook running the control software. It is designed in a way that local crafts enterprises can manufacture it using modern industrial materials like neopren and different composites.

Cleaning Robot (New in 2009) A second robot is used during the Open Challenge. We use an iRobot Roomba SE, which is a simple cleaning robot. It is equipped with infrared sensors, an infrared interface and a touch sensor. The main robot tracks its position using a Wiimote controller and controls it via WLAN.

2.2 Sensors and additional Hardware

Notebook The software runs on a Nexoc Osiris Notebook equipped with a 2,4 Ghz Intel Processor and 2 GB of RAM using Ubuntu Linux as operating system.

Odometry sensors The Pioneer robot has a built in system to report the robot pose, based on the readings from the wheel encoders. We use this data as a rough approximation of the relative robot motion between two SLAM iterations, which is then refined by laser scan matching and a particle filter.

Sonar sensors Our Pioneer 3 AT robot platform has two sonar rings (one scanning the front and one scanning the back) with eight sonar sensors each. Totally they cover 360 degrees, but with larger gaps between the sensors. The sonar sensors are used for collision avoidance and the data is displayed in the GUI. The sensors have a low resolution but they are used during the mapping an localization process to avoid transparent obstacles which cannot be seen by the LRF.

SICK S300 laser range finder (New in 2009) The SICK S300, which is mounted at the bottom and generates 270 degree scans. It has an adjustable angular resolution, while its maximal measured distance is 30m. It is used for mapping, localization and people tracking.

Hokuyo URG04-LX laser range finder The Hokuyo laser range finder generates 240 degree scans that measure the distance of the closest objects. The angular resolution of the laser range finder is 0.36° and the maximal measured distance is 5.6m. **(New in 2009)** It is mounted on the pan-tilt unit next to the zoom camera used for the object and face recognition as it is mounted and for people trakcing combined with the second LRF.

Pan-Tilt Unit (New in 2009) The DirectedPerception PTU-D46 is mounted on the top of the robot's neck. It is able to rotate 159 degrees in each direction and to tilt from $+31°$ to $-47°$ out of a horizontal position. Sensors are attached on top of this unit.

Zoom Camera (New in 2009) On top of the pan-tilt unit a Sony DFW-VL500 zoom camera is attached. This camera is able to zoom up to 12x magnification and provides a maximum of 30 color images per second.

Wiimote (New in 2009) A Nintendo Wiimote is mounted on the pan-tilt unit. It provides acceleration and infrared sensors and a bluetooth interface for communication. The Wiimote is used to to track a second robot.

3 Technology and Scientific Contribution

3.1 System Core Architecture

For participating in two different leagues, an architecture with a changeable configuration is needed. The heart of the software is a generic core, which is independent from hardware and purpose of the system. The core's main task is the exchange of messages in the system.

There are modules, each running as a single thread, which can be connected to the core and subscribe to messages. To keep these modules independent from each other the communication between them requires the core as a node. This means a message is sent to the core by a module and the core forwards the message to all the modules, which suscribed to this type of message.

Modules act as glue code, linking so-called workers and devices to the core system (see Fig. 2). They are supposed to handle messages and distribute data, providing little additional functionality. A Worker is a small set of program code (usually an object class) which mainly provides computing functionality, for example implementing a certain algorithm. In contrast, devices mainly provide access to a hardware components.

The system is highly configurable via a central registry stored as XML file. The registry contains various profiles, each one specifying its own set of modules to be loaded and configuration parameters to use.

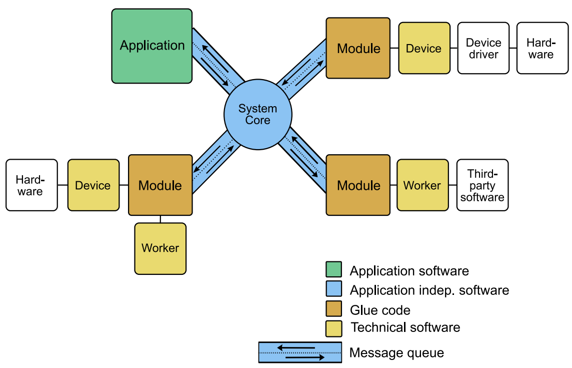

Fig. 2. The Architecture of the Robbie system.

3.2 Graphical Interface

Included in the framework is a GUI that can be run directly on the robot or on a computer connected via WLAN (see Fig. 3). The user interface is realized using Qt4 and OpenGL. This feature has shaped up as a very important tool for understanding and improving the complex algorithms needed for a fully autonomous robot.

The principal screen shows displays for various video streams (e.g. the ones provided by the cameras), a 3-dimensional view of the current sensor measurements and the generated map. In addition, various dialogues for starting the different games, monitoring the module activity and cpu usage, calibrating sensors, creating data sets for object and face detection, map editing, navigation and visualizing sensor data can be accessed.

3.3 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 a map of its environment in real time during normal operation. The map is

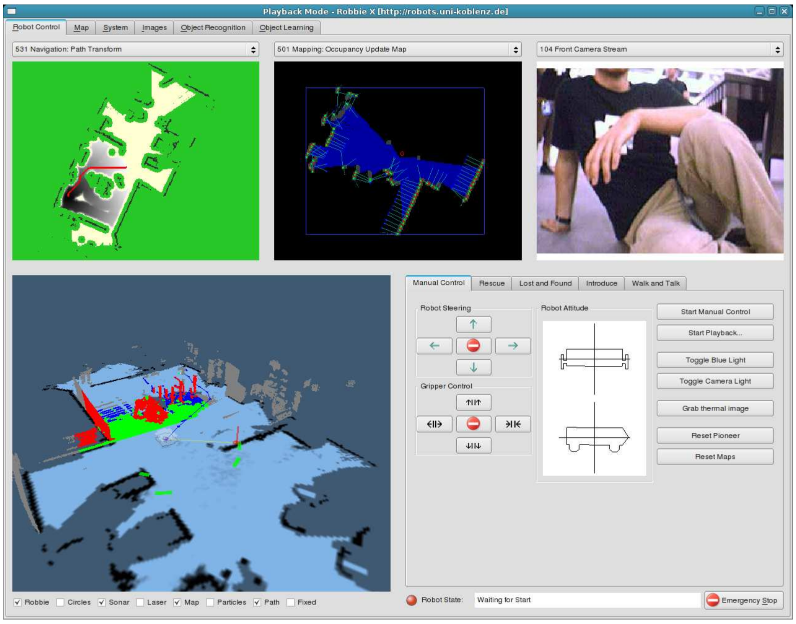

Fig. 3. The user interface of the operator station showing various visualizations of sensor data and system states. The application can be used for real-time surveillance of the robot as well as for playing back recorded sensor log files. In the @home league, this is mainly used by the developers for testing and evaluation.

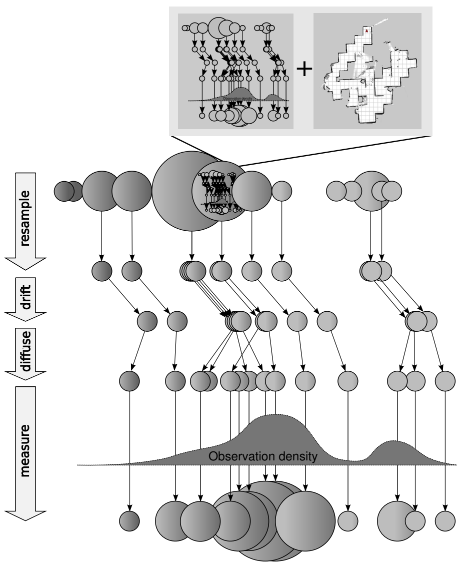

Fig. 4. Illustration of the Hyper Particle Filter proposed in [PP09].

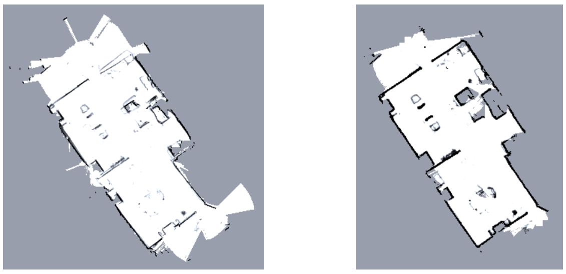

Fig. 5. Real-time maps of the @home arena. Left: Generated at RoboCup 2008, Right: Generated using the same sensor data with current algorithm. A significant improvement in geometric accuracy and visibility of obstacles has been achieved.

based on laser scans, odometry information and (optionally) measurements of the sonar sensors. Fig. 5 shows examples of such maps.

The map is stored in two planes. One plane counts how often a cell was surpassed by a laser beam, while the second plane holds information about how often the laser beam was reflected at this position. The occupancy probability for a cell is then calculated as the quotient of the two planes.

(New in 2009) To solve the SLAM problem, a novel approach coined Hyper Particle Filter (HPF) is used [PP09] (see Fig. 4). It is based on the concept of particle filter-based mapping [IB98,Pel08], in which the robot keeps a fixed number (in our case 1500) of different hypothesis' regarding its position and orientation in space.

In our approach, there are many particle filters running in parallel, each one with its own map and distribution of particles. For each particle filter, some measurements are randomly ignored, so that the results vary. The maps generated by each particle are weighted according to their quality, which is measured by its contrast.

3.4 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 enough information.

Thus, our navigation system, which is based on Zelinsky's path transform (see [Zel88,Zel91]), always merges the current laser range scan as unsurpassable object into the ocupancy 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 dynamical enviroments.

3.5 Autonomous Exploration

Several tasks in the @home league, like Lost And Found, require the robot to autonomously explore its environment.

For this purpose, we are using a novel exploration algorithm combining Yamauchi's frontier based exploration [Yam97] with Zelinsky's path transform. The path transform is extended in a way that not the cost of a path to a certain target cell is calculated, but instead the cost of a path that leads to a close frontier (see fig. 6). More details can be found in [WP07].

3.6 Human-Robot Interface

The robot is equipped with speakers and a microphone, which enables it to communicate via a speech interface. In addition, it has a small screen that will be used to display facial expressions and

Fig. 6. Illustration of the exploration transform algorithm described in [WP07]

state information. On the software side, we decided to use two Open Source libraries: *pocketsphinx*[3] for speech recognition and *espeak*[4] for speech output.

3.7 Object and Face Recognition

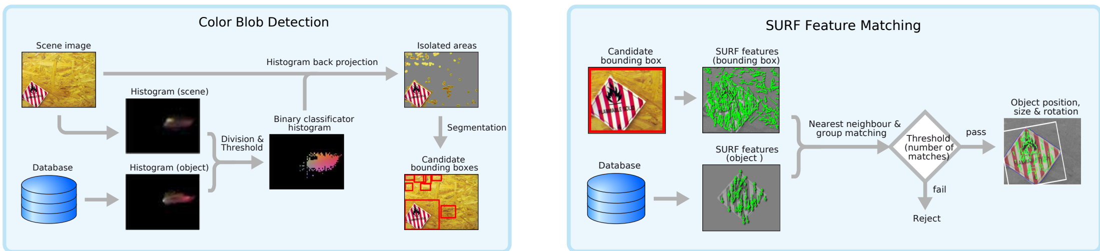

Fig. 7. Diagram showing the color- and feature-based detection steps we use to detect objects and faces.

The object recognition in Robbie is based on color and features by employing histogram back projection [Pau01] and Speeded Up Robust Features (SURF) [BTVG06]. We use the YUV color space, as it separates luminance and color information and is provided directly by our cameras.

The object recognition process is divided into five steps:

1. Calculate the histogram back projection to find regions of interest by color
2. **(New in 2009)** Enlarge regions of interest using the pan-tilt-zoom camera system
3. Calculate SURF features for these regions
4. Match the obtained features with the object database
5. Estimate the position of the objects by using the laser sensor and the absolute robot pose provided by the mapping module

For matching feature vectors, we use the nearest-neighbour-ratio of the euclidian distance as well as a variance-based consensus search by feature position, orientation and scale.

(New in 2009) For face recognition we use the results of the laser-based person tracking (section 3.8) as additional criteria.

3.8 Person Tracking

(New in 2009) The people detection and tracking is based on data provided by both LRFs. As the upper laser scans for the upper body of a human, the lower laser is scanning for the legs. Legs

[3] `http://www.speech.cs.cmu.edu/pocketsphinx/`
[4] `http://espeak.sourceforge.net/`

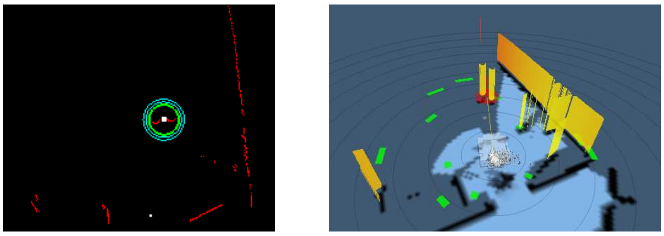

Fig. 8. Left: result of arc segment search in two-dimensional laser range data, Right: GUI visualization during follow mode. The red circle indicates the position of the tracked person.

are detected by looking for pairs of arc segments in the range data as in [XCRN05]. Then, the presence of the upper body part in the measurements of the upper laser scanner is verified. To increase accuracy, a Kalman filter [WB95] is used to predict the movement of tracked persons.

4 Conclusion

In this paper, we gave 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. Well-known techniques like particle filter- and grip map-based SLAM and path planning have been extended to achieve stable mapping, navigation and exploration in dynamic environments. Robust object and person detection and tracking is achieved by actively controlling laser range finders and a camera and by combining their measurements. All software components have been successfully intergrated into one system using a flexible module/message-based software architecture.

References

[BTVG06] Herbert Bay, Tinne Tuytelaars, and Luc Van Gool. Surf: Speeded up robust features. *ECCV*, 2006.

[IB98] Michael Isard and Andrew Blake. Condensation - conditional density propagation for visual tracking. *International Journal for Computer Vision*, 29(1):5–28, 1998.

[Pau01] Dietrich Paulus. *Aktives Bildverstehen*. Der Andere Verlag, Osnabrück, 2001.

[Pel08] Johannes Pellenz. Mapping and map scoring at the robocuprescue competition. In *Quantitative Performance Evaluation of Navigation Solutions for Mobile Robots (RSS 08, Workshop CD)*, 2008.

[PP09] Johannes Pellenz and Dietrich Paulus. Stable mapping using a hyper particle filter. *RoboCup Symposium 2009*, 2009. (submitted).

[WB95] Greg Welch and Gary Bishop. An introduction to the kalman filter. Technical report, Chapel Hill, NC, USA, 1995.

[WP07] Stephan Wirth and Johannes Pellenz. Exploration transform: A stable exploring algorithm for robots in rescue environments. *Workshop on Safety, Security, and Rescue Robotics, http://sied.dis.uniroma1.it/ssrr07/*, pages 1–5, 9 2007.

[XCRN05] M. Xavier, J.and Pacheco, D. Castro, A. Ruano, and U. Nunes. Fast line, arc/circle and leg detection from laser scan data in a player driver. In *Proceedings of the 2005 IEEE International Conference on Robotics and Automation (ICRA)*, 2005.

[Yam97] B. Yamauchi. A frontier-based approach for autonomous exploration. *1997 IEEE International Symposium on Computational Intelligence in Robotics and Automation*, page 146 ff., 1997.

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

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

Document date: 2009-02-02