About us

The Institute for Artificial Intelligence (IAI) at the University of Bremen, Germany, headed by Prof. Michael Beetz, investigates methods for cognition-enabled robot control. The research is at the intersection of robotics and Artificial Intelligence (AI) and includes methods for intelligent perception, dexterous object manipulation, plan-based robot control, and knowledge representation for robots.
Robots performing complex tasks in open domains, such as assisting humans in a household or collaboratively assembling products in a factory, need to have cognitive capabilities for interpreting their sensor data, understanding scenes, selecting and parameterizing their actions, recognizing and handling failures and interacting with humans. In our research, we are developing solutions for these kinds of issues and implement and test them on the robots in our laboratory. A particular focus of the group is on the integration of individual methods into complete cognition-enabled robot control systems and the release of the developed software as open-source libraries.
More information about the institute, the team and our research projects can be found on our website: http://ai.uni-bremen.de/
Two software libraries that origin from our laboratory and that are now used and supported by a larger user community are the KnowRob system for robot knowledge processing and the CRAM framework for plan-based robot control.

KnowRob – Robot Knowledge Processing KnowRob is a knowledge processing system that combines knowledge representation and reasoning methods with techniques for acquiring knowledge from different sources and for grounding the knowledge in a physical system. It provides robots with knowledge to be used in their tasks, for example action descriptions, object models, environment maps, and models of the robot's hardware and capabilities. The knowledge base is complemented with reasoning methods and techniques for grounding abstract, high-level information about actions and objects in the perceived sensor data.

KnowRob became the main knowledge base in the ROS ecosystem and is actively being used in different academic and industrial research labs around the world. Several European research projects use the system for a wide range of applications, from understanding instructions from the Web (RoboHow), describing multi-robot search-and-rescue tasks (SHERPA), assisting elderly people in their homes (SRS) to industrial assembly tasks (SMERobotics).
KnowRob is an open-source project hosted at GitHub[1] that also provides extensive documentation on its website – from getting-started guides to tutorials for advanced topics in robot knowledge representation[2].

[1] http://github.com/knowrob
[2] http://www.knowrob.org

CRAM – Robot Plans CRAM is a high-level system for designing and performing abstract robot plans to define intelligent robot behaviour. It consists of a library of generic, robot platform independent plans, elaborate reasoning mechanisms for detecting and repairing plan failures, as well as interface modules for executing these plans on real robot hardware. It supplies robots with concurrent, reactive task execution capabilities and makes use of knowledge processing backends, such as KnowRob, for information retrieval.

CRAM builds ontop of the ROS ecosystem and is actively developed as an open-source project on GitHub[3]. It is the basis for high-level robot control in many parts of the world, especially in several European research projects covering applications from geometrically abstract object manipulation (RoboHow), multi-robot task coordination and execution (SHERPA), experience based task parameterization retrieval (RoboEarth), and safe human robot interaction (SAPHARI).
Further information, as well as documentation and application use-cases can be found at the CRAM website[4].

In our group, we have a very strong focus on open source software and active maintenance and integration of projects. The systems we develop are available under BSD license, and partly (L)GPL.

[3] http://github.com/cram-code
[4] http://www.cram-code.org/

Proposed Topics for Summer of Code-Projects

In the following, we list three proposals for Summer of Code topics that contribute to the two aforementioned open-source projects CRAM and KnowRob.

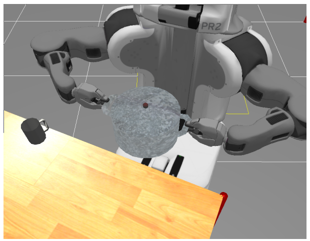
Topic 1: CRAM – Virtual Robot Scenarios in Gazebo

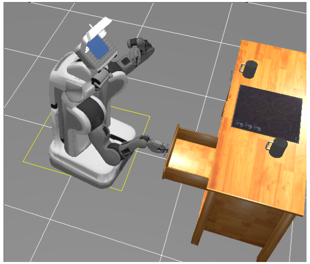
Main Objective The development of scenarios and tasks for human-sized robots. This is done using ROS, the Gazebo robot simulator, and CRAM. The scenarios to develop range from setups for comparatively simple tasks, such as pick and place, to more sophisticated workspaces requiring interaction with the environment, such as opening and closing drawers and doors.
This involves designing virtual environments for Gazebo and/or writing robot plans in Lisp using the CRAM high-level language, sending commands to virtual PR2 or REEM(-C) robots in Gazebo, and manipulating the artificial environment in there. The connection to an elaborate high-level system holds a lot of interesting opportunities.
The produced code will, when working in a simulated environment, be run on the real robot in our laboratory and become part of the high-level behaviour library for the connected robots.

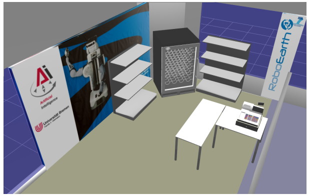
Task Difficulty The difficulty depends on the environment setup and task chosen. The design of pick and place scenarios in a simple environment work towards ensuring the system's robustness, while more elaborate interaction tasks require more situational intelligence in the robot plan.

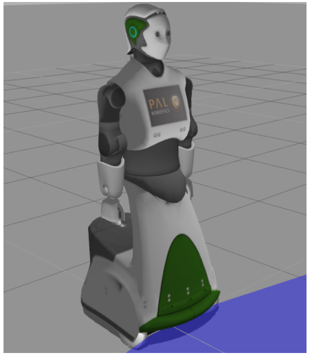
Requirements At least basic understanding in functional programming is advisable (ideally Lisp), basic knowledge in ROS helps, and ideally already having experience in simulation environments.

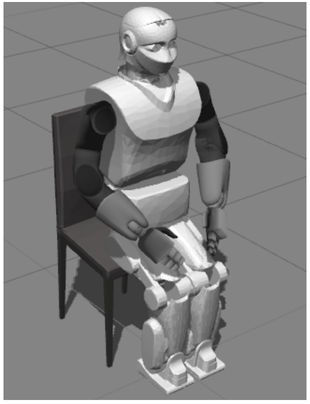
Expected Results We expect operational and robust contributions to the software library that can be used as part of a robot's control program.



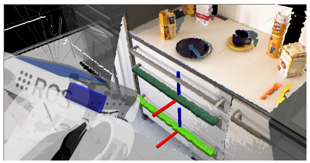








Topic 2: CRAM – Symbolic Reasoning Tools with Bullet



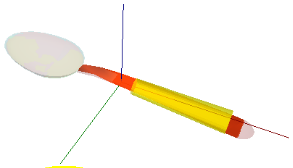
Main Objective Mapping the environment to the internal belief state representation and keeping track of changes in the environment to keep the belief state up to date based on manipulation and interaction tasks performed by the robot.

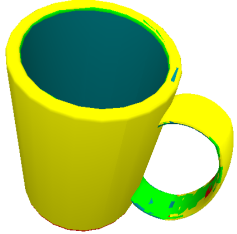
Task Difficulty Relatively simple, when making the existing track changes more robust, and more challenging when introducing new change tracking (like noting the angle of open doors after opening them and storing it in the belief state).

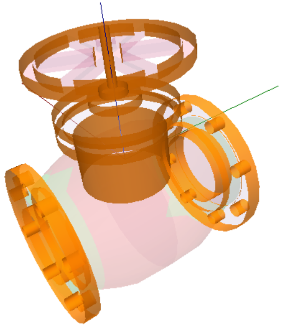
Requirements At least basic understanding in functional programming is advisable (ideally Lisp), basic knowledge in ROS helps. Also a good understanding of geometric shapes and coordinate transformations helps.

Expected Results We expect operational and robust contributions to the software library that can be used as part of a robot's control program.

Topic 3: KnowRob – Reasoning about 3D CAD models of objects







Main Objective Whenever robots interact with objects, they need information about their geometry. While the overall shape is sufficient for picking up an object, more information about its composition from parts and their meaning (e.g. a handle or a container) is needed when using objects as tools in more complex activities. We have developed an initial version of a program library that is able to analyze CAD models of objects, as they can be found in large databases on the Web, and to automatically identify such functional components. As part of this project, that software library is to be extended in order to recognized additional geometric primitives (e.g. torus, box) and to improve the segmentation and make it more robust. Alternatively, the focus could be more on improving the analysis infrastructure and integrating it more closely with the robot's control program.

Task Difficulty The difficulty depends on the exact topic and ranges from rather simple tasks at the infrastructure level to interesting research problems for the geometric analysis.

Requirements The algorithms for segmenting 3D meshes require solid understanding of 3D geometry. Good programming skills in Java are needed for implementing the algorithms and for evaluating their results. Depending on the exact topic, knowledge of machine learning methods or the ability to learn about them could be helpful.

Expected Results We expect operational and robust contributions to the software library that can be used as part of a robot's control program.