Knowledge Processing for Cognitive Robots

Moritz Tenorth, Dominik Jain and Michael Beetz

Knowledge processing methods are an important resource for robots that perform challenging tasks in complex, dynamic environments. When applied to robot control, such methods allow to write more general and flexible control programs and enable reasoning about the robot's observations, the actions involved in a task, action parameters and the reasons why an action was performed. However, the application of knowledge representation and reasoning techniques to autonomous robots creates several hard research challenges. In this article, we discuss some of these challenges and our approaches to solving them.

1 Introduction

Arguably, AI-based robotics started with the Shakey project at SRI as early as 1966 and the realization of an autonomous robot with human problem-solving skills was by many considered to be the holy grail of Artificial Intelligence [20]. The research on AI methods for autonomous robot control was framed within the so-called sense-plan-act architecture in which a sensing module was to map a sensed scene into a symbolic (first-order logic) representation of the scene, the "plan" module took the scene description, inferred a goal and generated a plan for achieving the respective goal, and the act module took the symbolic plan and translated it into control signals for the robot.

The research on Shakey immediately spawned seminal research work investigating the key components of the plan module that was considered to be *the* system component responsible for achieving "intelligent" problem-solving behavior. All of us have read the papers "Application of Intelligent Automata to Reconnaissance" [5], "STRIPS: A New Approach to the Application of Theorem Proving to Problem Solving" [9], "Learning and Executing Generalized Robot Plans" [10], "Application of Theorem Proving to Problem Solving" [12], and many others, which started up and dominated the research directions in AI planning and reasoning for decades. Only 15 years later the Shakey researchers became aware that the embedding of the AI methods into the perception and action mechanisms of the robot might be equally important for the research communities [19]. As a result, the Shakey researchers published edited versions of the original project proposals and progress reports as technical reports to make them accessible.

These events within the Shakey project were later paralleled by the development of the whole field of AI-based robotics. For a number of years Artificial Intelligence focussed on the investigation of suitable learning, reasoning, and planning methods that took complete symbolic representations for granted and produced symbolic representations of action specifications. This research direction, which failed to produce competently acting robots, was radically challenged by Brooks who proposed to achieve intelligent agency through control methods that do not rely on sophisticated representations and reasoning [4, 3]. More recently, the research field of developmental robotics has stressed that competent agency requires tightly integrated data structures and computational processes, which are best realized through the co-development of data structures and processes.

Our own research approach is to take state-of-the-art and leading-edge *knowledge processing methods*, i.e. methods for knowledge acquisition, representation, reasoning and learning, and make them work for autonomous robots by grounding the AI methods into the robot's perception and action mechanisms and the data structures of the robot and by developing "satisficing" methods [22] that work under reasonable assumptions about the robot's knowledge, the environment, and the tasks [1, 2, 23].

With autonomous robots becoming skilled enough to perform rather complex everyday manipulation tasks, knowledge processing is becoming more and more important, since those tasks require much knowledge of different kinds to be performed competently: The robot needs to infer the required actions from heavily underspecified commands. It needs to know which objects are involved, where to stand to pick them up, which grasp type and grasp force to use, and how to approach the objects. Moreover, different object states (like a cup being clean or dirty, empty or filled with coffee) influence how they are to be handled. Supplying the knowledge and reasoning mechanisms to infer the correct decision creates several research challenges:

- How to *ground symbolic representations into sensor data* and actions to be performed? How to continually update the knowledge to keep it consistent with the state of the world?
- How to represent *uncertain knowledge*, and how to perform sound inference on it?
- How to *acquire all the knowledge* that is necessary to competently perform everyday tasks?
- How to perform *adapt to changing conditions and learn new tasks*?
- How to find *suitable representations* that are expressive enough, but still allows for fast reasoning?
- How to attain *action-awareness*, and how to *predict the effects* of actions?
- How to *interact with humans*, communicate with them and interpret their actions?

In the following sections, we will discuss these challenges which we encountered while developing knowledge processing systems for autonomous household robots. We will introduce the problems and outline how we addressed them in our work.

2 The KnowRob System

A knowledge processing system serves as a common semantic framework for integrating information from different sources. Our system is built around the KnowRob framework that was introduced in [23] and has since been extended. KnowRob is implemented in Prolog and uses OWL, the Web Ontology Language, for representing knowledge. Figure 1 gives an overview of the system structure. OWL is a compromise between expressiveness and reasoning capabilities. Translating sensor data into this format is rather simple, the conversion of knowledge from more expressive sources requires specialized translation procedures.

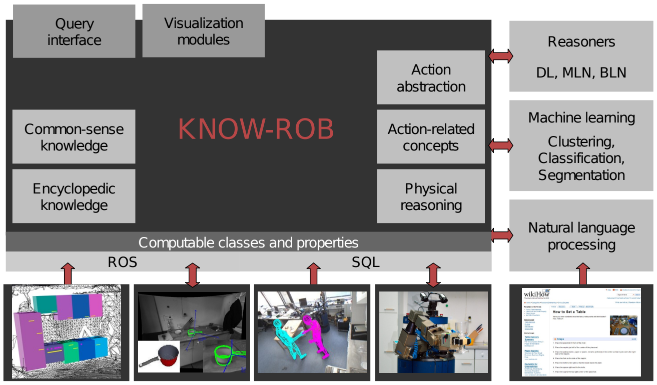


Figure 1: Overview of the KnowRob system. The knowledge base is tightly integrated with the robot's perception and planning modules and provides query and visualization interface, methods for loading external information, and several inference techniques.

KnowRob provides an extensible knowledge-based framework that allows to integrate different kinds of knowledge (static encyclopedic knowledge, common-sense knowledge, task descriptions, environment models, object information, observed actions, etc.) from different sources (manually axiomatized, derived from observations, or imported from the web). It supports different reasoning mechanisms (deterministic and probabilistic reasoning), clustering, classification and segmentation methods, and provides query interfaces as well as visualization tools.

As a knowledge processing system designed for robots, KnowRob is working on-line during the robot's operation, but it can base its decisions on all information it has acquired so far. For instance, it can learn models of human actions from motion-capture data that has been computed off-line and use the inferred results during on-line operation.

3 Research Challenges

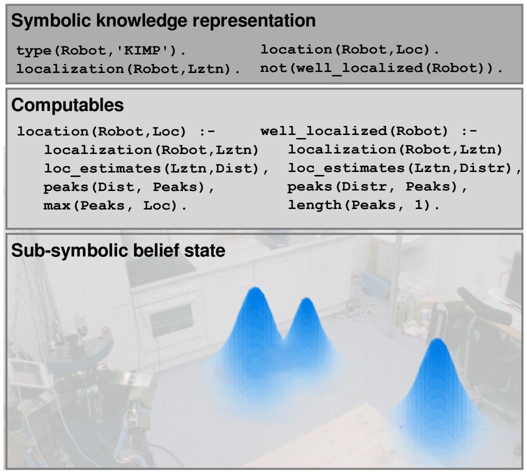
3.1 Symbol Grounding

When performing abstract symbolic reasoning about phenomena from the outside world, the robot needs to make sure the symbols are *grounded* [14]. That is, any symbol in the knowledge base needs to be related to the corresponding data structures in the robot's perception and control systems.

On the one hand, this means that the robot needs mechanisms to generate symbols out of its perception of the world, and to update its belief when the observations change. For example, the robot must be able to recognize objects and represent them internally as an instance of the respective object type. On the other hand, it also needs to link symbolic action descriptions to executable procedures with the same semantics as the action symbol.

A still open challenge is how to ensure consistency: Much of the information in the robot's knowledge base is generated from uncertain sensor data, and eliminating contradictions in this data using only deterministic representations is difficult. In the following sections, we present some methods to approach this problem: To compute data on demand, to store information only once, and to use probabilistic reasoning methods where conflicts are likely.

Our approach: Computables – Regard the World as a Virtual Knowledge Base We extend the classical first-order knowledge representation with *computable* predicates. Instead of being evaluated based on the axiomatized knowledge in the robot's knowledge base, they are computed by calling external methods. This allows to generate symbolic concepts out of observations or robot-internal data structures on demand during the reasoning process. In general, this method extends the robot's reasoning capabilities from manually stated symbolic knowledge to real world phenomena.



```
type(Robot,'KIMP').              location(Robot,Loc).
localization(Robot,Lztn).        not(well_localized(Robot)).
```

```
location(Robot,Loc) :-           well_localized(Robot) :-
   localization(Robot,Lztn)         localization(Robot,Lztn)
   loc_estimates(Lztn,Dist),        loc_estimates(Lztn,Distr),
   peaks(Dist, Peaks),              peaks(Distr, Peaks),
   max(Peaks, Loc).                 length(Peaks, 1).
```

Figure 2: This example illustrates how *computable* predicates ground symbolic knowledge (the location of a robot) in sub-symbolic data (a multi-hypothesis probability distribution).

Figure 2 illustrates the concept using the example of robot localization. On a sub-symbolic level, the robot estimates its position in the environment using a probabilistic, multi-hypothesis localization algorithm. In the example picture, there are three peaks in the probability distribution over robot positions, so the localization is rather uncertain. For performing abstract reasoning, the robot needs to determine e.g. the most likely pose or an estimation of the localization accuracy from this continuous data. This is done by small computational methods that are attached to the semantic properties. These methods, like *loc_estimates(L, D)*, perform the grounding by computing symbolic statements from sub-symbolic data. The example is simplified in that it does not take time into account; in the actual implementation, values like the pose of a robot or an object are represented as fluents; *location(R,L)* thus becomes *location(R,L,Time)*. Through the use of computable predicates, the symbolic information is always computed from the most current data and updated when the world changes. Computables are also used for generating object instances based on the current robot perception, or to compute qualitative spatial relations, like an object being on top of another one, based on the objects' positions.

While being a rather simple modification from a technical point of view, the use of computable predicates has several important implications: First, the resulting representations are inherently grounded. Second, computables help to ensure consistency with the outer world, since information is always generated from the latest observations, and inside the knowledge base, since information is usually stored only once. Other views on this information, like qualitative spatial relations, are computed from it on demand, which makes outdated information less likely. Third, the system does not discard information: If abstract descriptions are needed, they can be generated, but the more detailed representation is still in the background, like for instance the robot location probability distribution. In a more traditional approach, the knowledge would be described with fixed granularity and without access to the underlying data structures. Obviously, the implementations of the computable predicates have to make sure that the computed results are consistent with the robot's knowledge, for example to prevent the robot from being in multiple locations at once.

3.2 Uncertainty

Household robots do not operate in idealized artificial worlds that could reasonably be described in logical axioms. Real-world environments are highly dynamic, the robot being but one of several agents, and many aspects do not follow deterministic patterns but are, especially in the light of partial observability, more adequately represented as probabilistic dependencies. This is especially true for human preferences and behavioral habits, which must certainly be considered in a domestic robot's knowledge base if the robot is to adapt its own behavior to suit the needs at hand. Also for information obtained from noisy sensors, it is important that the uncertainty in these observations can be represented and, if several uncertain sources of information are combined, correctly propagated.

Moreover, robots will often be instructed to perform tasks that are not fully specified. A command pertaining to a fairly complex task will rarely include all the information that is necessary for the robot to derive a sequence of actions that will achieve the implied goal, for the goal itself may be subject to uncertainty and more than one state could be considered an appropriate solution. Robot plans for such complex tasks are, therefore, not completely specified – a problem that does not arise in industrial robots or in artificial planning domains. Instead, a robot needs to infer the course of action, the set of objects to consider, action parameters like the right positions, suitable trajectories, correct grasps, and many more — all of which may depend on the task context, individual preferences, past experience and many other parameters.

To address the issues named above, a robot needs reasoning techniques allowing it to infer information that is missing in commands it receives, to reason about potential effects of activities it observes and causes of situations it observes – or, more generally, to base its decisions and update its beliefs based on what can be considered to be most (likely to be) appropriate given its past experience.

In artificial intelligence, probabilistic graphical models provide a well-established formalism for the representation of uncertainty. In real-world environments, however, the set of entities we may need to reason about will vary widely; propositional models with a fixed set of random variables do not suffice. Furthermore, a direct coupling between the relational knowledge in our logical knowledge base and random variables in probabilistic models is highly desirable.

Our approach: Probabilistic First-Order Models. We use statistical relational models [11] to represent probabilistic knowledge. These models are first-order, abstracting away from concrete entities and representing instead general principles about objects having similar properties (cf. universal quantification). Viewed pragmatically, they essentially represent templates for the construction of graphical models: For any concrete set of objects we want to consider, the relational model generates, by repeatedly materializing its template structures, a concrete probabilistic model. This model is typically represented as a graphical model, which contains as random variables logical ground atoms (i.e. instances of predicates) that represent statements about the objects under consideration. In this way, statistical relational models can be viewed as a means of unifying statistical and relational knowledge. As concrete representation formalisms, we use Markov logic networks [21] and Bayesian logic networks [15].

An example application that combines the computables introduced in Section 3.1 with the probabilistic reasoning methods is shown in Figure 3. The task is to infer which objects need to be added to complete a partial table setup. To solve this task, the robot needs to detect the objects and load them into the knowledge representation using the computable predicates. Using probabilistic inference and models learned from observations of human meals and the objects involved, the robot can conclude what meal the table is most likely being set for, what utensils and food items are needed for that, and thus what it should add. Figure 3 shows the camera image of the table on the left side and the object instances in the knowledge base that were created from the detected items on the right side. The red objects on the table were recognized and are passed as evidence to the probabilistic inference engine. The results are visualized in the upper region, with the hue value corresponding to their likelihood (red objects are more likely than orange, green and blue). One can see that the system infers a plate to eat from and a cup to drink the detected coffee from as mandatory, and other items like a glass for the iced tea as very probably needed.

3.3 Knowledge Acquisition

Robots that are to act skillfully in human environments need an enormous amount of knowledge. A challenge that has long been neglected is how to acquire this knowledge. In small-scale laboratory settings, it is often still possible to hand-code the knowledge required to handle a few objects in a pre-defined way and to communicate with a human on a limited set of topics.

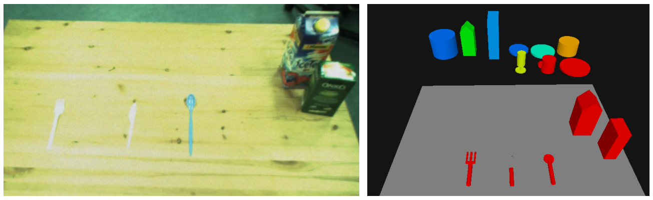


Figure 3: Example application: Inferring the objects that are missing given a partial table setup. Left: Camera image showing objects on the table. Right: Corresponding object instances in the knowledge base and objects that are inferred to be missing. The hue value corresponds to the probability.

This changes dramatically when the robot comes to a real environment and is to perform several complex tasks, acquire new tasks or new variants of known ones, manipulate new objects, and understand human commands in colloquial language.

For being accepted by humans as useful household companions, robots should also quickly adapt to people's habits and learn from few examples. It is thus important to re-use existing knowledge, ideally both skills and models another robot has already learned, and information that was originally intended to be used by humans.

Our approach: Import Knowledge from the WWW In our ongoing research, we are investigating how information from the Internet can be used to improve the robot performance. The World Wide Web is among the largest resources of knowledge of any kind: Web sites like ehow.com or wikihow.com contain step-by-step instructions for thousands of tasks. In addition, there are many sites with cooking recipes. Many of these descriptions cover not only the default version of a task, like baking normal brownies, but also provide instructions for variations like diabetic brownies, peppermint brownies, or gluten-free brownies.

More detailed information on how to manipulate objects can be obtained from video tutorials that show how exactly a cooking task is to be performed. Object models can be generated using image search engines like Google images or repositories of 3D object models like the Google 3D warehouse. These models help the robot ground the abstract descriptions into its perception, i.e. to recognize and manipulate previously unknown objects that appear in the instructions.

A challenge is to make use of this information: Instructions are written in ambiguous natural language and need a lot of common-sense knowledge to be understood correctly. We developed a system to translate the instructions from natural language to a representation in description logic [25], including semantic parsing, disambiguation, and the resolution of the corresponding concept in the ontology. Another problem is irrelevant or unrelated information. The system presented in [16] is able to retrieve 3D object models from the web, filter out irrelevant ones, and match the resulting model in the robot's environment.

Other important resources are public large-scale knowledge bases and the methods and resources developed as part of the Semantic Web initiative. The Cyc ontology [17] contains a huge amount of knowledge and seems to emerge as a quasi-standard in robot knowledge representation [23, 6]. The WordNet [8] lexical database helps robots to understand utterances in natural language. Common-sense knowledge is collected in the Open-Mind Indoor Common Sense (OMICS) database [13].

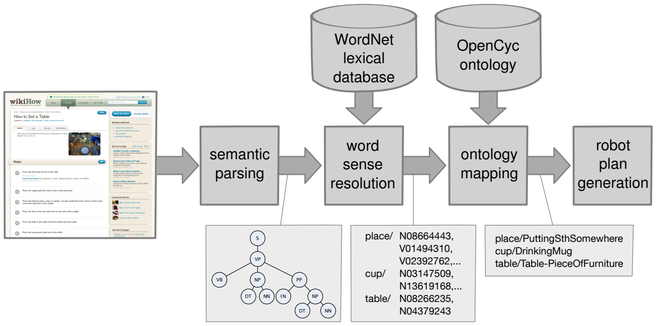


Figure 4: Procedure for importing task instructions from natural-language sources like web sites. The descriptions are parsed, and the word senses are resolved to ontological concepts.

3.4 Representation Language

In a robotic system, information has to be represented and processed at various levels of abstraction: From raw sensor measurements like the image of a camera or a distance measurement from a laser range finder, interpreted sensor data like clusters of point cloud data, recognized and localized objects, information about properties of object types, up to actions, action parameters, plans, and meta-knowledge about these plans, like problems that can occur during their execution. All these pieces of information are correlated and describe different aspects of objects and actions with different granularity. Since robots exist over time, they do not only need to describe the current state of the world, but also previously perceived world states, past intentions, actions that were performed, etc.

The task of the knowledge representation is to provide the means to describe this information on different abstraction levels and from different sources, assign meaning to the data, and allow to automatically combine it to perform useful inference. It further needs to ensure that the same word means the same thing in all components of the system so that e.g. the result of an action recognition system can be related to a similar plan in the robot's plan library or a command a human has given.

The choice of a knowledge representation formalism determines both what the robot can describe and what it can do with its knowledge. Davis et al. [7] give an overview over different approaches, their representational power, the primitives the representation is composed of, and the kinds of reasoning it supports.

Our approach: Description logic extended with Computable Predicates KnowRob is based on description logics as representation language, which is both light-weight, structured, and still expressive enough for most applications. The knowledge is stored in OWL, which became a common knowledge interchange format that is supported by many applications, loaded into Prolog, and can be accessed via Prolog predicates.

The methods described in Section 3.2 can be queried from within KnowRob and extend the system with probabilistic inference capabilities. Using the computables described in Section 3.1, the system accesses information from the outer world, and can also include specialized powerful inference mechanisms like clustering, classification or other kinds of computation into the reasoning process. Computables allow to perform complex inferences using fast procedural implementations, which can be important for robots interacting with the real world – where results need to be found in a short time. Memory modules for perception and a logged belief state of the control program provide the robot with the ability to reason about the past and to learn from experience.

3.5 Learning

Domestic robots should seek to adapt to the environments they operate in and especially to the humans within these environments. As a robot makes new observations pertaining to properties of the environment or human preferences and behaviors, it should attempt to incorporate these observations (or rather the abstract pieces of information that can be derived from them) into its knowledge base. Particularly aspects of the world that are subject to uncertainty and cannot be fully axiomatized call for learning methods in order to extract the latent patterns in the vast amounts of relational data that come in via the robot's perception system.

In particular, domestic robots should learn about the preferences and habits of the individuals they are to assist and serve. Robots should, for example, learn about the food and drinks people like to consume, the places at which they prefer to be seated, the utensils and objects they prefer to use for particular tasks, etc. Moreover, robots should learn about the environment, e.g. about the storage locations of particular types of objects and the roles of places, devices and immovable objects and in activities. By learning relations from observation, the robot can acquire much common-sense knowledge it needs for its tasks.

Our approach: Statistical Relational Learning The methods of statistical relational learning offer a sound way of combining relational knowledge representation, learning and reasoning within a single framework. By using a physically grounded perception system that abstracts away from low-level sensory data and represents the information as relational data, i.e. logical atoms with well-defined semantics, we obtain relational databases that may serve as the inputs to statistical relational learning problems.

3.6 Introspection and Prediction

Being able to predict which effects an action has, or how action parameters influence the result, is highly important for planning actions, for verifying if the chosen action parameters will have the desired effect, and for checking for side-effects of actions. In a quasi-artificial world in which all actions have well-defined prerequisites and effects, this is a simple task: If the world state matches the preconditions of an action, and if the robot performs this action, the resulting state can be described by the action's postconditions.

As usual, reality looks different: Actions can easily fail, small variations in the choice of parameters can determine the success, for example if a glass is securely grasped, falls out of the hand due to a too low grip force, or gets broken by the robot. Side-effects can be inherent to the actions or caused by failures, e.g. collisions with other objects.

Our approach: Prediction based on Physical Simulation and Semantic Models of the Robot and its Capabilities Prediction just based on logical inference would require an extreme amount of axiomatized knowledge including temporal and spatial reasoning, kinematics, detecting collisions etc. Instead, we are using a realistic physical simulation [18] that gets parametrized with the knowledge the robot has about its environment. A detailed semantic robot model describes its size, kinematics, dynamics, and capabilities of actors and sensors. Plans can be executed within this simulated environment, changes in the world are be logged with a God's eye view and translated into logical statements. Obviously, a simulation is only an abstracted model of reality and will therefore not always produce the exact result, but it is likely to be much better than what logical inference on a limited, axiomatized model will yield.

Based on the simulation result, the robot can answer queries regarding the outcome of an action, e.g. if the desired result has been obtained, or if unexpected events like collisions have occurred. Based on these results and data collected during actually performed actions, the robot can learn models of what it can do, how fast it can do something, or what can go wrong with which actions. It can also answer questions regarding why something was done, or search for solutions why something failed.

3.7 Interaction with Humans

Interacting with humans means to communicate, verbally and non-verbally. Verbal communication skills are important to receive commands, and to ask for more information or confirmation. Understanding natural language is thereby challenging, not only due to ambiguities, but also since humans are used to communicate with people with similar common-sense knowledge. Therefore, they convey much information with very few words that need to be interpreted in the right context.

Another aspect is non-verbal communication, that is both to recognize human actions and intentions and to show the own intentions. Performing actions similar to humans, for example using a similar sequence of actions, similar trajectories, or similar arm postures, can make them easier to understand. Though these challenges are closely related to perception and planning, the robot also needs knowledge to interpret observations of human actions and to parametrize the action execution.

Our approach: Verbal Communication and Knowledge-based Action Interpretation For understanding natural language, we have the techniques that are described in [25] in the context of importing task instructions from the WWW. Since the methods are fast, reliable, and able to understand reasonably complex instructions, they can also be used for (near) real-time communication.

We consider non-verbal communication, such as non-intrusive observation of human activities, similarly important. The Automated Models of Everyday Activities (AM-EvA, [2]) integrate techniques for human motion tracking, for learning motion primitives, for motion segmentation, and for abstracting from motions to actions and activities with statistical relational models describing action properties in the complete activity context. All these modules can be accessed from the robot's knowledge base to analyze observed activities from different viewpoints and at different granularities. Parts of the system have been applied, in conjunction with a transformational planning system, to the imitation of observed human manipulation activities [24]. By performing tasks similar to humans, they can better be understood by people.

4 Conclusions

On the one hand, knowledge processing is an essential resource for autonomous robots that perform complex tasks in dynamic environments. Robots need advanced reasoning capabilities to infer the control decisions required for competently performing complex tasks like everyday manipulation. Their knowledge processing system has to provide them with common-sense knowledge, with the ability to reason about observations of the environment, and with methods for learning and adapting over time. On the other hand, though knowledge representation and reasoning are well-established techniques in AI, their application to the problems in robotics is anything but trivial and poses several hard research challenges. Symbol grounding, reasoning about complex relations while taking uncertainty into account or learning in a complex environment are only some of the challenges. Issues like the acquisition of the large amount of required knowledge, the prediction of the outcome of complex actions, or the interaction with humans also need to be tackled.

We believe that robotics can provide both an interesting application and a set of challenging research problems to the area of knowledge representation and reasoning.

5 Acknowledgments

This work is supported in part within the DFG excellence initiative research cluster *Cognition for Technical Systems (CoTeSys)*, see also www.cotesys.org.

References

[1] Michael Beetz, Dominik Jain, Lorenz Mösenlechner, and Moritz Tenorth. Towards Performing Everyday Manipulation Activities. *Robotics and Autonomous Systems*, 2010. To appear.

[2] Michael Beetz, Moritz Tenorth, Dominik Jain, and Jan Bandouch. Towards Automated Models of Activities of Daily Life. *Technology and Disability*, 22, 2010.

[3] R. Brooks. Intelligence without reason. In *Proceedings of the Twelfth International Joint Conference on Artificial Intelligence*, pages 569–595, Sidney, Australia, 1991.

[4] R. Brooks. Intelligence without representation. *Artificial Intelligence*, 47:139–159, 1991.

[5] L.S. Coles, C.A. Rosen, B. Raphael, T.D. Garvey, R.O. Duda, and STANFORD RESEARCH INST MENLO PARK CA. Application of Intelligent Automata to Reconnaissance., 1969.

[6] Marios Daoutis, Silvia Coradeshi, and Amy Loutfi. Grounding commonsense knowledge in intelligent systems. *J. Ambient Intell. Smart Environ.*, 1(4):311–321, 2009.

[7] R. Davis, H. Shrobe, and P. Szolovits. What is a knowledge representation. *AI Magazine*, 14(1):17–33, 1993.

[8] C. Fellbaum. *WordNet: an electronic lexical database*. MIT Press USA, 1998.

[9] R. O. Fikes and N. J. Nilsson. STRIPS: A new approach to the application of theorem proving to problem solving. Technical Report 43r, AI Center, SRI International, 1971.

[10] R.E. Fikes, P.E. Hart, and N.J. Nilsson. Learning and executing generalized robot plans. *Artificial intelligence*, 3:251–288, 1972.

[11] Lise Getoor and Ben Taskar. *Introduction to Statistical Relational Learning (Adaptive Computation and Machine Learning)*. The MIT Press, 2007.

[12] C. Green. Application of theorem proving to problem solving, 1969.

[13] Rakesh Gupta and Mykel J. Kochenderfer. Common sense data acquisition for indoor mobile robots. In *Nineteenth National Conference on Artificial Intelligence (AAAI-04*, pages 605–610, 2004.

[14] S. Harnad. The symbol grounding problem. *Physica D*, 42:335–346, 1990.

[15] Dominik Jain, Stefan Waldherr, and Michael Beetz. Bayesian Logic Networks. Technical report, IAS Group, Fakultät für Informatik, Technische Universität München, 2009.

[16] Ulrich Klank, Muhammad Zeeshan Zia, and Michael Beetz. 3D Model Selection from an Internet Database for Robotic Vision. In *International Conference on Robotics and Automation (ICRA)*, 2009.

[17] C. Matuszek, J. Cabral, M. Witbrock, and J. DeOliveira. An introduction to the syntax and content of Cyc. *Proceedings of the 2006 AAAI Spring Symposium on Formalizing and Compiling Background Knowledge and Its Applications to Knowledge Representation and Question Answering*, pages 44–49, 2006.

[18] Lorenz Mösenlechner and Michael Beetz. Using physics- and sensor-based simulation for high-fidelity temporal projection of realistic robot behavior. In *19th International Conference on Automated Planning and Scheduling (ICAPS'09).*, 2009.

[19] N. J. Nilsson. Shakey the robot. Technical Report 323, AI Center, SRI International, 1984.

[20] N.J. Nilsson. Eye on the Prize. *AI Magazine*, 16(2):9, 1995.

[21] Matthew Richardson and Pedro Domingos. Markov Logic Networks. *Mach. Learn.*, 62(1-2):107–136, 2006.

[22] H. A. Simon. Models of bounded rationality. In *Vol II: Behavioural economics and business organization*. MIT press, 1982.

[23] Moritz Tenorth and Michael Beetz. KnowRob — Knowledge Processing for Autonomous Personal Robots. In *IEEE/RSJ International Conference on Intelligent RObots and Systems.*, 2009.

[24] Moritz Tenorth and Michael Beetz. Priming Transformational Planning with Observations of Human Activities. In *IEEE International Conference on Robotics and Automation (ICRA). Accepted for publication.*, 2010.

[25] Moritz Tenorth, Daniel Nyga, and Michael Beetz. Understanding and Executing Instructions for Everyday Manipulation Tasks from the World Wide Web. In *IEEE International Conference on Robotics and Automation (ICRA). Accepted for publication.*, 2010.

Contact

Moritz Tenorth
Technische Universität München
Intelligent Autonomous Systems Group Boltzmannstr. 3
85748 Garching
Tel.: +49 89 289 26924
Fax: +49 89 289 17757
Email: tenorth@cs.tum.edu