robOCD: Robotic Order Cups Demo
An Interactive Domestic Service Robotics Demo

Stefan Schiffer, Tobias Baumgartner, Daniel Beck, Bahram Maleki-Fard,
Tim Niemueller, Christoph Schwering, and Gerhard Lakemeyer

Knowledge-Based Systems Group, RWTH Aachen University, Aachen, Germany

The Bigger Picture

- Application in Domestic Service Robotics (DSR) as is aimed for in RoboCup@Home league [3]
- Combine modules for interaction with deliberation
- Make use of sophisticated logic-based high-level control which allows for decision-theoretic planning like in [10]

The Robot Caesar

- Intelligent domestic service robot [6]
- Based on former soccer platform
- Component-based control software [7]
- Logic-based [9] high-level control using a Golog [2] dialect called ReadyLog [1]

The Perception

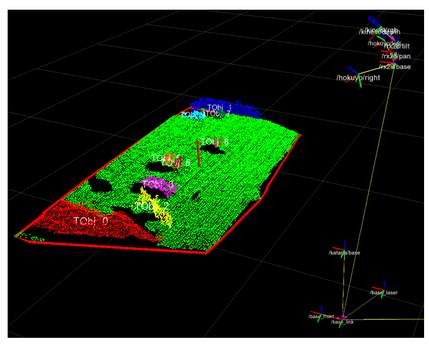
- Data from RGB-D camera (Kinect)
- Fit model for tables and cups

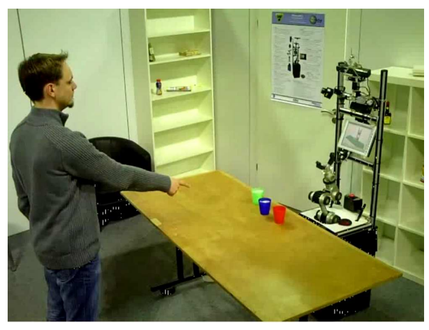
The Interaction Modules

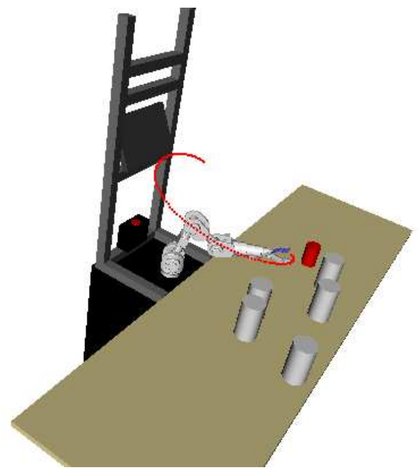
- Robust speech recognition [4]
- 3D (pointing) gesture recognition [5]
- Post output to central blackboard

The Execution Cycle

- Robot perceives scene with initial cup setup and generates local model for motion planning
- User anounces color of cup to place at a position and indicates the desired positions with pointing
- Robot starts planning to compute a program that re-orders the cups with a minimum #moves
- Robot plans trajectories using OpenRAVE [11] and moves cups with 5 DOF arm (Katana6M)







The Task

- Robot should help re-decorating the table
- Gather desired setup of cups on desk from user
- Come up with optimal course of action wrt #moves

The High-level Program

ReadyLog [1] program for robOCD.

- p_1 to p_4 denote four positions on the table
- I_i and P_i are variables that hold color of cup at position i in initial and goal situation, respectively
- $pos(C)$ returns position of cup with color C

```
1  proc order_cups_demo,
2     get_Initial_Order(I1, I2, I3);     %% perceive initial order
3     get_Goal_Order(P1, P2, P3);        %% inquire about goal order
4     sort_cups(P1, P2, P3, 4);          %% start planner
5  endproc

6  proc sort_cups(P1, P2, P3, H),
7     solve(H, reward_cup(P1, P2, P3),
8        while(¬(p1 = pos(P1) ∧ p2 = pos(P2) ∧ p3 = pos(P3))) do
9           pickBest(cup, {red, green, blue},
10             pickBest(to, {p1, p2, p3, p4}, move_cup(cup, pos(cup), to)))
11       endwhile
12    endsolve
13 endproc
```

The Result of Planning

Simplified ReadyLog policy for example run

- initial order was “green, blue, red”
- desired order is “red, green, blue”

```
1  exogf_Update, if ¬done then
2     move_cup(blue, cup_position(blue), p4),
3     exogf_Update, if ¬done then
4        move_cup(green, cup_position(green), p2),
5        exogf_Update, if ¬done then
6           move_cup(red, cup_position(red), p1),
7           exogf_Update, if ¬done then
8              move_cup(blue, cup_position(blue), p3)
9  done
```

Extended Scenario

- Robot drives around looking for tables
- Moves cups from one table to another
- Uses localization & path planning
- Shown at RoboCup German Open & int’l conf.

Knowledge-Based Systems Group
RWTH Aachen University
Prof. G. Lakemeyer, Ph.D.
http://kbsg.rwth-aachen.de/

References

[1] Ferrein, A., Lakemeyer, G.: Logic-based robot control in highly dynamic domains. Robotics and Autonomous Systems **56** (2008) 980–991

[2] Levesque, H.J., Reiter, R., Lespérance, Y., Lin, F., Scherl, R.B.: Golog: A logic programming language for dynamic domains. J Logic Program **31** (1997) 59–84

[3] Wisspeintner, T., van der Zant, T., Iocchi, L., Schiffer, S.: Robocup@home: Scientific Competition and Benchmarking for Domestic Service Robots. Interaction Studies. Special Issue on Robots in the Wild **10** (2009) 392–426

[4] Doostdar, M., Schiffer, S., Lakemeyer, G.: Robust speech recognition for service robotics applications. In: Proc. of the Int’l RoboCup Symposium 2008 (RoboCup 2008). Volume 5399 of LNCS., Springer (2008) 1–12

[5] Schiffer, S., Baumgartner, T., Lakemeyer, G.: A modular approach to gesture recognition for interaction with a domestic service robot. In: Proc. Int’l Conf. on Intelligent Robotics and Applications. LNCS, Springer (2011) 348–357

[6] Schiffer, S., Ferrein, A., Lakemeyer, G.: Caesar – An Intelligent Domestic Service Robot. Journal of Intelligent Service Robotics (2012) to appear

[7] Niemueller, T., Ferrein, A., Beck, D., Lakemeyer, G.: Design Principles of the Component-Based Robot Software Framework Fawkes. In: Proc. Int’l Conf. on Simulation, Modeling, and Programming for Autonomous Robots, Springer (2010)

[8] Niemueller, T., Ferrein, A., Lakemeyer, G.: A Lua-based Behavior Engine for Controlling the Humanoid Robot Nao. In: Proc. Int’l RoboCup Symposium 2009 (RoboCup 2009). Volume 5949 of LNCS., Springer (2009)

[9] Reiter, R.: Knowledge in Action: Logical Foundations for Specifying and Implementing Dynamical Systems. The MIT Press (2001)

[10] Boutilier, C., Reiter, R., Soutchanski, M., Thrun, S.: Decision-theoretic, high-level agent programming in the situation calculus. In: Proc. of the 17th Nat’l Conf. on Artificial Intelligence (AAAI’00), Menlo Park, CA (2000) 355–362

[11] Diankov, R., Kuffner, J.: Openrave: A planning architecture for autonomous robotics. Tech. Rep. CMU-RI-TR-08-34, Robotics Institute, Pittsburgh, PA (2008)