Combining Perception and Knowledge Processing for Everyday Manipulation - K-CopMan

Dejan Pangercic, Moritz Tenorth, Dominik Jain, Michael Beetz

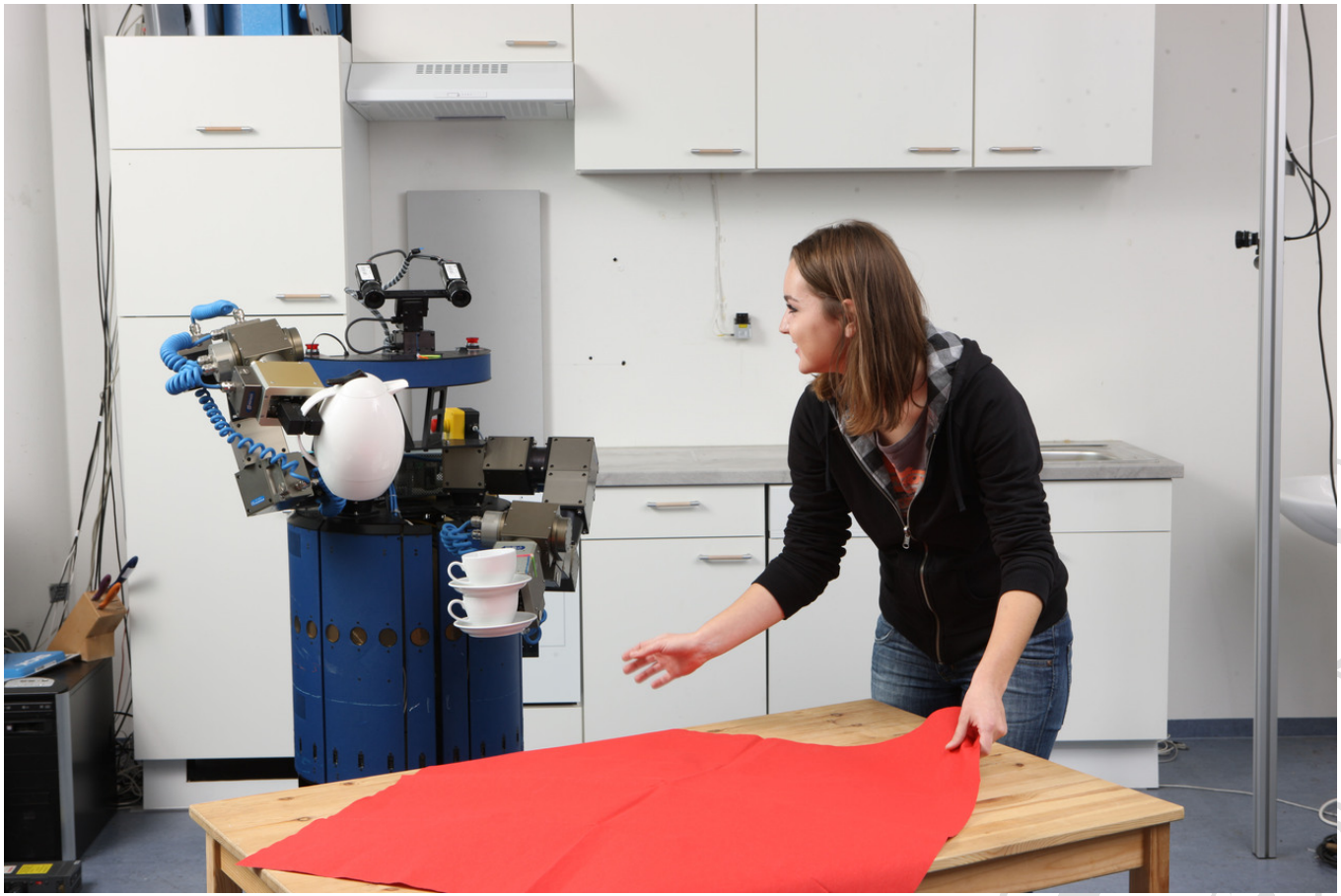
Intelligent Autonomous Systems
Technische Universität München

IROS 2010, October 2010
Taipei, Taiwan

Intelligent Autonomous Systems

Scenario/Motivation

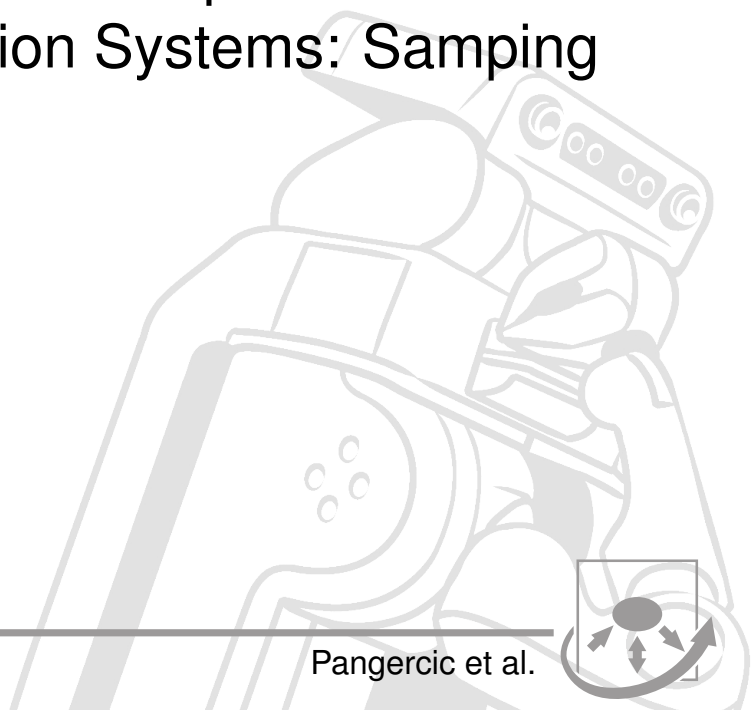
What is missing on the table?



*

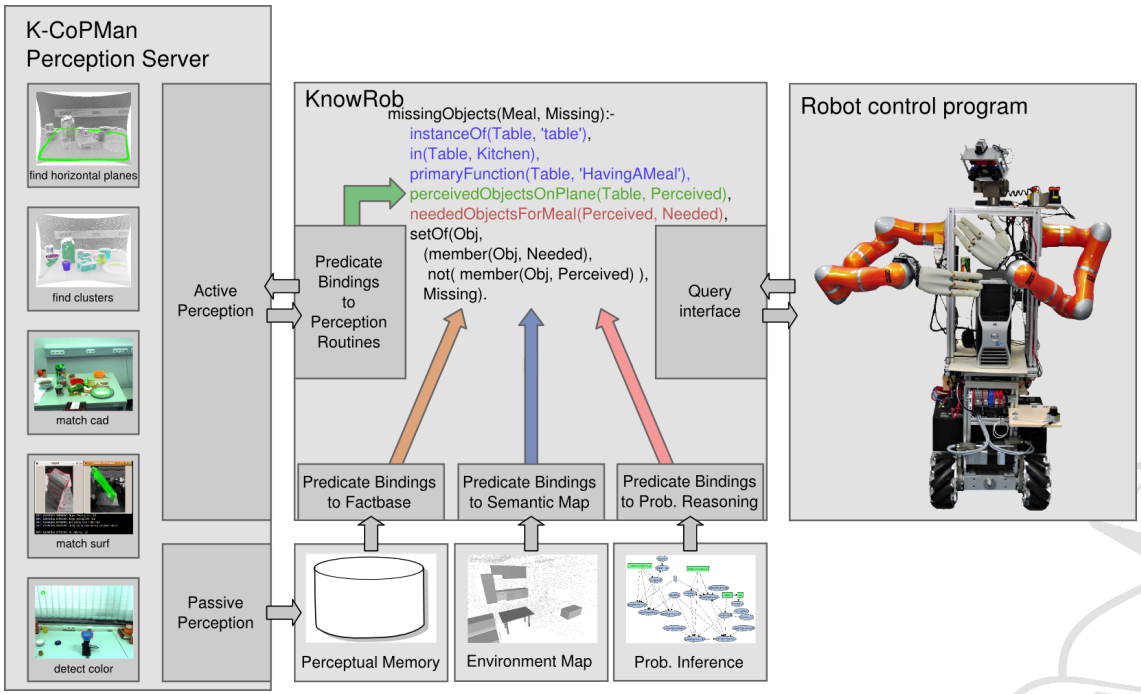
RW

- I. Horswill, Integrating vision and natural language without central models, in In Proc. of the AAAI Fall Symposium on Embodied Language and Action
- B. Neumann and R. Möller, On Scene Interpretation with Description Logics, in Cognitive Vision Systems: Samping the Spectrum of Approaches



System Overview

Overview

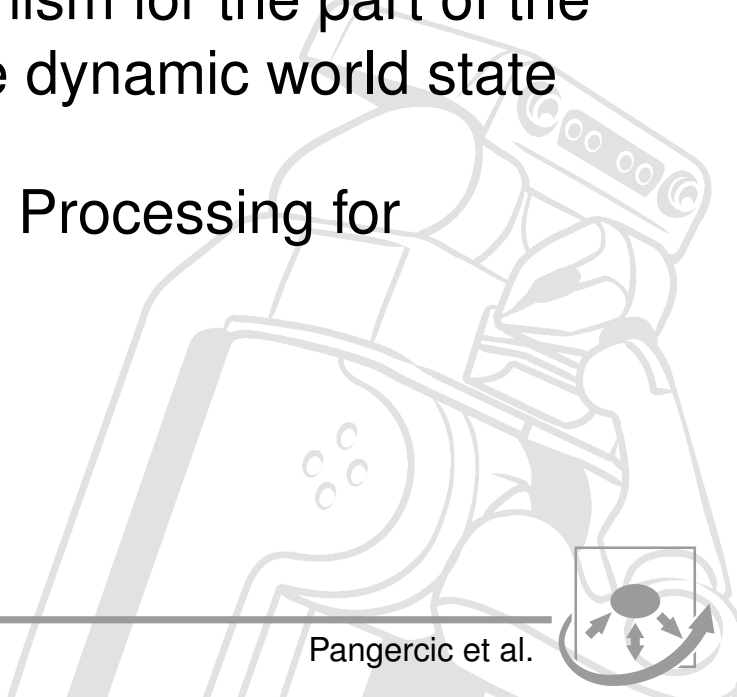


- Provides the robot with abstract symbolic knowledge about the perceived scenes
- Uses abstract symbolic knowledge for accomplishing perception tasks
- Answers new types of queries that require the combination of knowledge processing and perception.

Knowrob Extensions

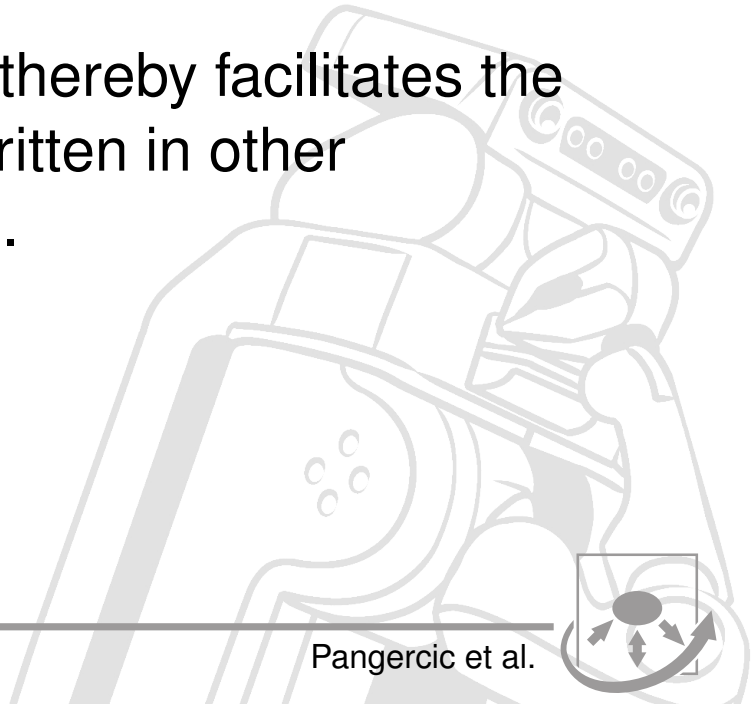
- provides a set of predicates that abstract away from the robot's perceptual mechanisms and transforms the perceptual tasks and their results into a logical representation suitable for knowledge processing and decision-making
- provides a continual update mechanism for the part of the knowledge base that represents the dynamic world state

[IROS2009:] Moritz Tenorth, Knowledge Processing for Autonomous Personal Robots



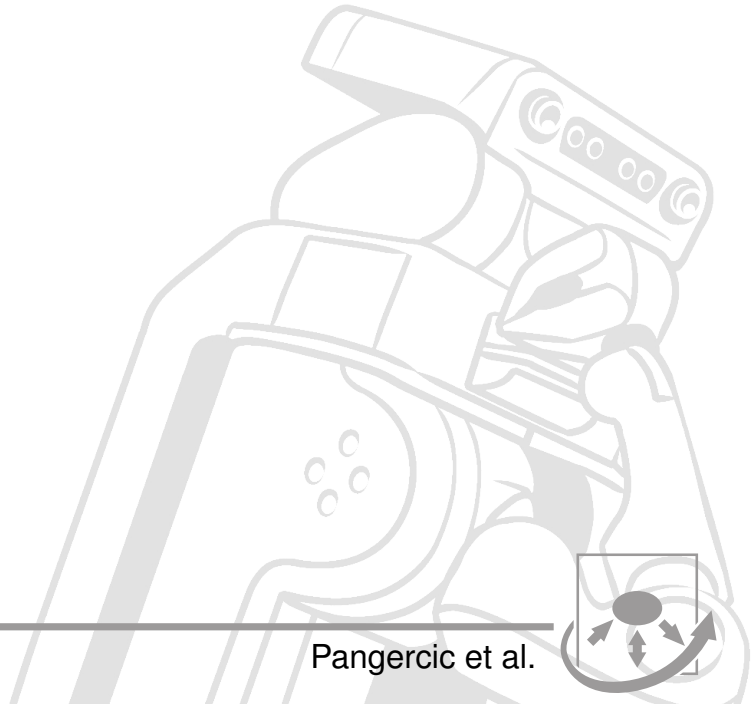
K-CopMan Technical Details

- interface layer to open-source SWI Prolog.
- combines fast inference and computation with declarative, logics-based semantics.
- can even run in feedback loops up to 10 Hz to make the robot action-aware.
- Prolog's foreign language interface thereby facilitates the integration of perception routines written in other programming languages like C/C++.



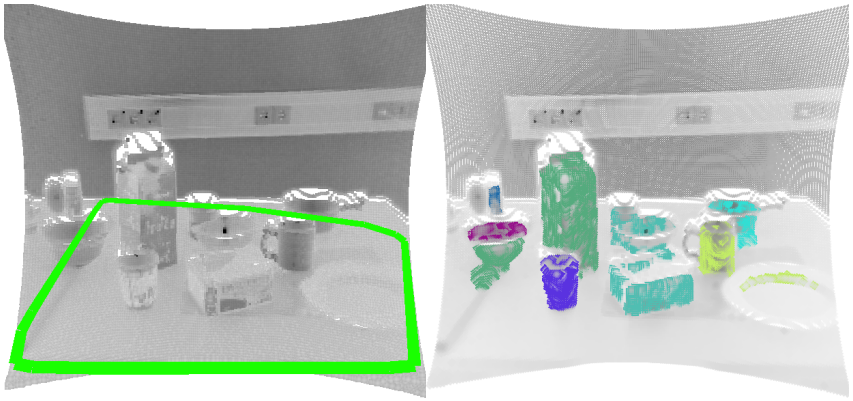
perceivedObjectsOnPlane

```
perceivedObjectsOnPlane(Plane, Perceived) :-
    onPlane(Plane),
    setOf(Obj-Hyp,
          (  on(Obj, Plane),
             category(Obj,Cat),
             uniqueId(Id),
             objectInstace(Obj,KnownObj),
             Obj-Hyp = [Id,Obj,Cat,KnownObj]),
          Perceived).
```

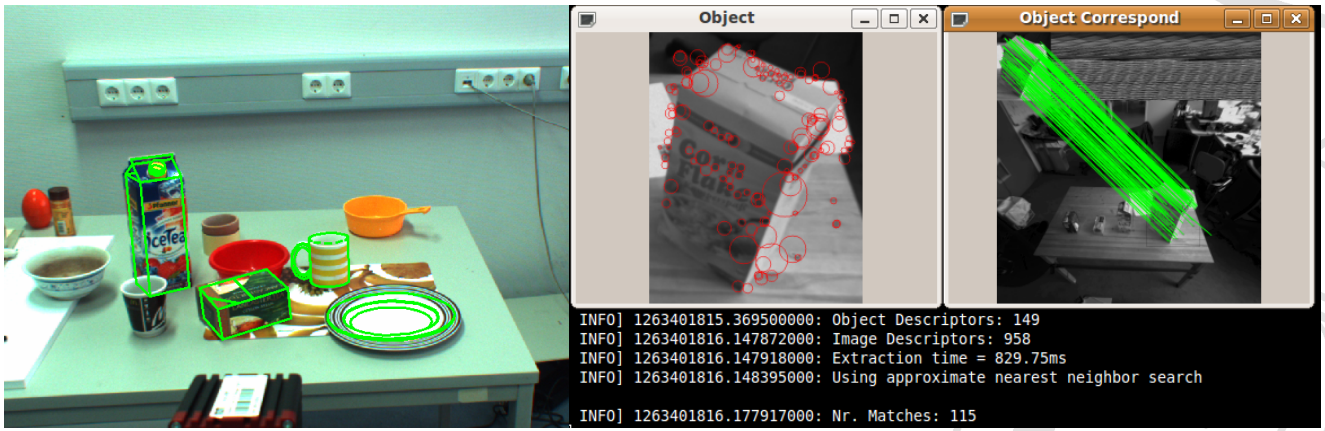


Perception Routines

Preprocessing:

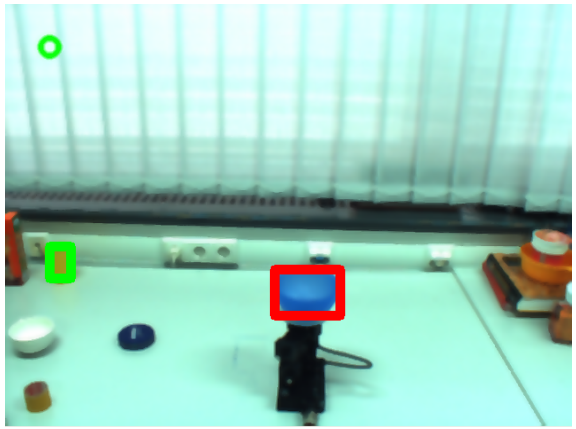


CAD model & Chamfer & SURF matching:

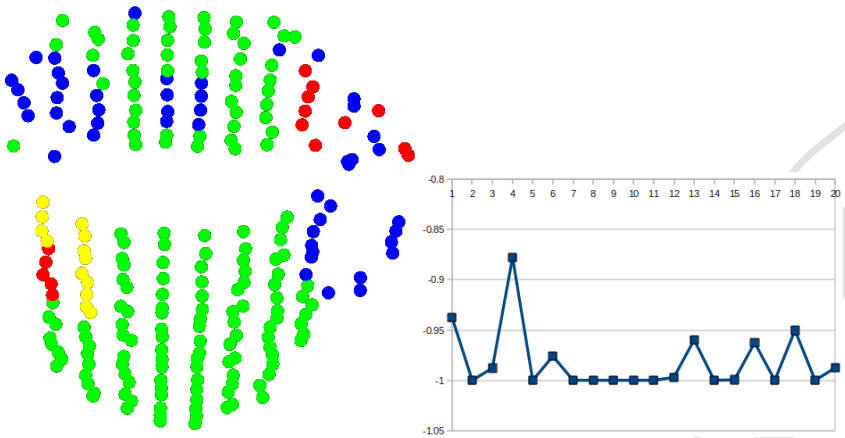


Perception Routines Cont.

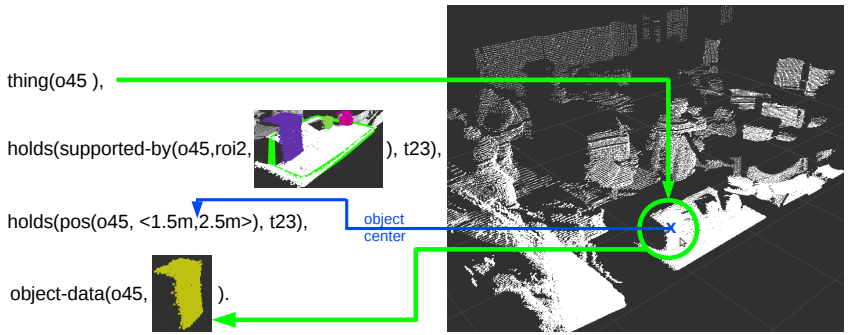
Color segmentation:



Global Radius-based Surface Descriptor (GRSD):

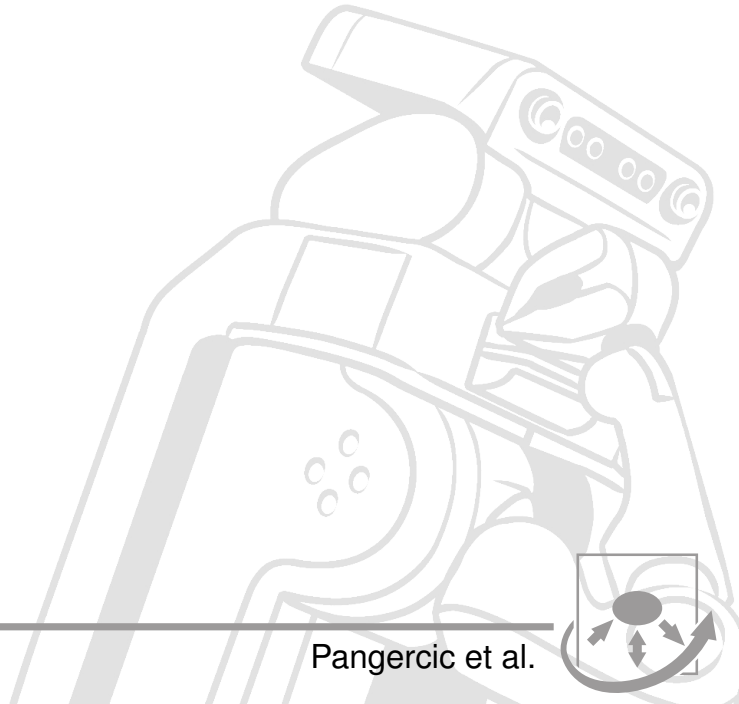


Passive Perception & Memory

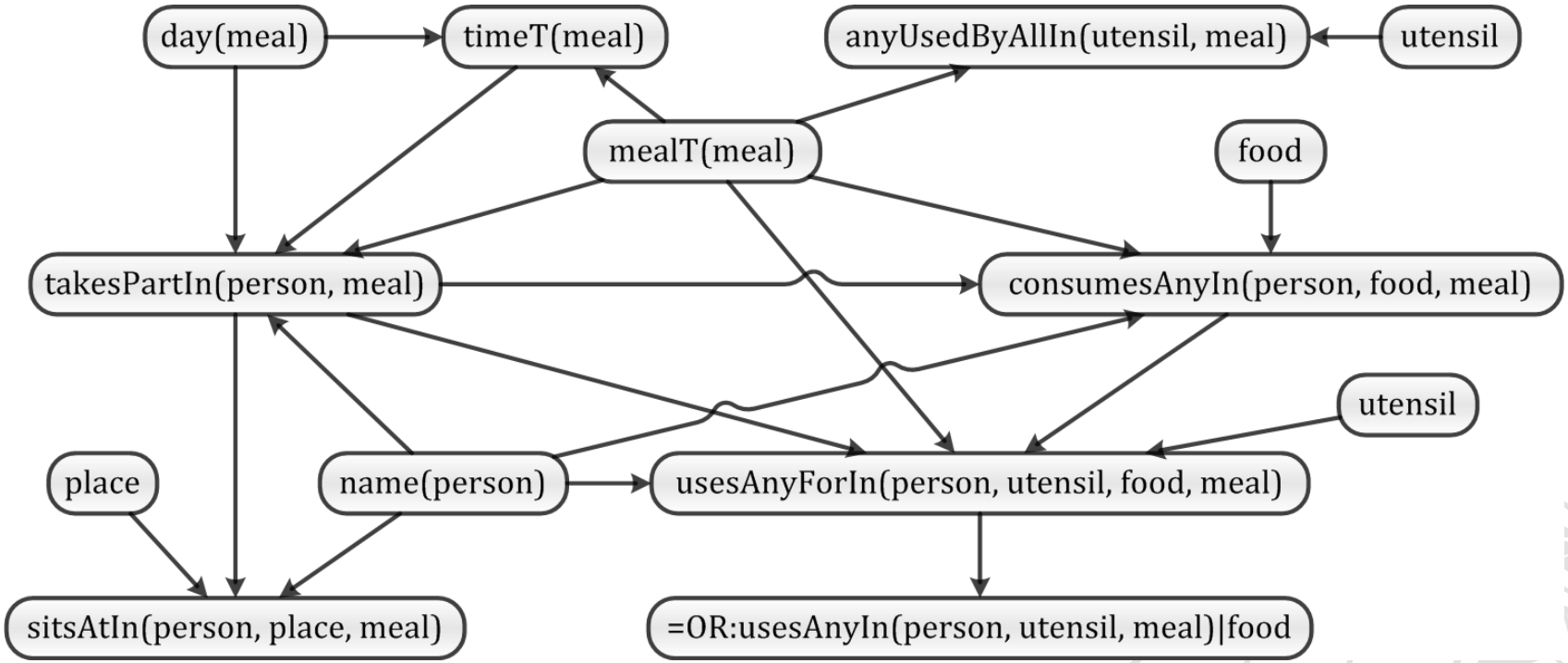


K-Copman Predicates

- ***holds(onPlane(Obj,Plane),ti)***
- ***holds(position(Obj,Pos),ti)***
- ***holds(spatial-rel(Obj_1,Obj_2),ti)***
- ***categorize(Obj, Cat)***
- . . .

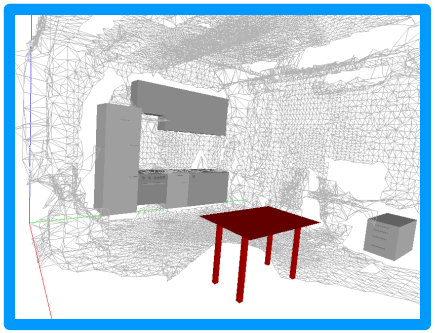


Probabilistic First-Order Reasoning



All Together - Missing Objects Query

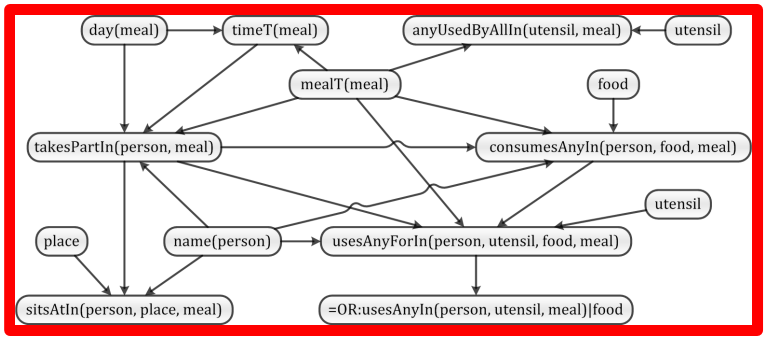
Semantic Map, Encyclopedic Knowledge

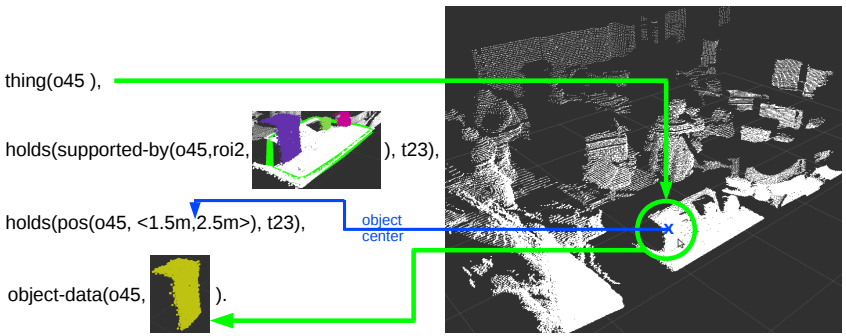


K-Copman perception server

```
missingObjects(Meal, Missing):-
  instanceOf(Table, 'table'),
  in(Table, Kitchen),
  primaryFunction(Table, 'HavingAMeal'),
  perceivedObjectsOnPlane(Table, Perceived),
  neededObjectsForMeal(Perceived, Needed),
    setOf(Obj,
      (member(Obj, Needed),
        not( member(Obj, Perceived)),
      Missing).
```

First-Order Probabilistic Reasoning





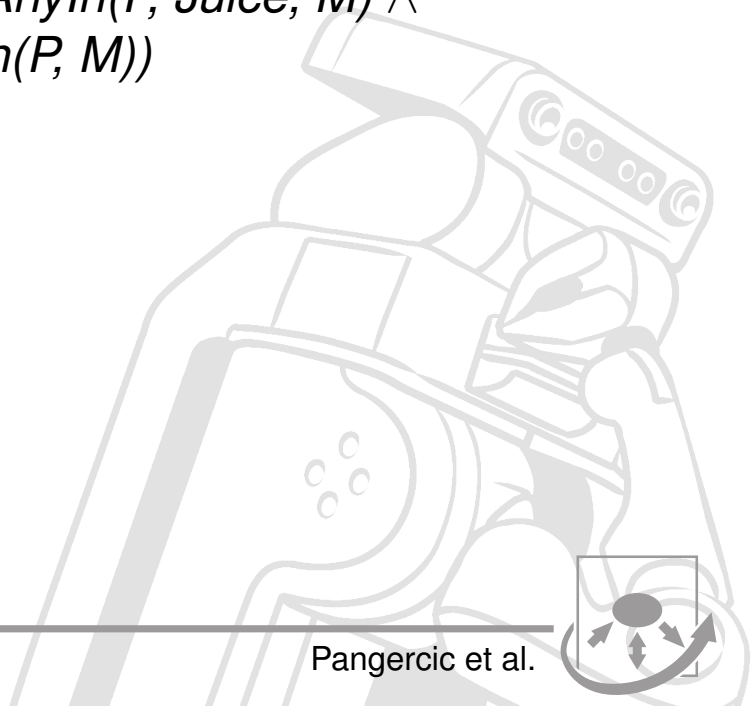

```
perceivedObjectsOnPlane(Plane, Perceived) :-
  onPlane(Plane),
  setOf(Obj-Hyp,
  (    on(Obj, Plane),
    category(Obj,Cat),
    uniqueId(Id),
    objectInstace(Obj,KnownObj),
    Obj-Hyp = [Id,Obj,Cat,KnownObj]),

  Perceived).
```

Probabilistic Query

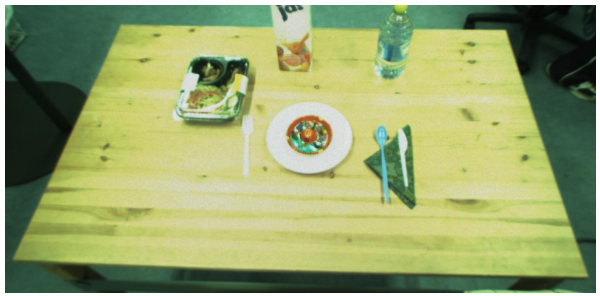
$$P(usesAnyIn(P, ?u, M), consumesAnyIn(P, ?f, M) \mid mealT(M) = Lunch \wedge usesAnyIn(P, Plate, M) \wedge usesAnyIn(P, Knife, M) \wedge usesAnyIn(P, Fork, M) \wedge usesAnyIn(P, Spoon, M) \wedge usesAnyIn(P, Napkin, M) \wedge consumesAnyIn(P, Salad, M) \wedge consumesAnyIn(P, Pizza, M) \wedge consumesAnyIn(P, Juice, M) \wedge consumesAnyIn(P, Water, M) \wedge takesPartIn(P, M))$$

$$\approx \langle\langle\ Glass: 1.00, Bowl: 0.85, Cup: 0.51, \ldots\rangle,$$
$$\langle\ Soup: 0.82, Coffee: 0.41, Tea: 0.14, \ldots\rangle\rangle$$

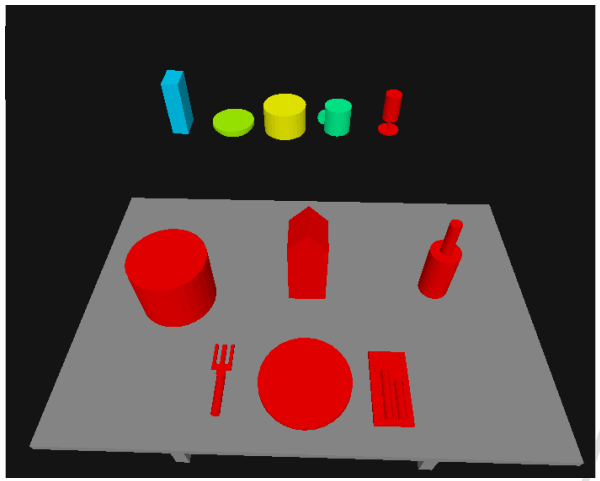


Example Scene, Breakfast

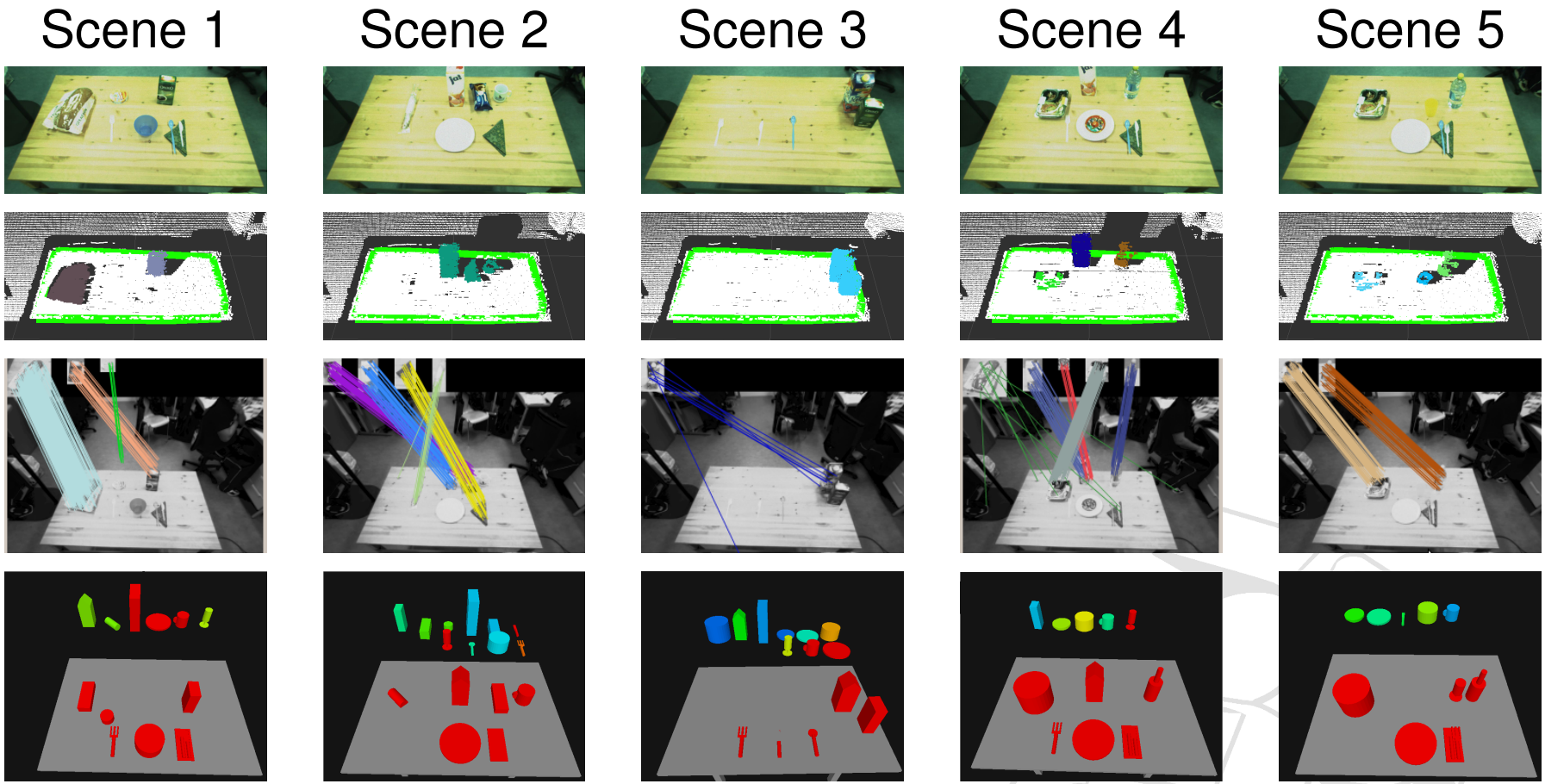
Perceived Objects: plate, fork, spoon, knife, napkin, salad, juice, pizza, water



Inferred Objects: coffee, soup, bowl, cup, glass



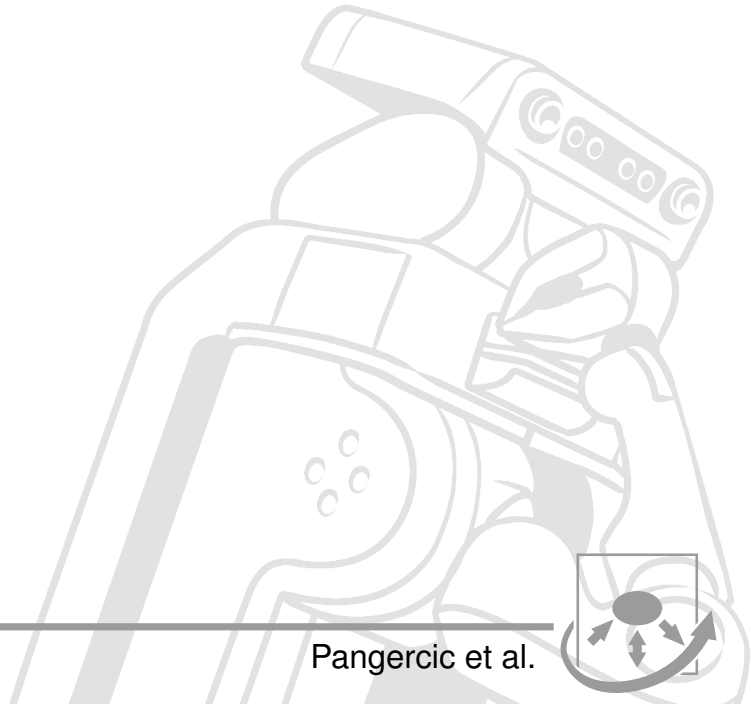
Evaluation results



NB:Last row depicts inferred missing objects, hue indicates probability: Red corresponds to 1.0, with orange, yellow, green and blue denoting declining probabilities in this order.

Future Work

- More and more specific perception routines
- Spatio-temporal reasoning
- Life-long learning using the passive perception
- Perception-to-knowledge cues



Thanks

Thanks!
Available in TUM ROS Package Repository:
`http://tum-ros-pkg.svn.sourceforge.net/`
(knowrob, prolog_perception)
Contact:
dejan.pangercic@cs.tum.edu