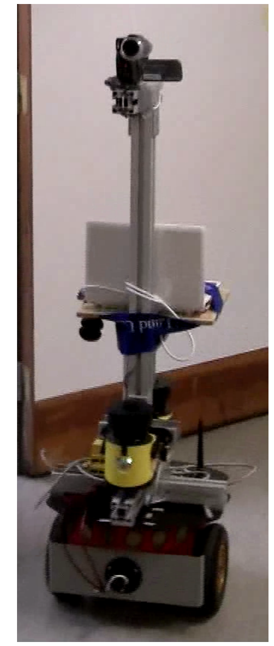
AI in Robot(ic)s

Applied artificial intelligence (EDA132)
Lecture 12
2017-02-24
Elin A. Topp

Course book (chapter 25), images & movies from various sources, and original material
(Some images and all movies removed for the uploaded PDF)

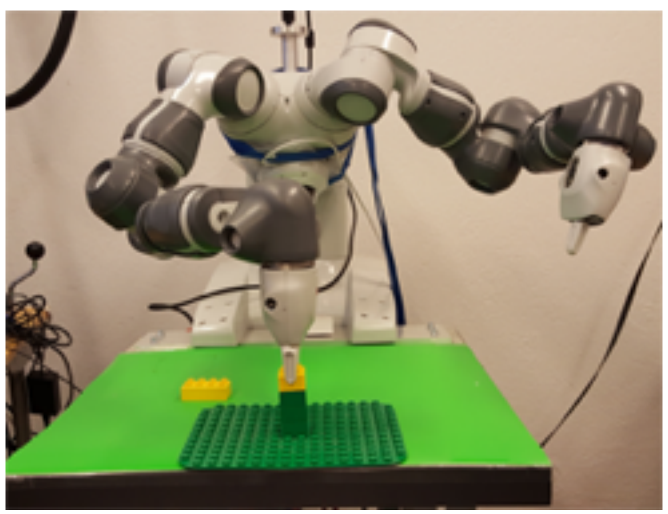
What is a “Robot”?



✓ ✓ ✓

Honda Asimov

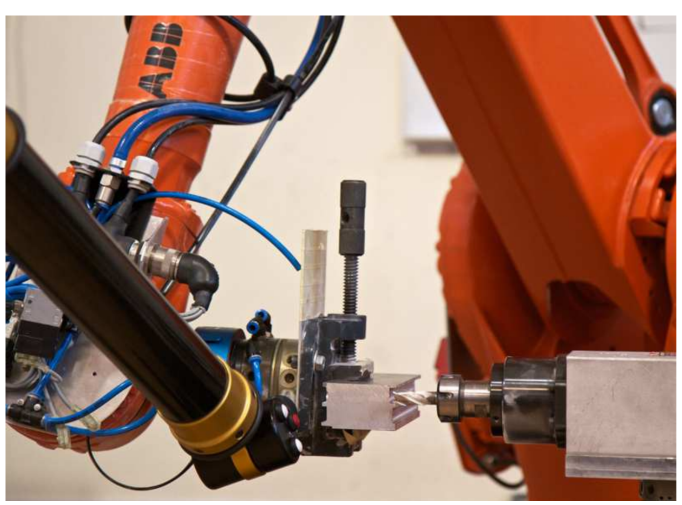
✓



✓

?

Keepon



Leonardo

?

✓

✓

iCub ✓

…

Robots, and what they can do…

How far have we come?

ABB robots and their precision... 2009 (Youtube "ABB robots / Fanta cans")

Frida "feels" when work's done... 2013 (Youtube "Magnus Linderoth, sensorless force sensing")

YuMi wraps gifts… 2015 (https://youtu.be/FHGC9mSGpKI)

Types of robots

Industrial robots vs. service robots vs. personal robots / robot toys

Static manipulators vs. mobile platforms (vs. mobile manipulators)

Mechanistic vs. humanoid / bio-inspired / creature-like

For all in common:
A robot is a physical agent in the physical world
(with all the consequences that might have... ;-)

(Darpa Urban Challenge 2007,
Georgia Tech "Sting Racing" crash)

(Darpa Rescue Challenge 2015,
Robots falling - MIT DRC, foot tremble)

Ethics detour

Robots as embodiment of artificially intelligent systems -
but even reasoning mechanisms can only build upon a given baseline.

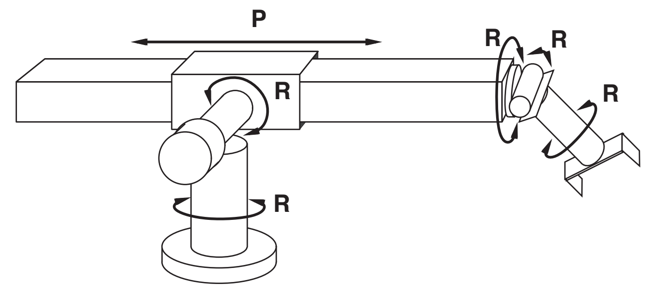
So far, systems will take instructions literally, and only reason within given limits.

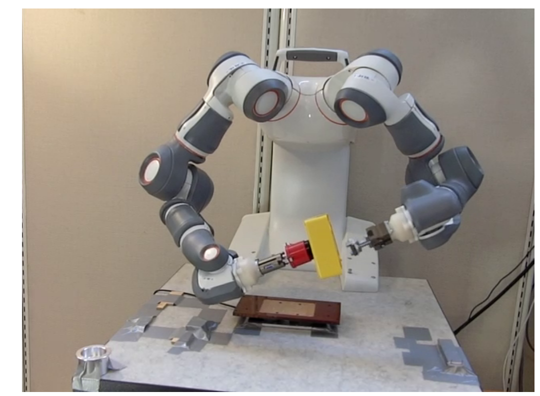
AI-systems must be capable of explaining themselves, and we should not expect them to be more than they are!

Excerpt from Robot & Frank, "stealing"

Robot actuators - joints and wheels

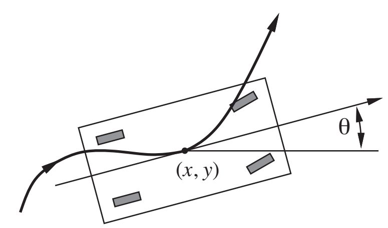
6 DOF (6 “joint”) arm:





2x7 DOF (“humanoid” torso “YuMi” / Frida):

2 (3 effective) DOF synchro drive (car):



2 (3 effective) DOF differential drive (Pioneer p3dx):

3 DOF holonomic drive (“shopping cart”, DLR’s Justin):

Kinematics - controlling the DOFs

Direct (forward) kinematics (relatively simple):

Where do I get with a certain configuration of parts / wheel movement?

Inverse kinematics (less simple, but more interesting):

How do I have to control joints and wheels to reach a certain point?

Dynamics - controlling consequences of movement

Dynamics:

Make the robot move (and move stuff) without falling apart, or crashing into things

How much payload is possible?

How fast can I move without tipping over?

What is my braking distance?

How do I move smoothly? (ask the automatic control people ;-)

Weight: ca 1300 kg

Payload: ca 150 kg

Dynamics in practice

Dynamics also gets you into two problems: direct and inverse dynamics.

Direct dynamics:

Given masses, external forces, position, velocities and acceleration in the joints / wheels, what forces / moments are put to the depending joints and the tool centre point (TCP)? "Rather" simply solvable, at least more or less straight forward.

Inverse dynamics (again, more interesting than direct dynamics):

While solving the *inverse kinematics* problem is nasty, but still "only" a bunch of linear equations, solving the *inverse dynamics* problem leaves you with a bunch of more or less complex differential equations.

Supporting parts: Sensors

In a predictable world, we do not need perception, but good planning and programming

As the world is somewhat unpredictable, some perception is useful, i.e., robots / robot installations need sensors.

Passive / active sensors.

Range / colour / intensity / force / direction ...

Optical / sound / radar / smell / touch ...

Most common for mobile robots: position (encoders / GPS), range (ultrasound or laser range finder), image (colour/intensity), sound

Most common for manipulators: position (encoders), force / torque, images, (range - infrared, laser RF)

Sensors on a mobile robot

Microphones (sound)

Ultrasound (24 emitters / receivers) (range)

Camera (image - colour / intensity)

Laser range finder (SICK LMS 200) (range)

Infrared (range / interruption)

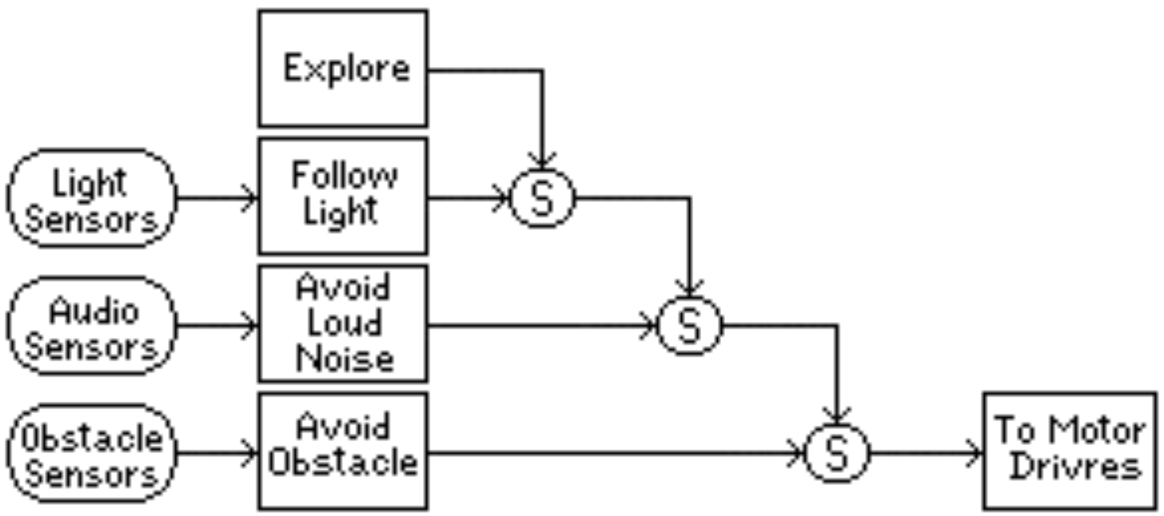
Bumpers (touch)

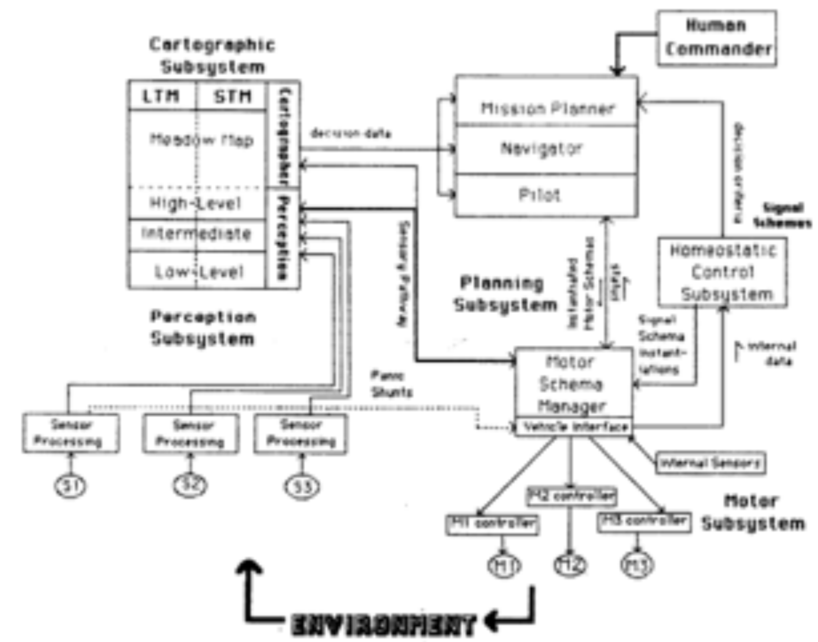
Wheel encoders (position / pose)

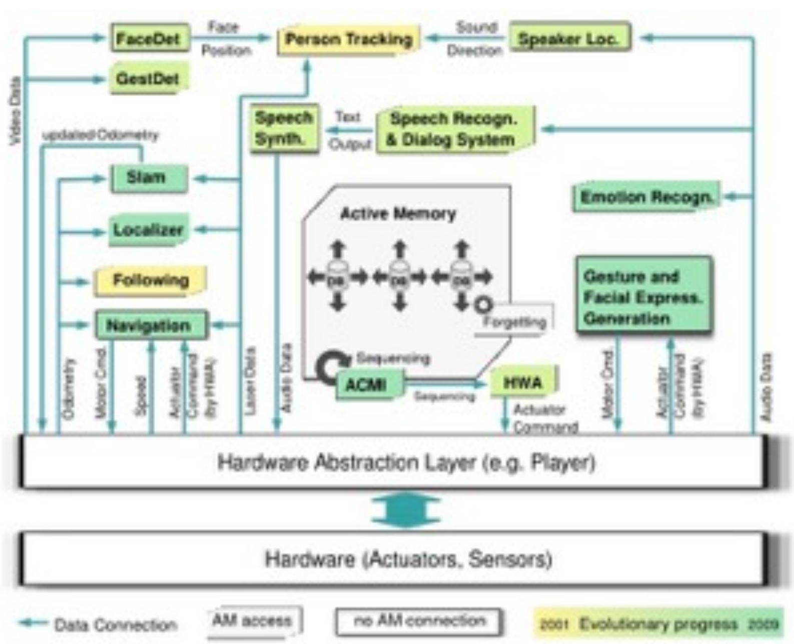
System integration

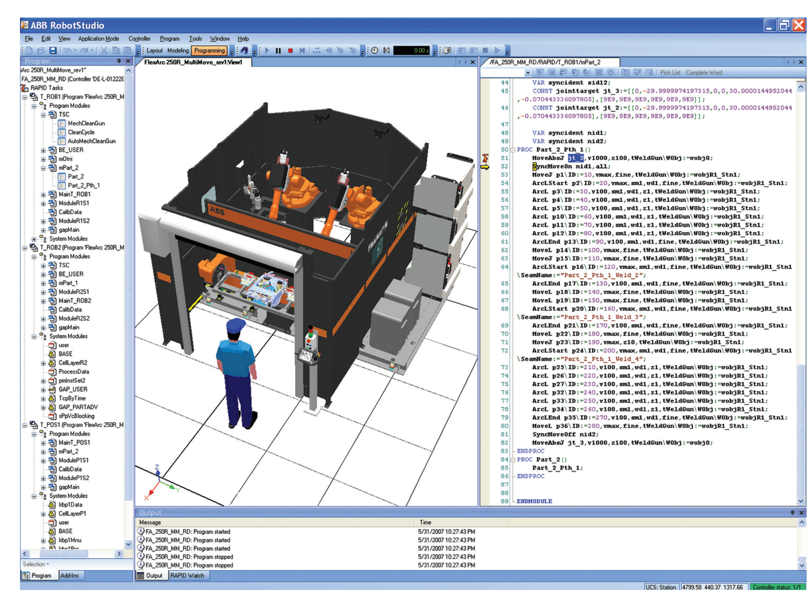
Make sensors, actuators and algorithms work together

Architectures, "operating systems", controllers, programming tools ...









System integration - system is bigger than the sum of its components

Research video from user study

“Flur / Tuer” - “Corridor / Door”

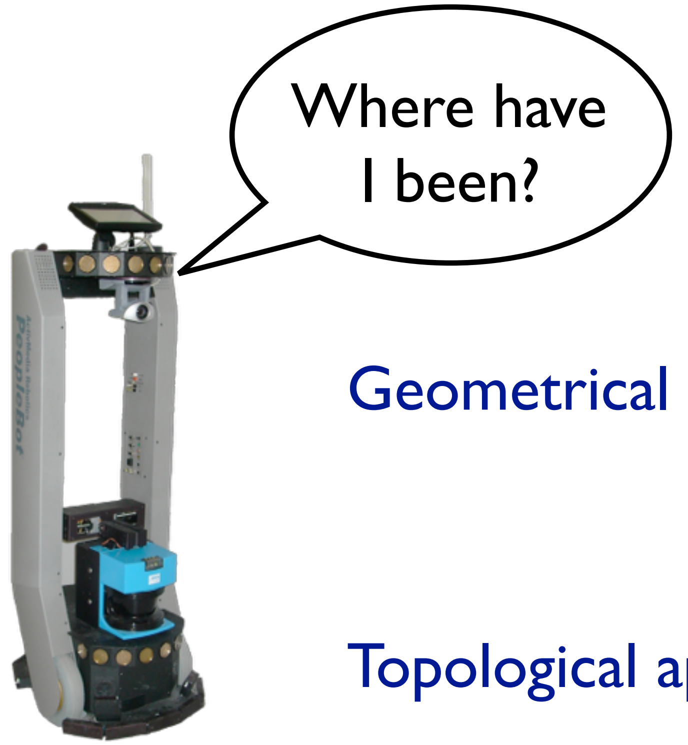
Outline

AI in Robotics - integrating the "brain" into the "body" (just SOME examples!)

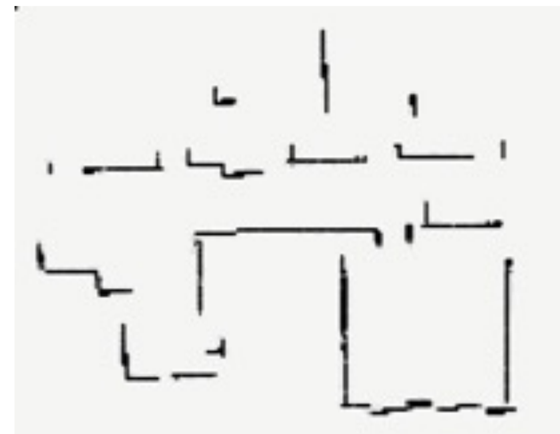
- Probabilistic methods for Mapping & Localisation
- Deliberation & High level decision making and planning

- SJPDAFs for person tracking
- Identifying interaction patterns in Human Augmented Mapping with BNs
- Knowledge representation, reasoning, and NLP to support HRI and high-level robot programming

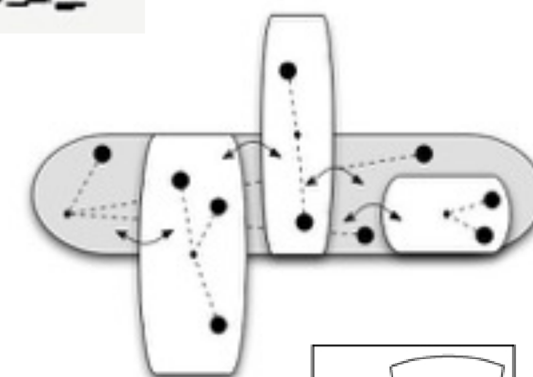
Mapping



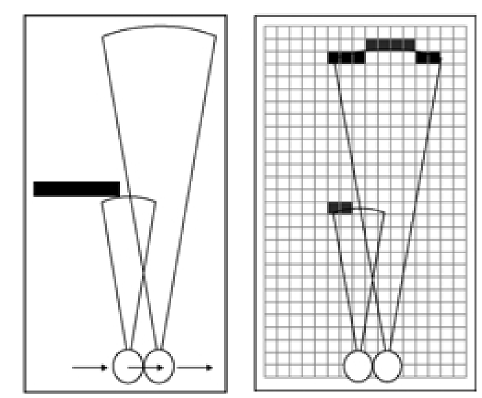
Geometrical approaches



Topological approaches

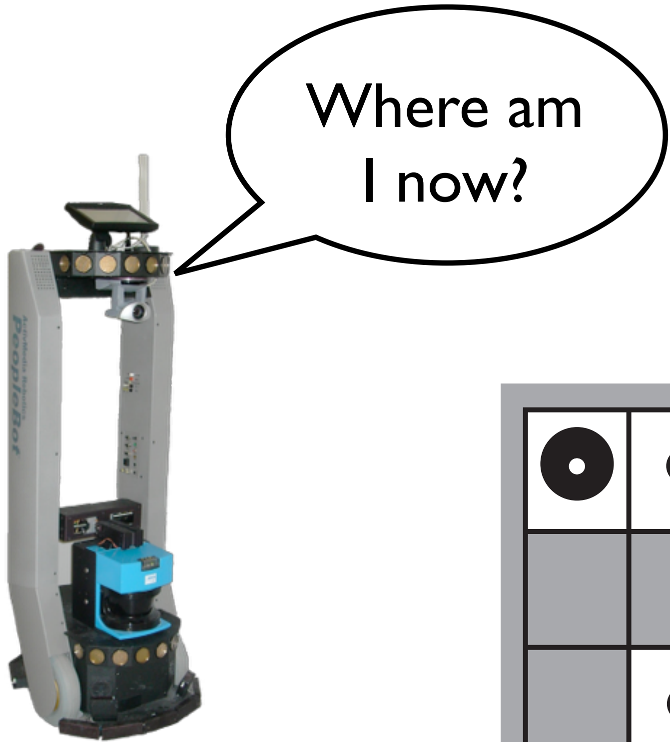


Occupancy grid approaches (e.g., Sebastian Thrun)

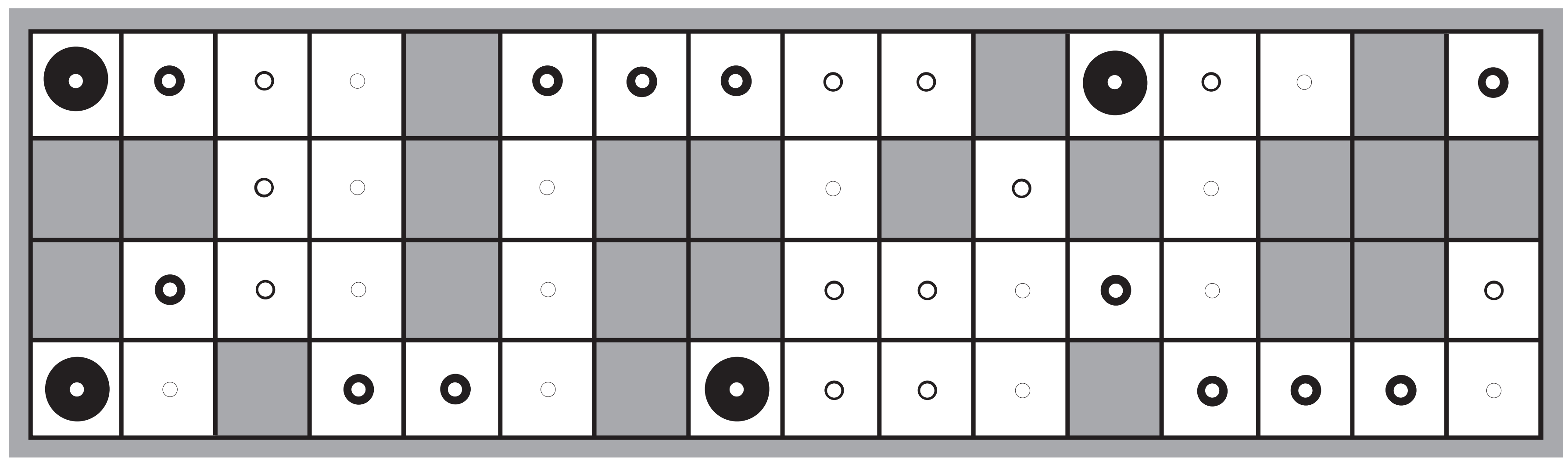


(Hybrid approaches)

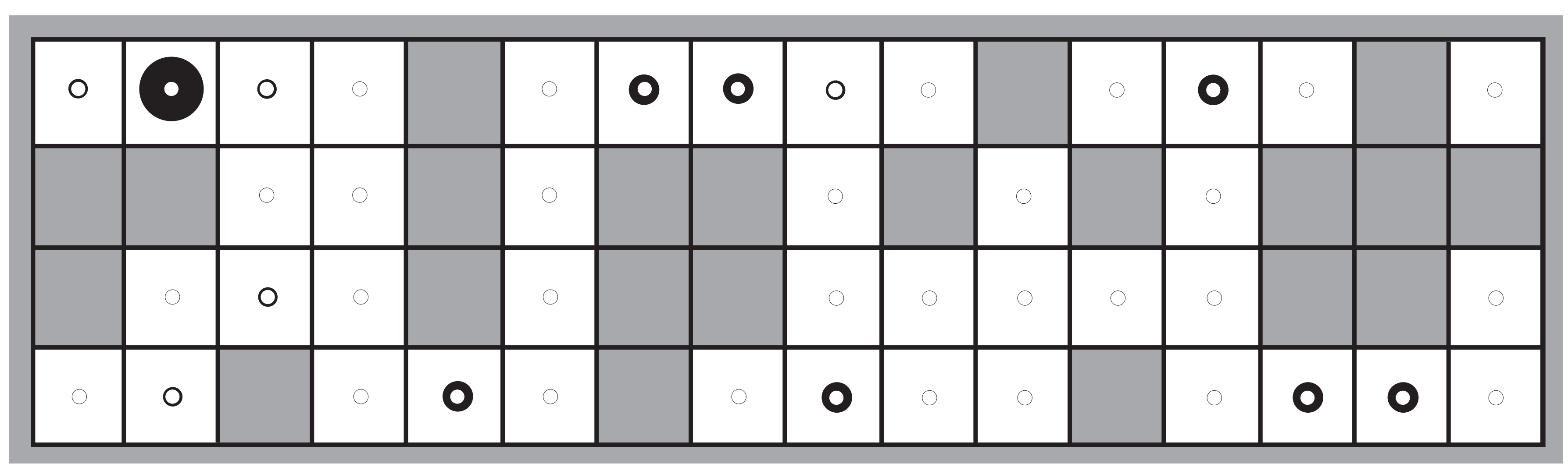
Localisation



HMM in a grid world



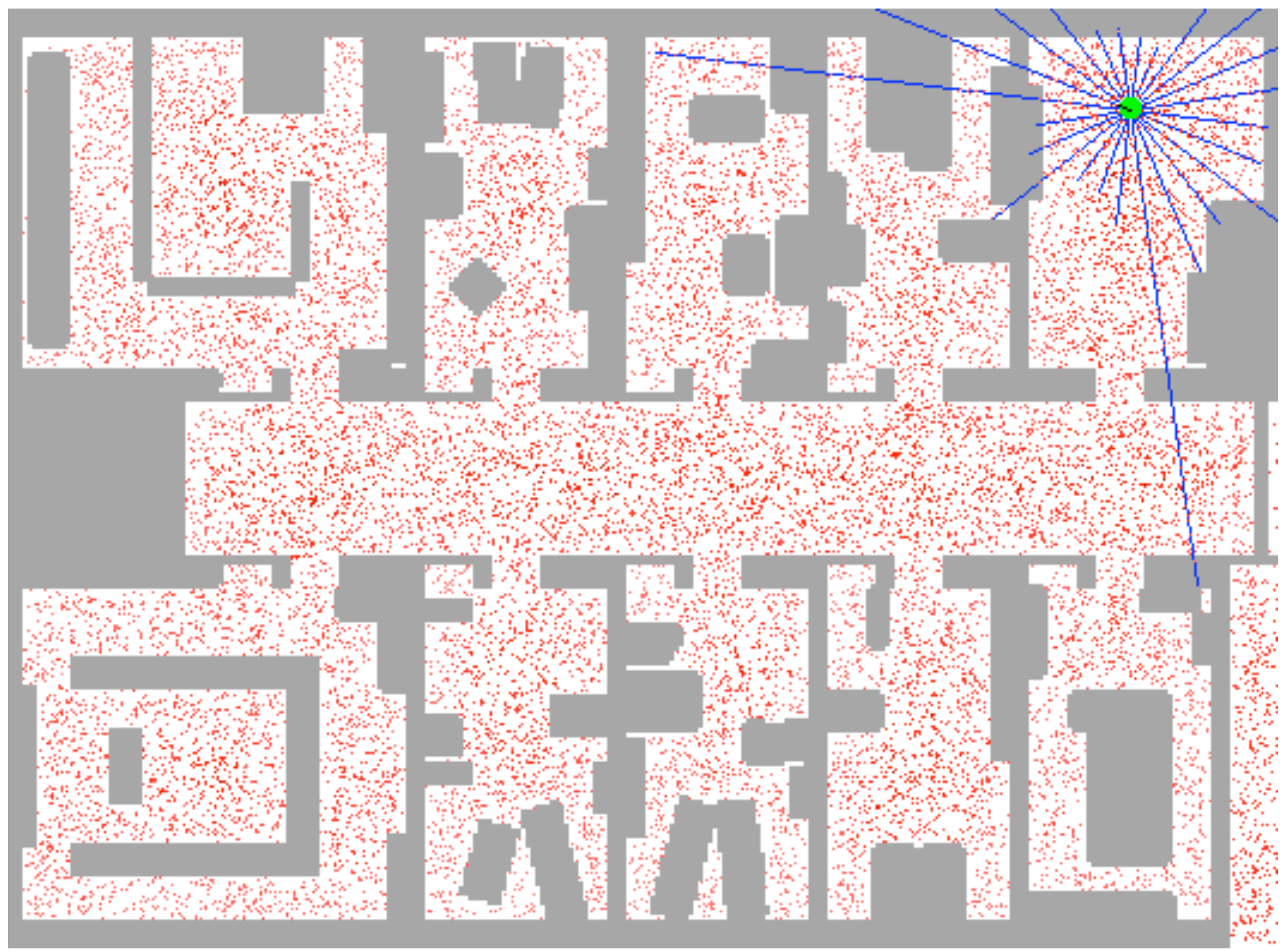
(a) Posterior distribution over robot location after $E_1 = NSW$



(b) Posterior distribution over robot location after $E_1 = NSW, E_2 = NS$

Localisation

E.g., Monte Carlo Localisation (D. Fox, S. Thrun, *et al.*)



Data filters for state estimation

0. Represent state, identify system function

1. Estimate / predict state from model applying the function

2. Take a measurement

3. Update state according to model and observation (measurement)

Used for position tracking, detection of significant changes in a data stream, localisation ...

E.g., particle filters (Monte Carlo), Kalman filters

Particle filter

1. Represent possible positions by samples (uniform distribution) $\boldsymbol{x} = (x, y, \theta)$

2. Estimate movement / update samples according to assumed robot movement + noise

3. Take a measurement z

4. Assign weights to samples according to posterior probabilities (Bayes!) $P(x_i \mid z)$

5. Resample (pick "good" samples, use those as new "seeds", redistribute in position space and add some noise), continue at 2.

Kalman filter

Represent posterior with a Gaussian.

Assume linear dynamical system
(F, G, H system matrices, u measurement, v, w, gaussian noise)

$x(k+1) = F(k)\ x(k) + G(k)\ u(k) + v(k)$ (state estimate)

$y(k+1) = H(k)\ x(k) + w(k)$ (output)

1. Predict based on last estimate:

$x'(k+1 \mid k) = F(k)\ x'(k \mid k) + G(k)\ u(k) + v(k)$

$y'(k+1 \mid k) = H(k)\ x'(k+1 \mid k) + w(k)$

2. Calculate correction based on prediction and current measurement:

$\Delta x = f(y(k+1), x'(k+1 \mid k))$

3. Update prediction:

$x'(k+1 \mid k+1) = x'(k+1 \mid k) + \Delta x$

Mapping & Localisation: Chicken & Egg?

Simultaneous localisation and mapping (SLAM)

While building the map, stay localised!

Use filters to “sort” landmarks:

Known? Update your pose estimation!

Unknown? Extend the map!

Deliberation in, e.g., a navigation system

A robotic system might have several goals to pursue, e.g.,

- Explore the environment (i.e., visit as many areas as possible and gather data) and build a map
- Use a certain strategy (e.g., follow the wall to the right)
- Do not bump into things or people on the way
- Go "home" for recharging in time

Behaviours (e.g., as used by Arkin) can take care of each of the goals separately

Particular perception results can be fed into a control unit for decision making

This decision making unit (deliberation process) can assign weights (priorities) to the behaviours depending on the sensor data.

E.g., when battery level sensor reports a certain level, only the "going home" behaviour and immediate obstacle avoidance are allowed to produce control output, exploring and wall following are ignored.

More complex decisions / plans

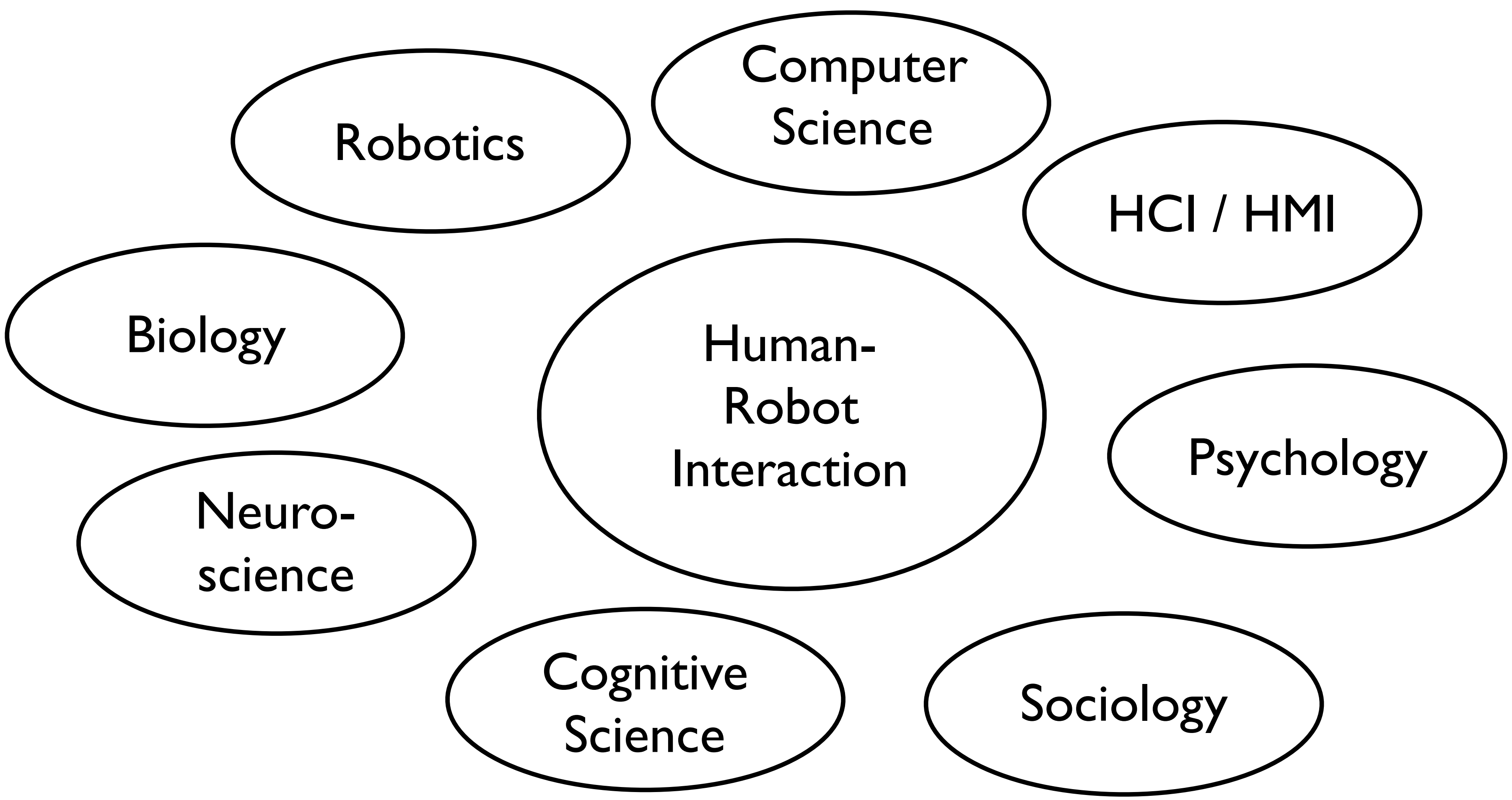
If the system does not only involve one robot with several "competencies", but several robots with partly overlapping, partly complementary abilities, the decisions are to be taken to another dimension:

- Given a task, what do I need to know to fulfill it?
- Do I know these things?
- Given I know what to do, do I have the means (robot) to do it?
- If yes, which one?
- Given different steps and parts of a task, can things be done in parallel?
- By which robot?
- What if something goes wrong with one part of the plan? Does this affect the whole task execution, or only one of the robots?

HRI - going beyond pressing buttons

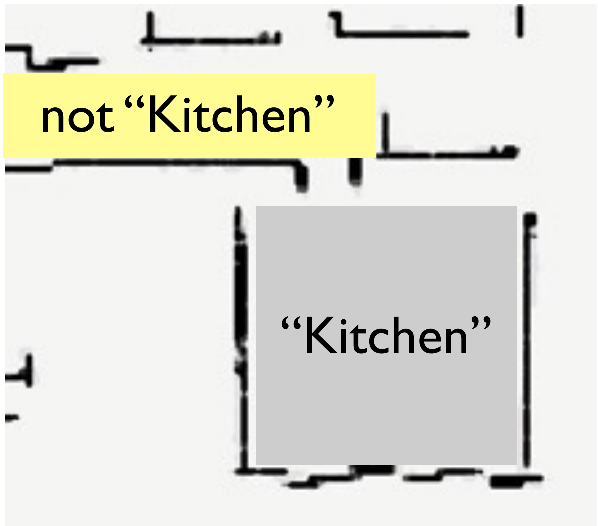
Human-Robot Interaction is quite new as a research field of its own

Like AI and Robotics themselves it is quite multidisciplinary



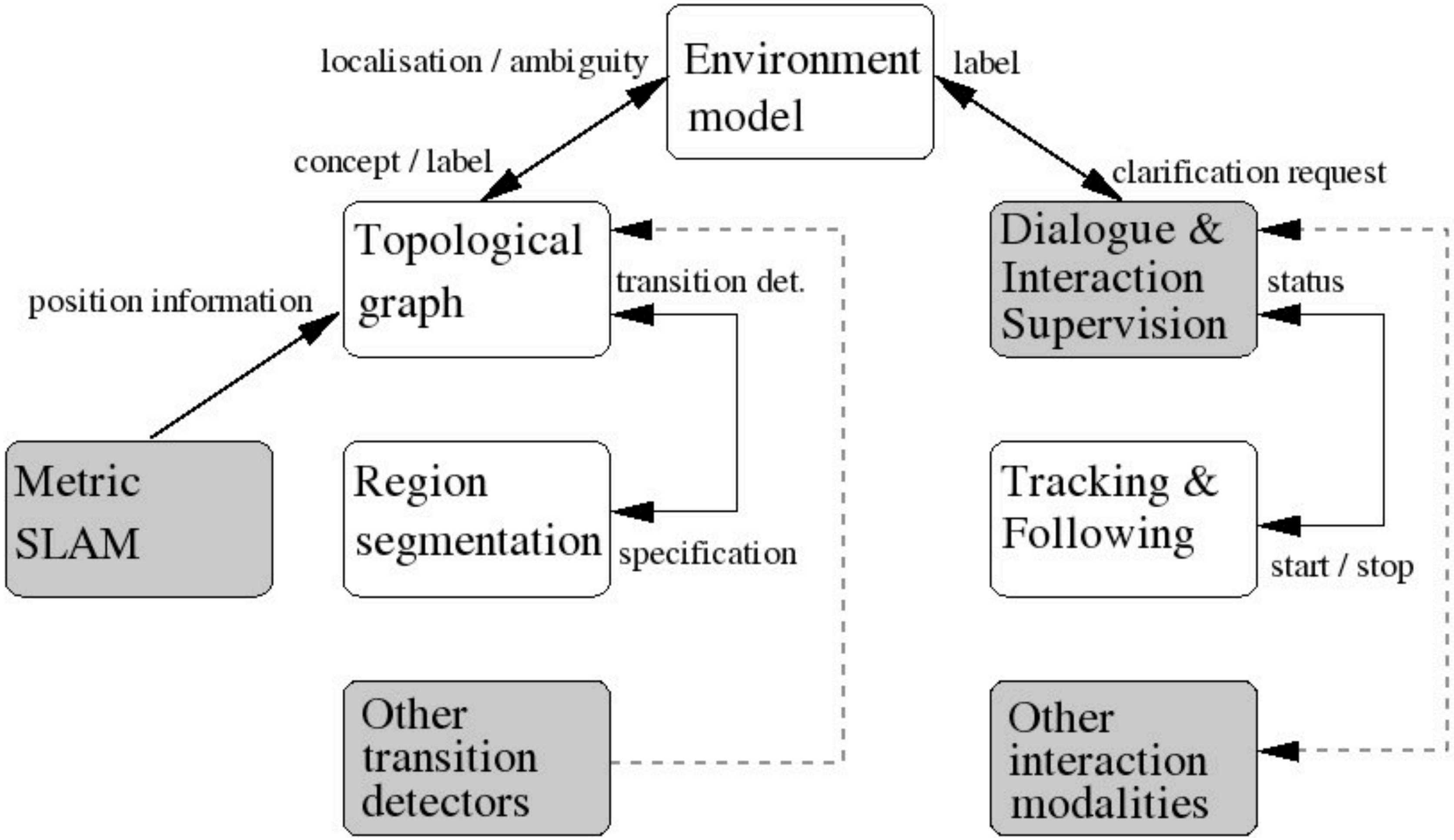
Human augmented mapping - an example for work in HRI

- Integrate robotic and human environment representations



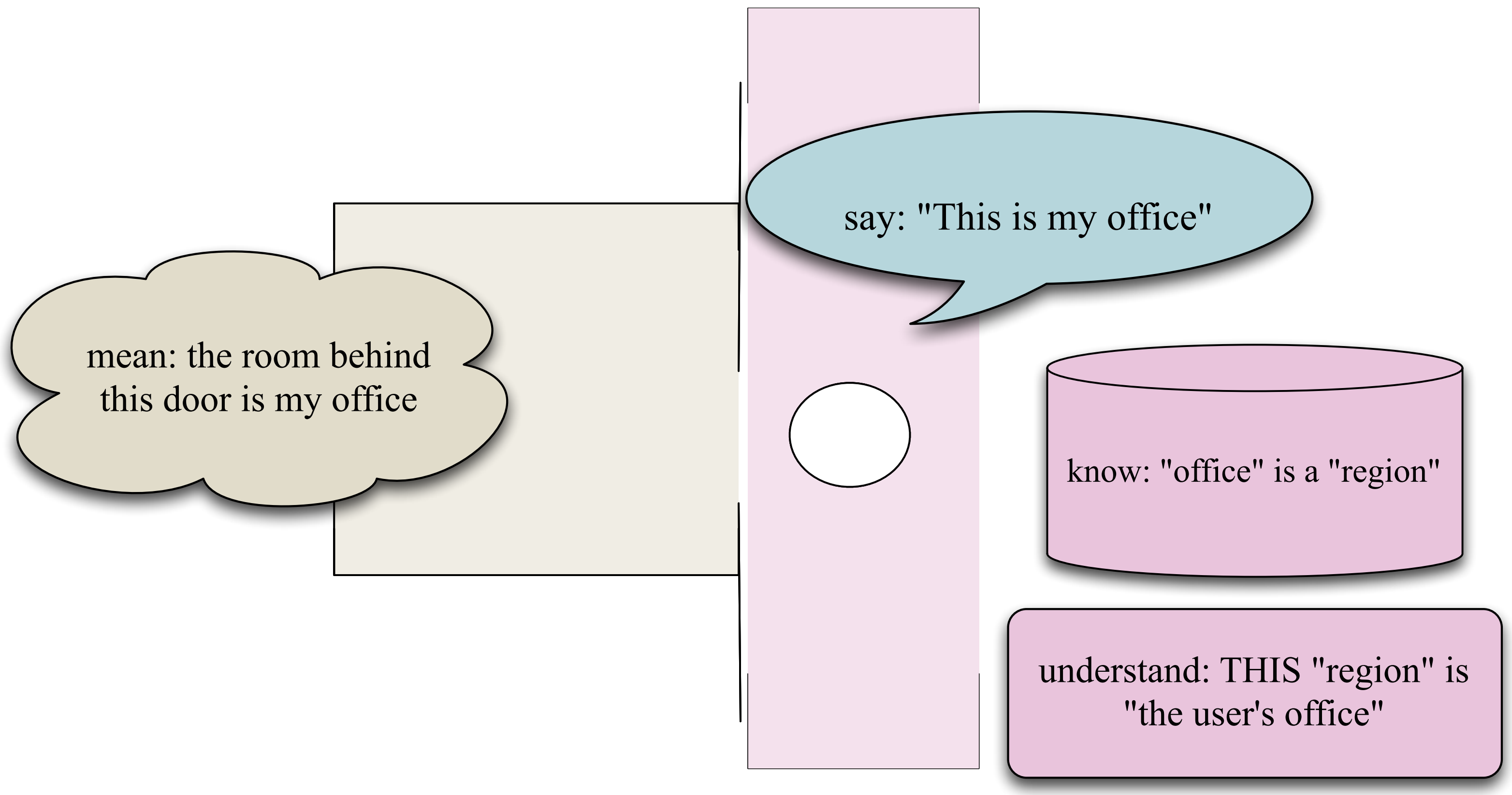
- Home tour / guided tour as initial scenario

Human augmented mapping - overview



Tracker "live" demo

What if…


say: "This is my office"
mean: the room behind this door is my office
know: "office" is a "region"
understand: THIS "region" is "the user's office"

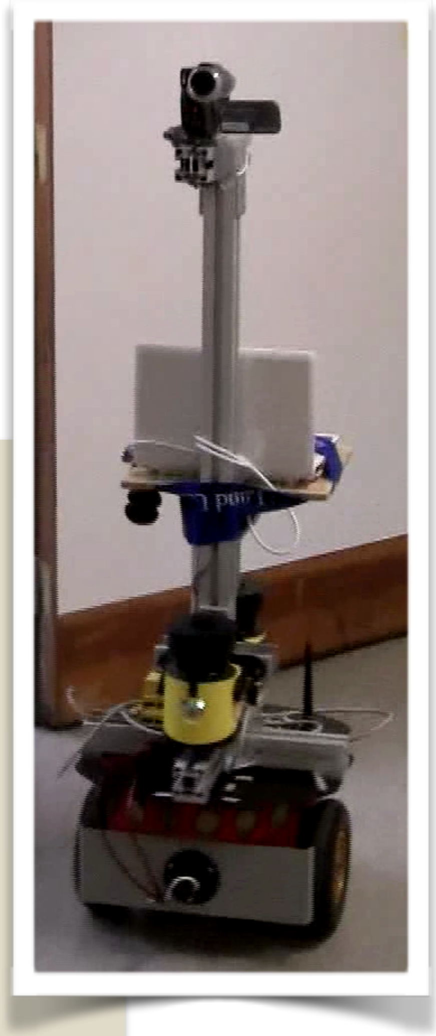
Interaction patterns?

Can we repeatedly, with several subjects, in a clearly designed set-up, observe any structure, frequent strategies, "interaction patterns", that correspond to the spatial categories *Region*, *Workspace*, and *Object* when people present an indoor environment to a mobile robot?

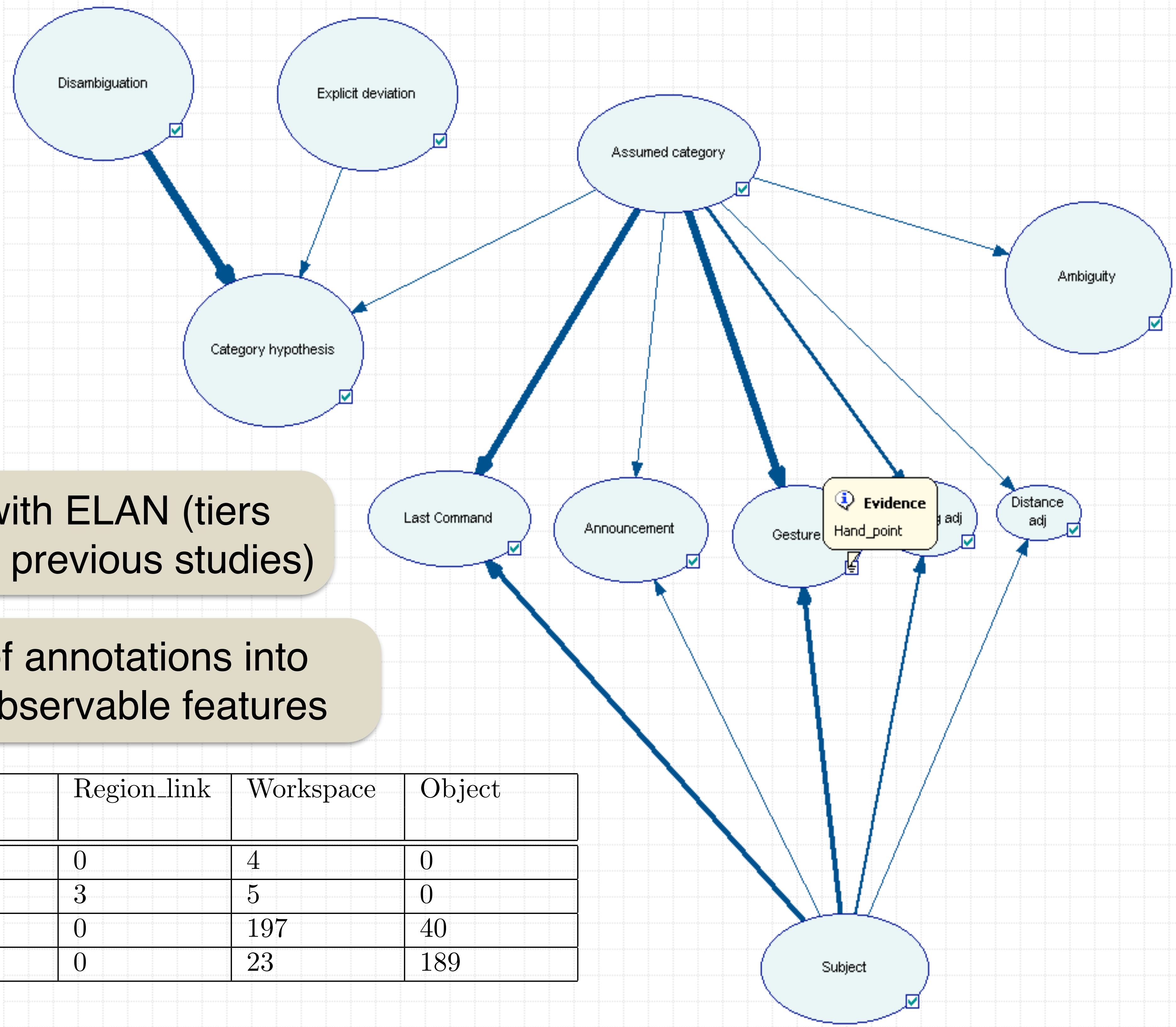
37 Participants

Guide the robot
(three rooms/regions, at least three small objects and three locations/workspaces according to suggestion list)

Video (one external camera and one on the robot) and robot sensor data were stored for later analysis.



Interaction patterns!



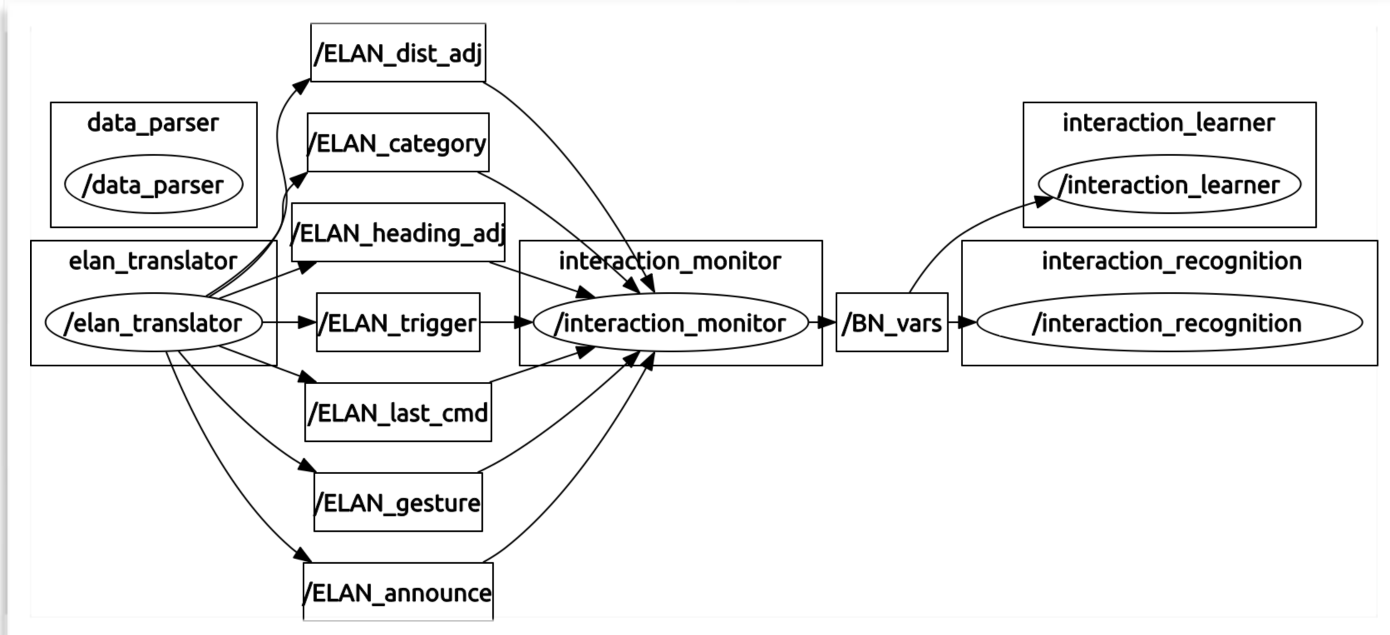
Annotation of videos with ELAN (tiers according to results from previous studies)

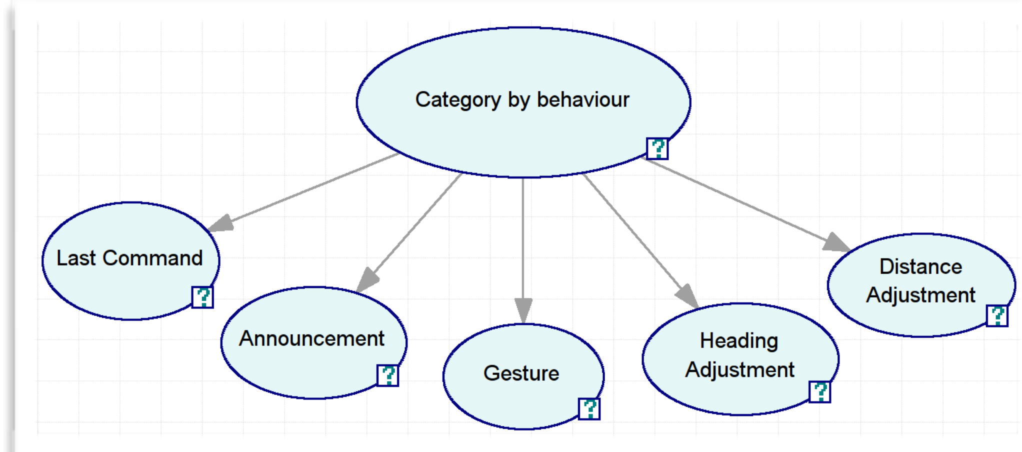
Manual summary of annotations into potentially system observable features

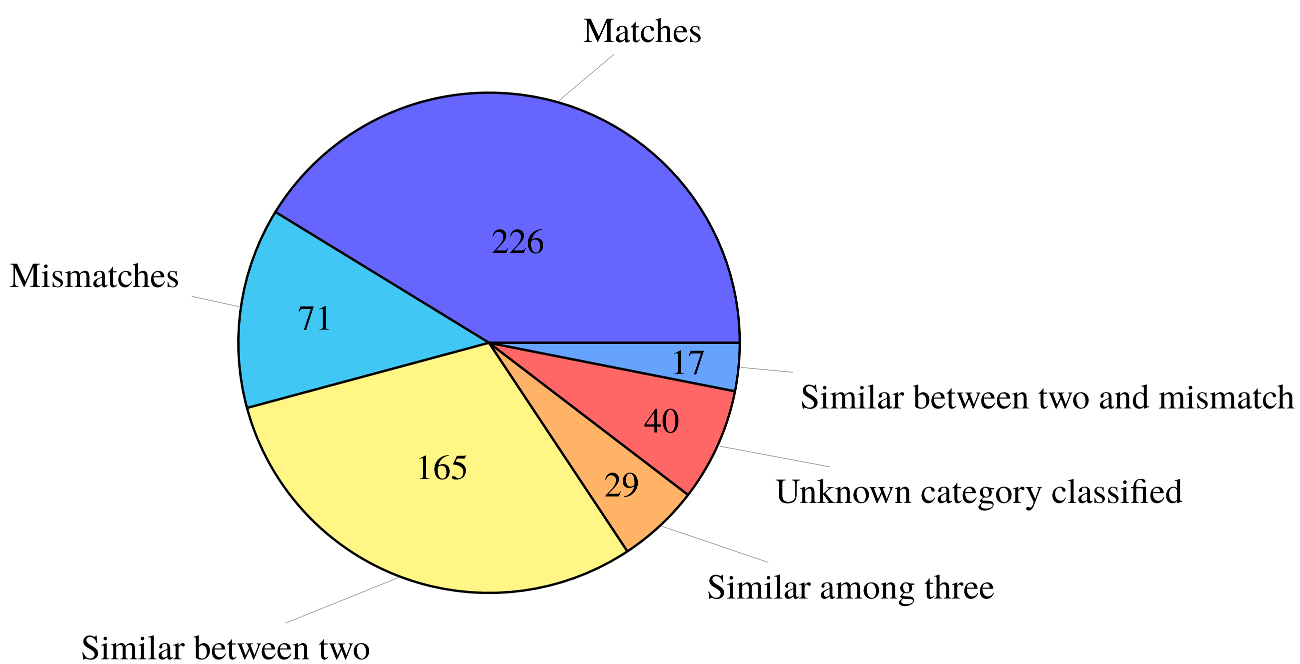
Prediction Definition	Region	Region_link	Workspace	Object
Region	62	0	4	0
Region_link	16	3	5	0
Workspace	5	0	197	40
Object	0	0	23	189

Elin A. Topp, "Interaction Patterns in Human Augmented Mapping"
Special Issue on Spatial Interaction and Reasoning for Real-World Robotics, RSJ Advanced Robotics, vol 5, issue 31, March 2017

Automated detection and identification







71 clear mismatches:

40 *objects -> workspace* (mostly chairs)

17 *workspaces -> region*

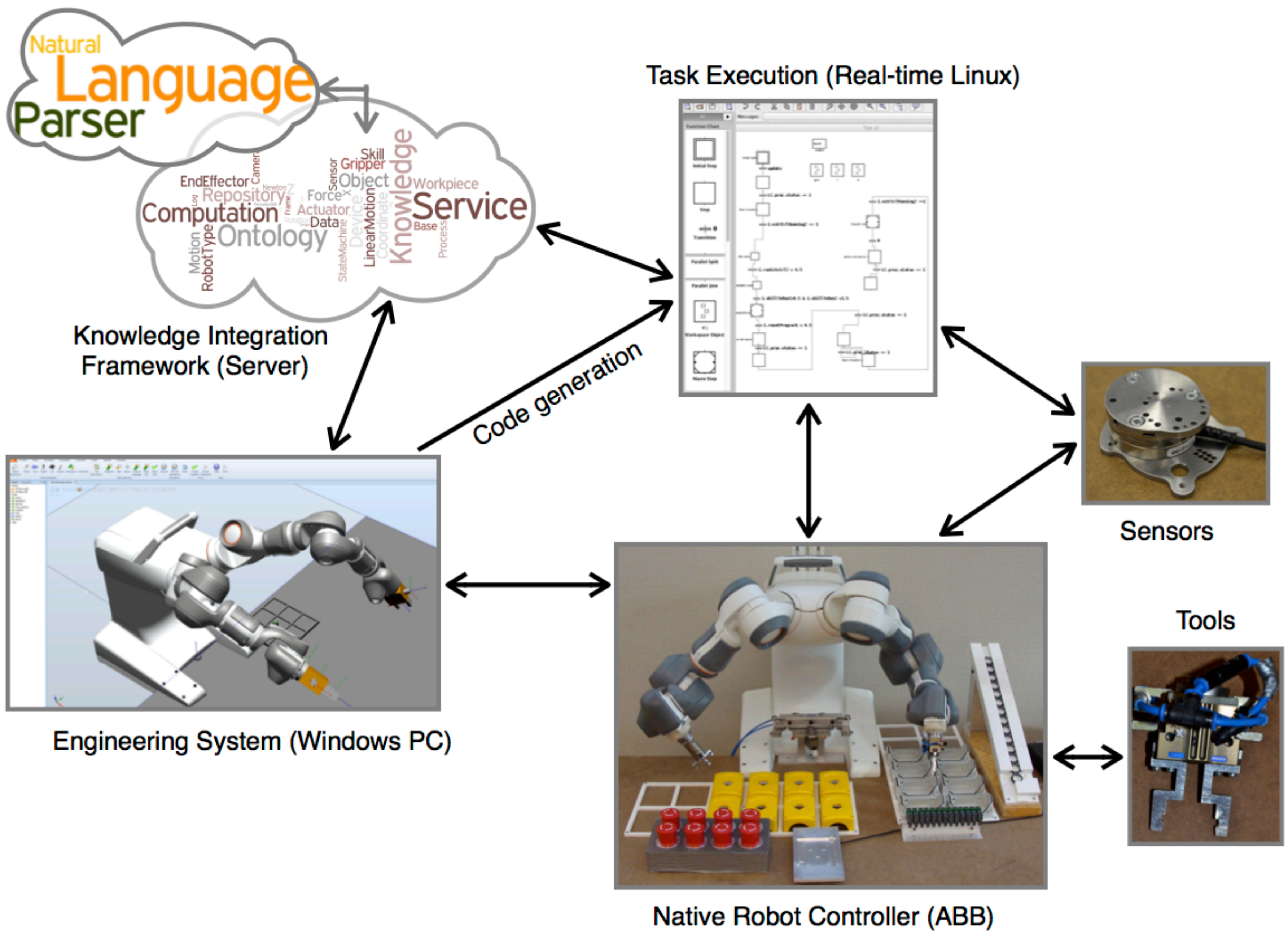
6 *regions -> workspace*

(Felip Martí Carillo and Elin A. Topp,
"Interaction and Task Patterns in Symbiotic, Mixed-Initiative Human-Robot Interaction",
AAAI-WS on Symbiotic Cognitive Systems, February 2017, Phoenix, AZ, USA)

NLP-based programming

(Maj Stenmark, 2013)

The AI-bits behind…



(Maj Stenmark, 2014)

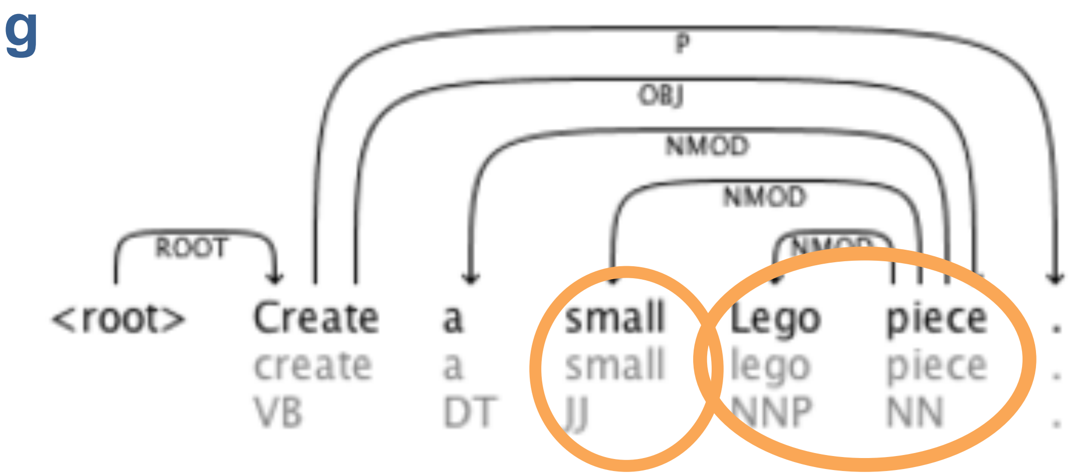
NLP-based programming

	Create	a	small	Lego	piece	.
create.01		A1				

Predicate-argument structures

	Create	a	small	Lego	piece	.
create.01		A1				

Map to existing commands or programs



Skills and knowledge

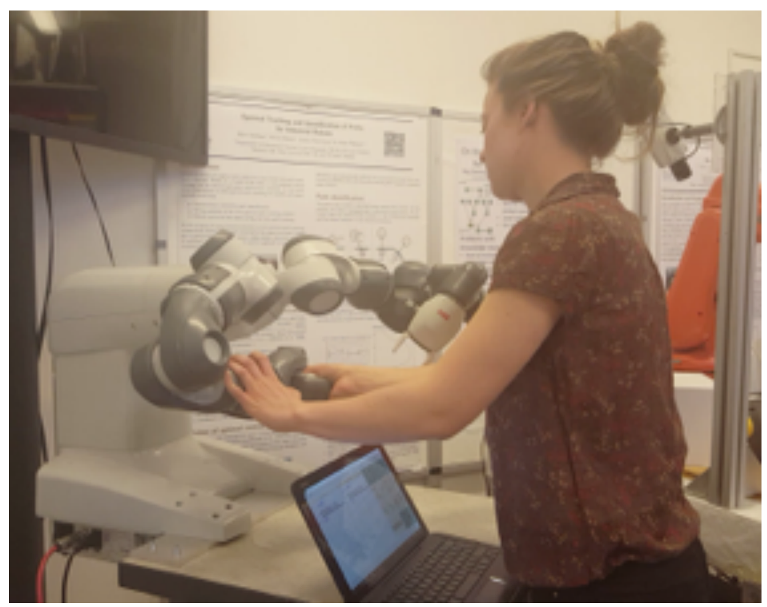
Devices

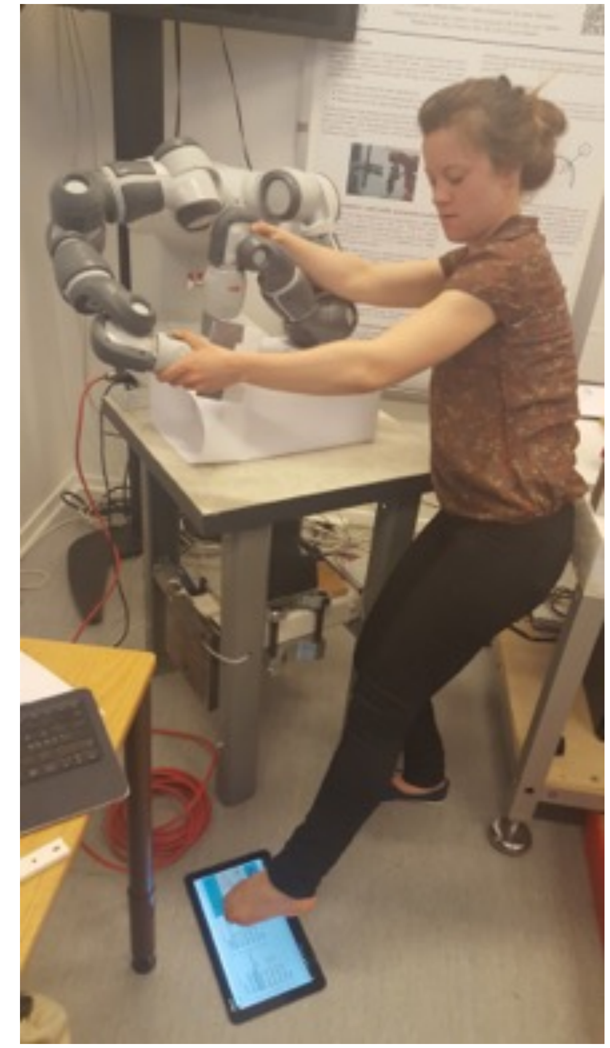
- PhysicalObject
  - Device
    - Actuator
      - Motor
    - Communication
    - CommunicationPort
    - CompoundDevice
    - Computer
    - Controller
    - EnvironmentDevice
    - ManipulationAndHandlingDevice
      - Displacement
      - Fixture
      - Gripper
      - Robot
        - ArticulatedRobot
        - CartesianRobot
        - HexapodRobot
        - MobileRobot
        - ParallelKinematicRobot
        - ScaraRobot
        - SimpleKinematicRobot
        - SpecialKinematicRobot
      - ToolChanger

Skill types

- Skill
  - AdditionalFunction
  - DiagnosticFunction
  - MainFunction
    - LightingFunction
    - ManipulationAndHandlingFunction
      - Move
      - Secure
        - Attach
        - ChangeTool
        - Clamp
        - Detach
        - Grasp
        - Release
        - Unclamp
    - ManufacturingFunction
    - OpticFunction
      - AcquireImage
      - Focalize
    - Processing
      - ProcessImages
    - SensorFunction

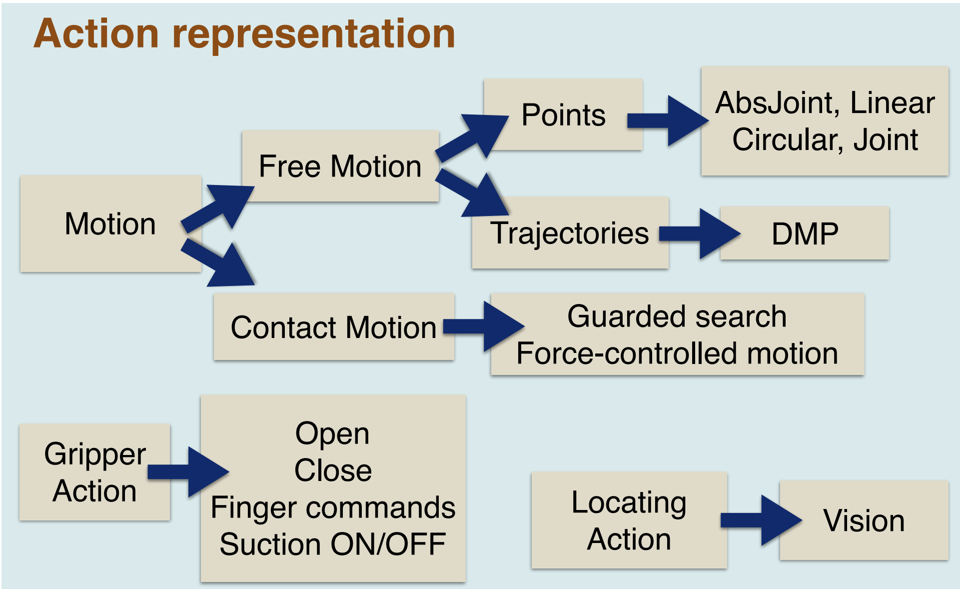
However …

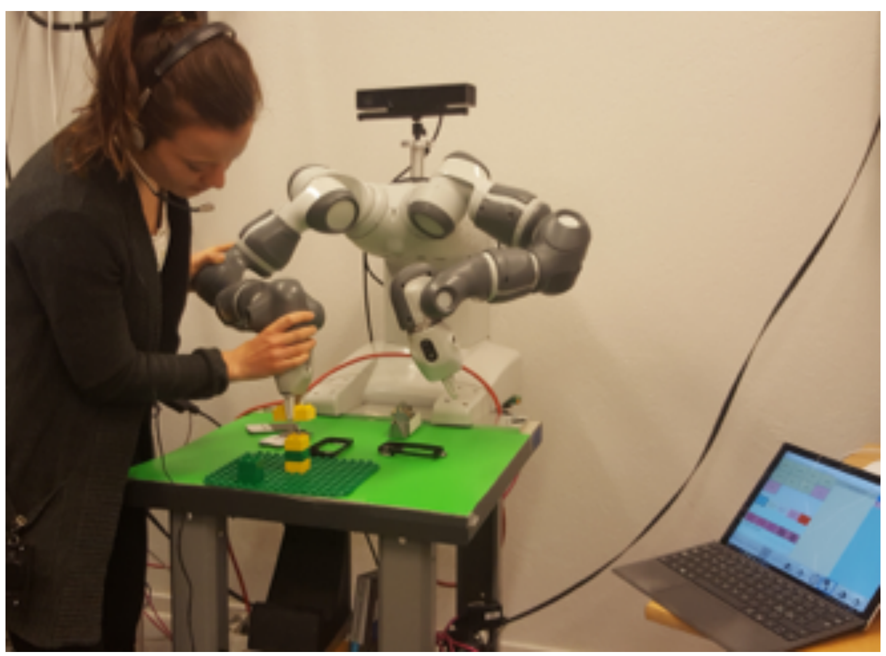


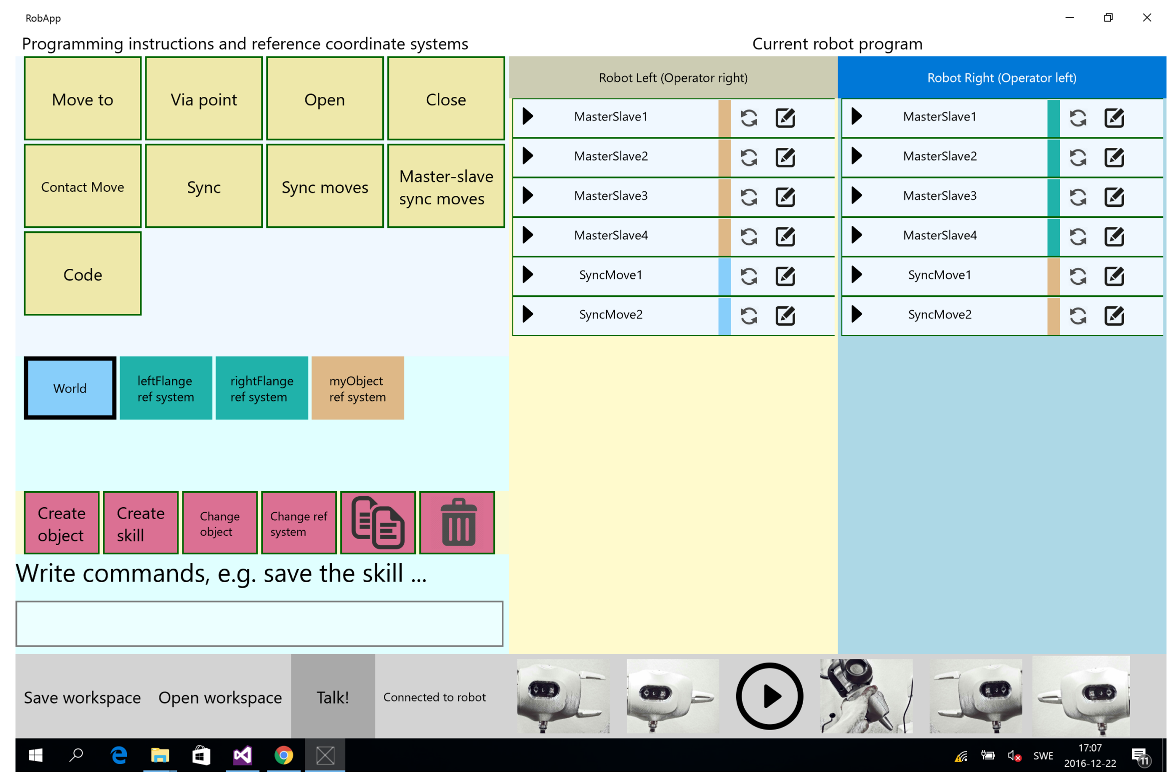


Even though the robot has lead-through built in, and
even though we could use NLP and high-level instructions to
make use of our skill representation -

… we must get the skills into the system!

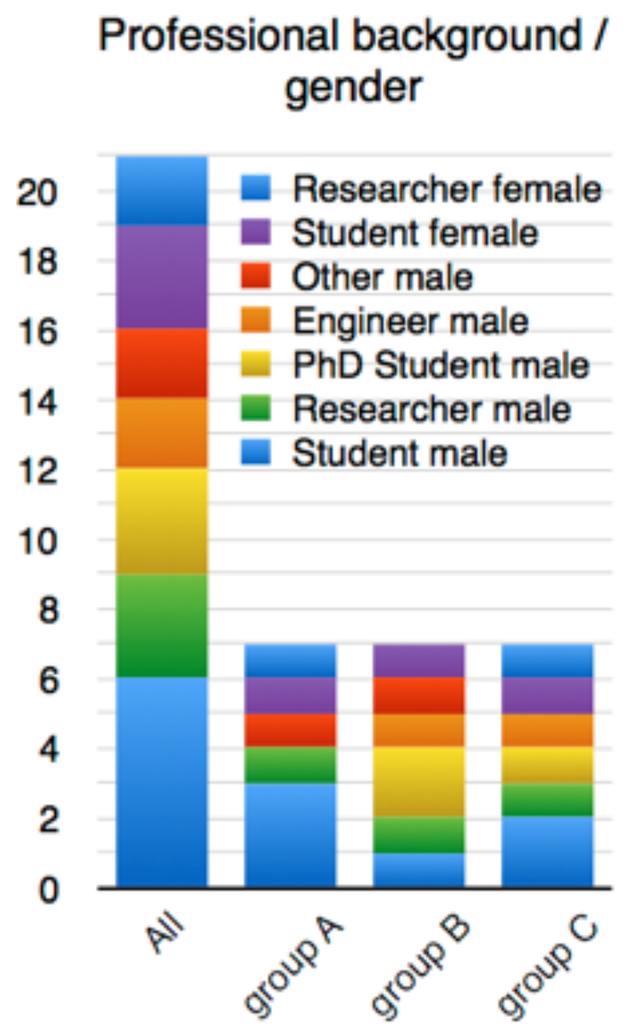


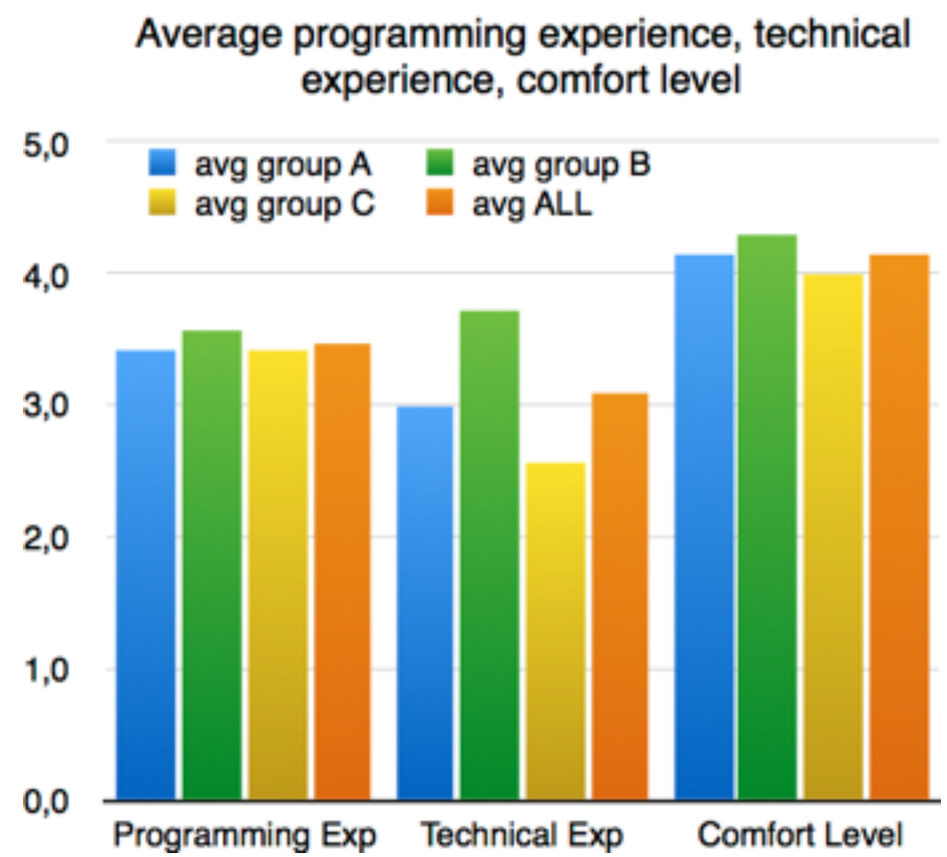


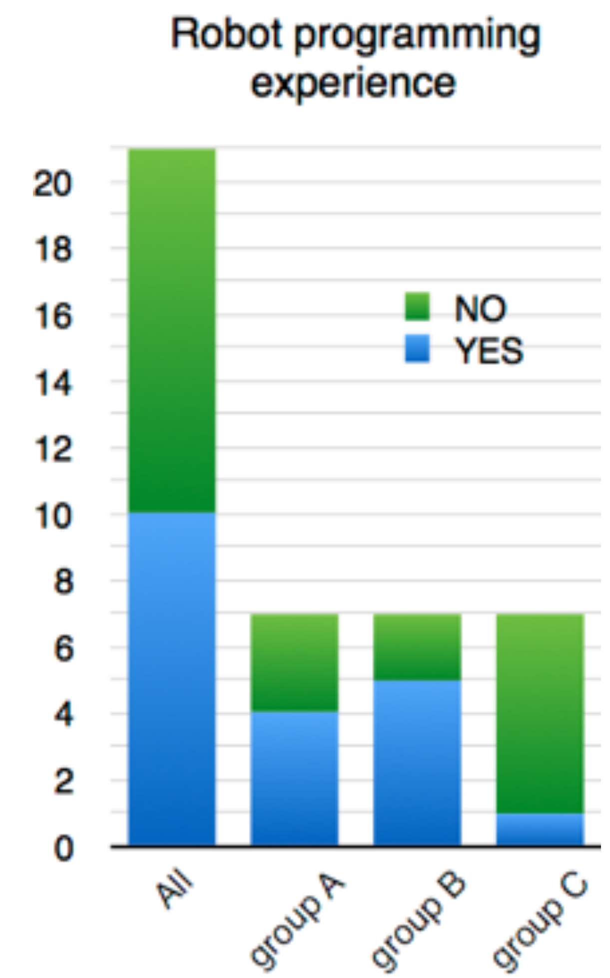


(Maj Stenmark, Mathias Haage, Elin A. Topp, and Jacek Malec,
"Supporting Semantic Capture during Kinesthetic Teaching of Collaborative Industrial Robots",
ICSC-IW on Semantics in Engineering and Robotics, January 2017, San Diego, CA, USA)
(Maj Stenmark, Mathias Haage, Elin A. Topp, and Jacek Malec,
"Making Robotic Sense of Incomplete Human Instructions in High-Level Programming for Industrial Robotic Assembly",
AAAI-WS on Human-Machine Collaborative Learning, February 2017, San Francisco, CA, USA)

Does skill re-use help?
Can non-experts program the robot?







Two phases: I: Step 1 (create "pick up and insert a 2x2 Duplo on another one" - skill) and
II: Steps 2-4 "repeat" Step 1 (different conditions) with a 2x4 Duplo

Three Conditions: A: re-use your step 1 skill
B: re-use a provided, expert-made skill
C: build everything from scratch

Yes! and Yes!

Research video, user study

Kindergarden teacher programs YuMi

Maj Stenmark, Mathias Haage, and Elin A. Topp,
“Simplified Programming of Re-usable Skills on a Safe Industrial Robot - Prototype and Evaluation”,
ACM / IEEE Conference on Human-Robot Interaction, March 2017, Vienna, Austria

Robotics and Semantic Systems @CS

- Master's projects (Ex-jobb)
  - Internal (research oriented) or external (industry related)
  - International
- Lab visit to the Robotlab in M-huset
- Contact us: Jacek, Pierre, Elin or other members of the group: Klas Nilsson, Mathias Haage, Sven Gestegård Robertz
- Course EDAN70, Project in Computer Science, VT2
- Course MMKN30, Service Robotics (through IKDC)