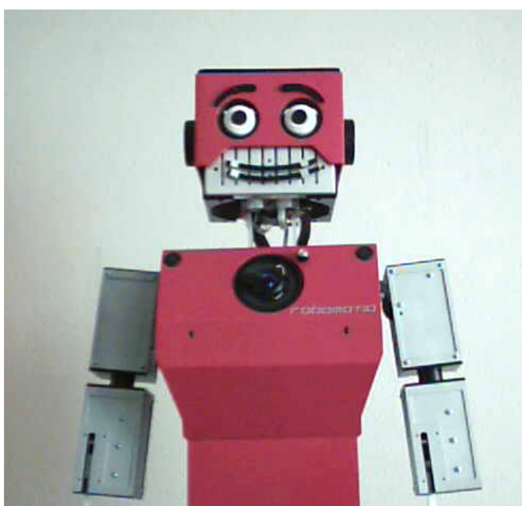
One-Shot Learning through Task-Based Natural Language Dialogues

Matthias Scheutz
Human Robot Interaction Laboratory
Tufts University
Medford, MA 02155, USA
http://hrilab.tufts.edu/



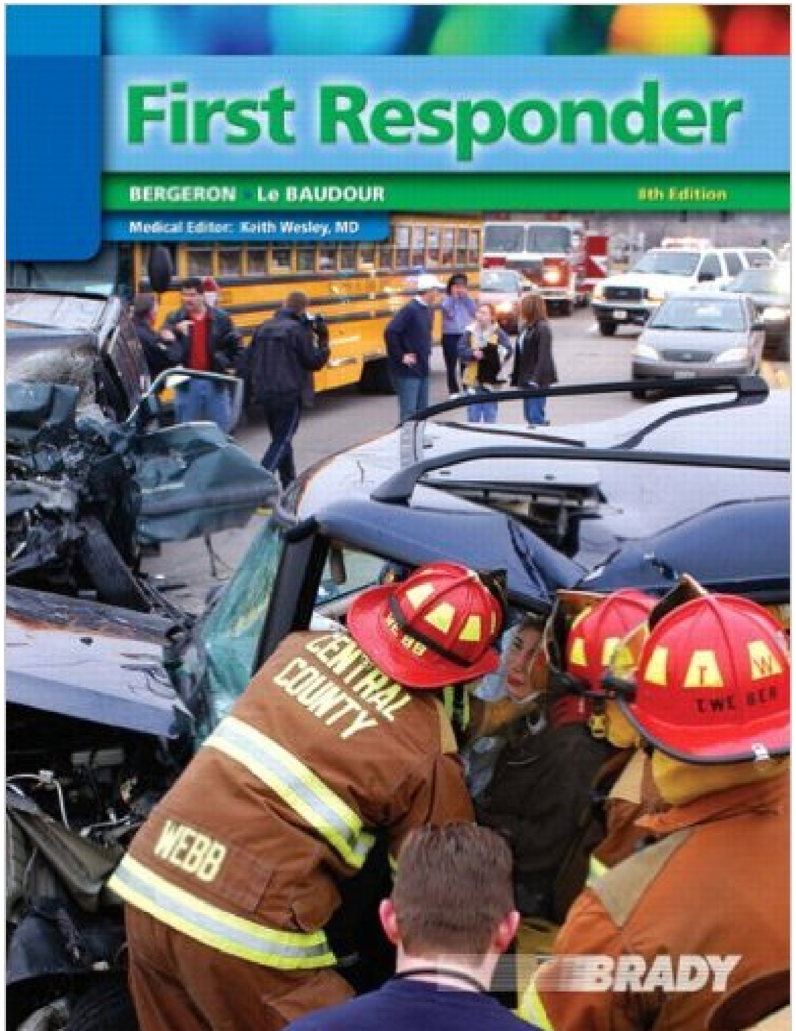
Closed-world vs. open-world tasks

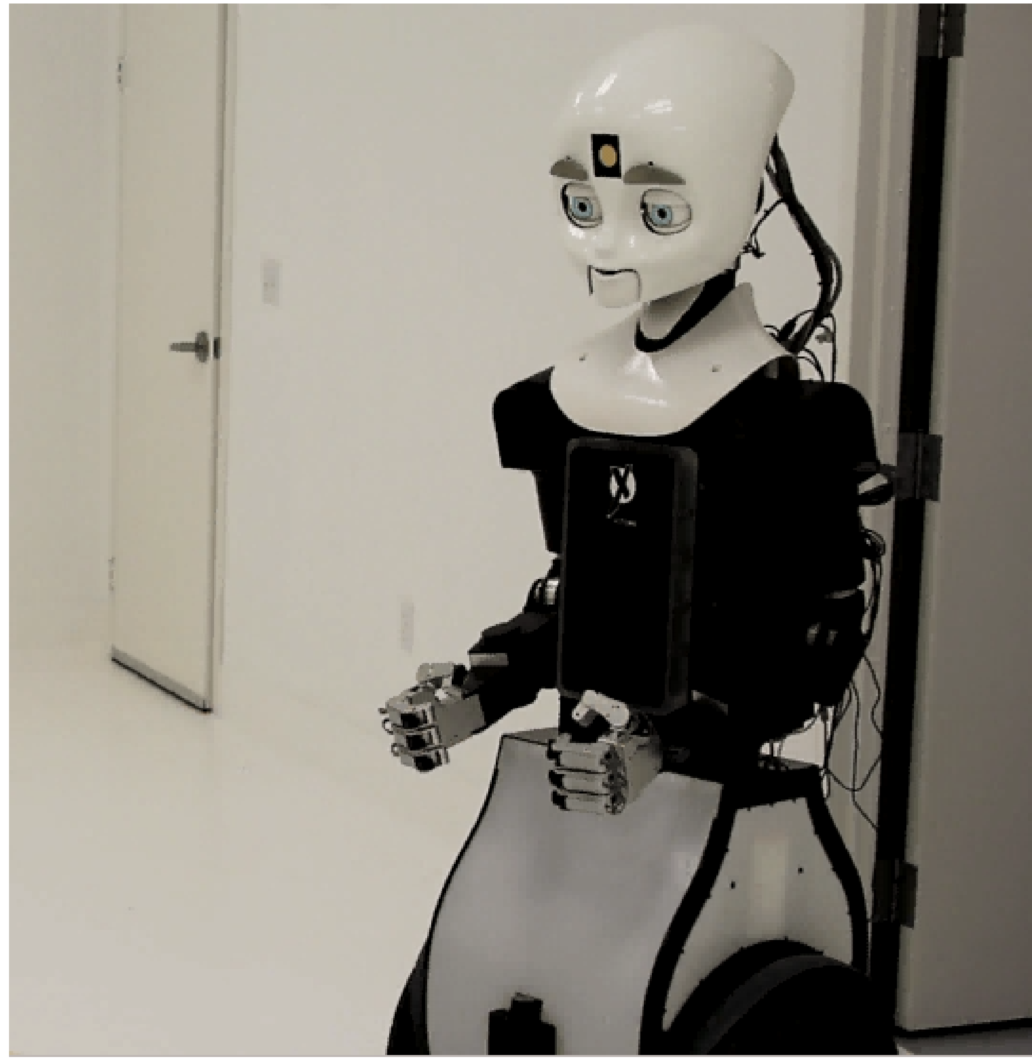
- *Closed-world* tasks: **all possible environmental, task-based, human- and interaction-oriented aspects are known at the beginning and during the task**
- *Open-world* tasks: some aspects of the **environment**, **task**, **human** and **interaction** are **unknown** at the beginning and during task execution, a prerequisite for many robotic application domains (service robotics, space robotics, etc.)
- Open-world tasks require robotic architectures that can deal with unknowns and acquire new knowledge on the fly
- Note that the robotic architecture should **not** impose any limitations on what can be acquired online (e.g., new object and event percepts, primitive actions, skills, concepts, rules, words, norms, etiquette, etc.)
- Most current robotic systems can only handle open world tasks in a very limited way

Open worlds are very challenging!

- Consider a simple task where a robot gets instructed to retrieve an object that it does not know about in an unknown environment...
  - First, how can the robot understand an instruction if it does not know the word (nor the object denoted by it)?
  - How does the robot know what to look out for?
  - And how can the robot resolve references to unknown locations?
  - Then, how can we formulate a goal that makes reference to an object whose type and location the robot does not know?
  - And when the robot finally finds the object, how will it be able to retrieve the object (e.g., pick it up and carry it)?

A motivating USAR open-world tasking example





The limits of data-driven methods

- *What learning methods can be used for open-world tasks?*
- While data-driven methods have turned out to be very successful in AI and robotics, they are based on two core assumptions that

    (1) data sets are available for training
    (2) training is not time-critical

- Neither assumption is typically met in open-world scenarios (such as Urban Search and Rescue missions), where robots must be able to acquire and use new knowledge quickly, on the fly, during task performance
- Hence, we need to *supplement* data-driven methods for cases where one of the two conditions is not applicable
- One-shot learning is an instance of such a method where an agent can learn something new from **just one exposure**

One-shot learning

- A typical version of **one-shot visual learning** would entail seeing a new object (maybe together with a linguistic label) and then being able to recognize novel instances of the same object type (e.g., from different angles and in different poses and contexts)
- A typical version of **one-shot action learning** would entail observing an action performed by some agent (maybe together with a linguistic label) and then being able to perform the action right away and, ideally, recognize novel instances of the same action
- Similarly, we can envision one-shot learning of events (involving many new objects and possibly complex actions)
- Clearly, such one-shot learning is a **very difficult problem** as it is not clear what to focus on based on only one exposure (i.e., what the essential, defining properties are of the type to be learned)

NL-guided one-shot learning

- This is where natural language (NL) can help!
- The key idea is to exploit the structured representations present in NL descriptions **to first infer intended relevant entities** such as objects, their parts and mereological relationships in object learning, actions and their parts and parameters in action learning, etc.
- Then we can build **structured representations together with processing schemes operating on them** (e.g., that allow for visual object recognition in different contexts and poses, action performance in different contexts, etc.)
- Will give three sets of examples of NL-guided one-shot learning during *task performance*:
  - one-shot object learning
  - one-shot action learning
  - one-shot rule learning

One-shot object learning

A robot is located in front of a table with several new objects with a human instructor :

H: Do you see a medkit?
R: What does a medkit look like?

The robot does not know what a medkit looks like, henc it immediately asks and gets an answer from the human instructor:

H: It is a white box with a red cross on it.
R: OK.

After having received the definition of “medkit” and been able configure and complete a visual search, the robot now knows how to recognize medkits (despite the various distractor objects).

H: Now do you see one?
R: Yes.

And when the medkit is moved to a different location in a different pose, the robot can still recognize and point to it

H: How about now?
R: Yes.
H: Where is it?
R: It’s over there.

One-shot object learning

Extracting object descriptions

- From the natural language expression

  "a white box with a red cross on it"

  we can generate the logical expression

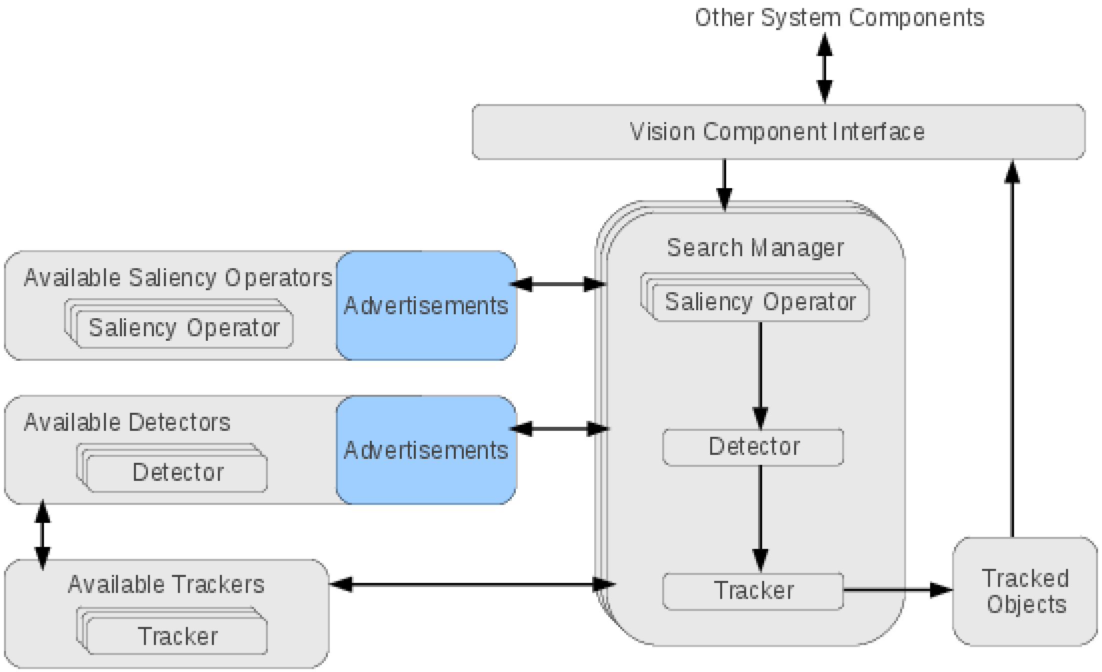
  $$\mathbf{color}(x,\text{white}) \wedge \mathbf{type}(x,\text{box}) \wedge \mathbf{color}(y,\text{red}) \wedge \mathbf{type}(y,\text{cross}) \wedge \mathbf{on}(y,x)$$

- Then we search for perceptual systems in the robotic architecture that have detectors for colors, as well as objects/textures of type "box" and "cross"

- Note that the detectors might exist in different subsystems (e.g., a "box" detector might be offered by 3D sensors generating point clouds, while color and textures might be obtained from RGB cameras)

How to handle unknown objects

- Once the type “medkit” is defined, objects of the type “medkit” can be searched for by looking for white boxes and red cross and checking whether the constraint “**on**(*y*,*x*)” is satisfied
- Note the **ambiguity** between 2D textures and 3D objects which cannot be resolved based on the linguistic expression alone...



architecture of the vision system for online learning of new object detectors combining available saliency operators and feature detectors

One-shot action learning

- Multiple ways in which we could learn actions online (e.g., from physics simulations, human demonstrations, online crowdsourcing, explicit instructions, etc.)
- Have explored three ways of doing that:
  - explicit verbal instruction (can be spoken or written and does not require presence of a human instructor)
  - explicit verbal instruction with human demonstration (parts of the instruction can only be understood by watching the human perform the action)
  - online generalization and adaptation of already known actions to a new context (i.e., environmental setting, objects, constraints, etc.)

One-shot action learning

- Suppose we were to get the following explicit instruction:

  To follow means that you should stay within one meter of me.

- Then we can generate a control law represented as **action script**:

  follow(*x*:actor,*y*:actor)
    maintain(
      <(distFrom(*x*,*y*),1)
)

  and immediately run it using the architecture's "Action Interpreter"

One-shot action learning

The robot is asked to perform a task it does not know how to perform, which it notices.

H: Aandy, please pick up the medkit.
A: I do not know how I pick the medkit up.
H: Let me show you.
A: Okay.

The instruction begins using a simple dialogue paired with visual demonstrations that allow the robot to resolve indexicals like “this” as well as positions.

H: Put your hand above the medkit with your palm down like this.
A: Okay.
H: Close your hand.
A: Okay.
H: Lift your hand.
A: Okay.

During the instructions, the robot pays attention to each step and records it.

H: And that's how you pick it up.
A: Okay.

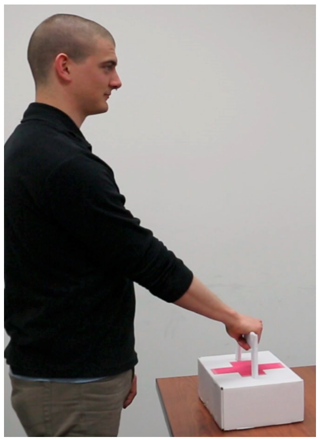
The robot has learned the activity and is immediately able to perform it when asked.

H: Now you try.
A: Okay.

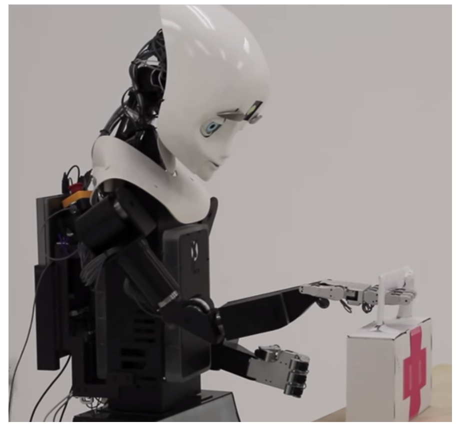
One-shot action learning

One-shot action learning

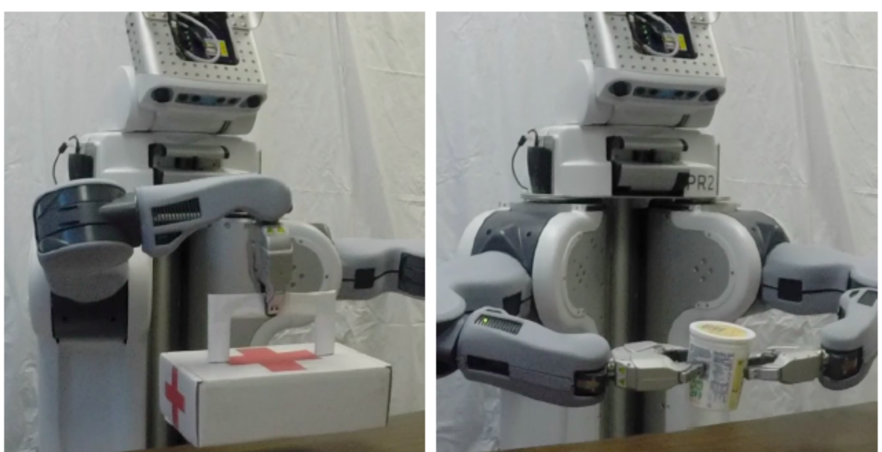
Human demonstrates picking up a



Robot learns to pick up a medkit

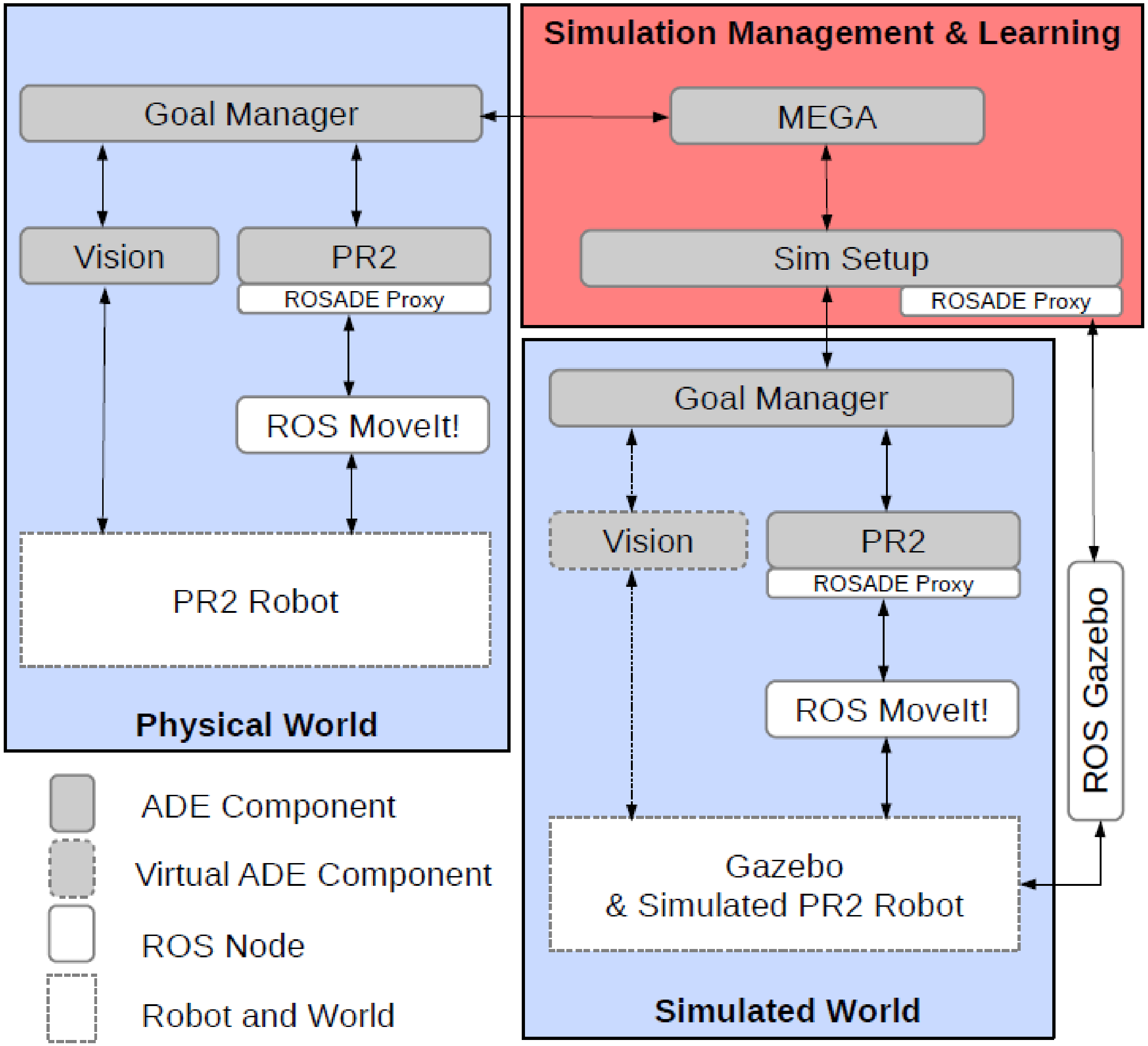


Robot uses same learned action to pick up a briefcase

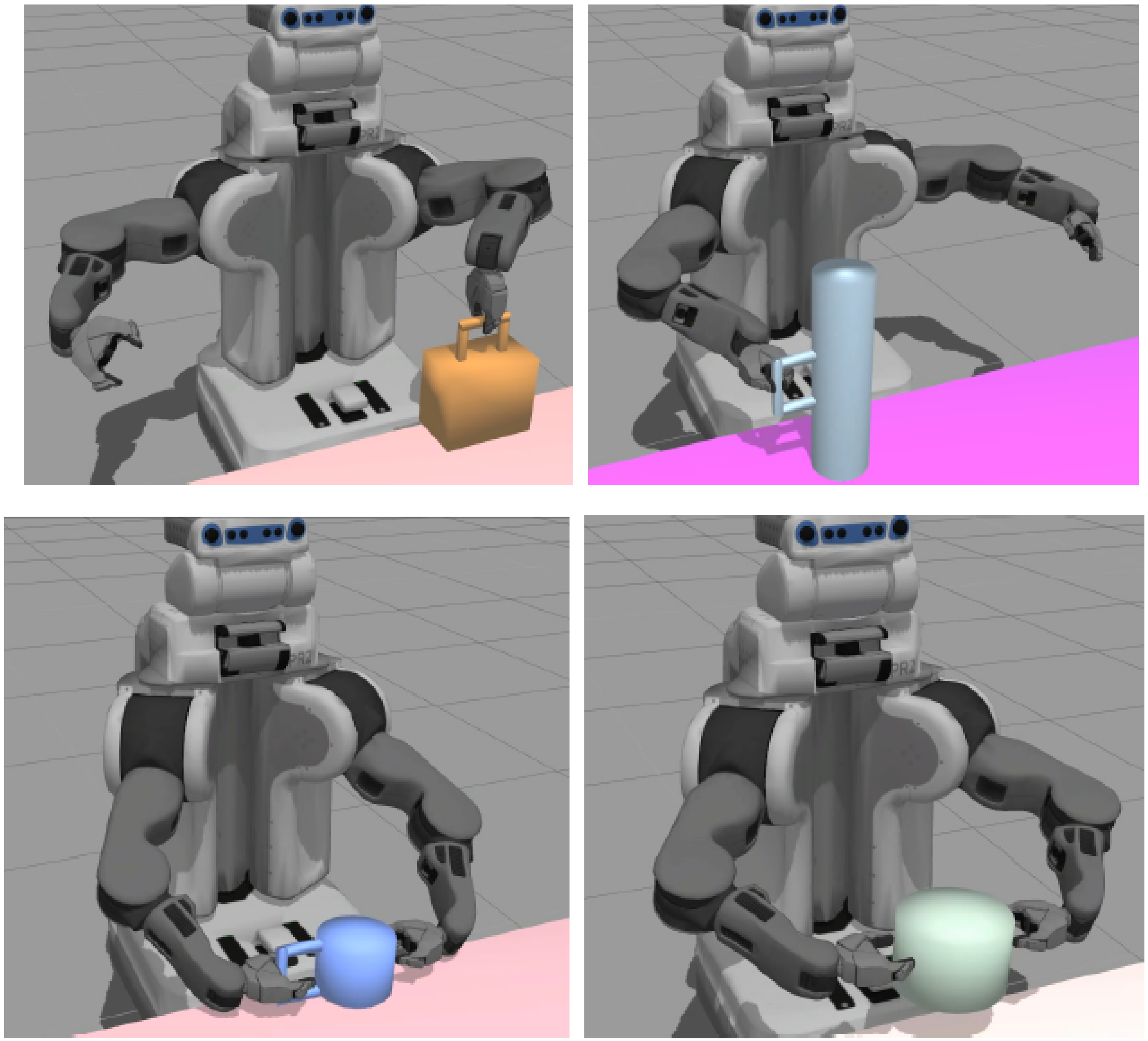


Robot learns a modfied action to pick up a tub with both arms using mental simulation

One-shot action learning

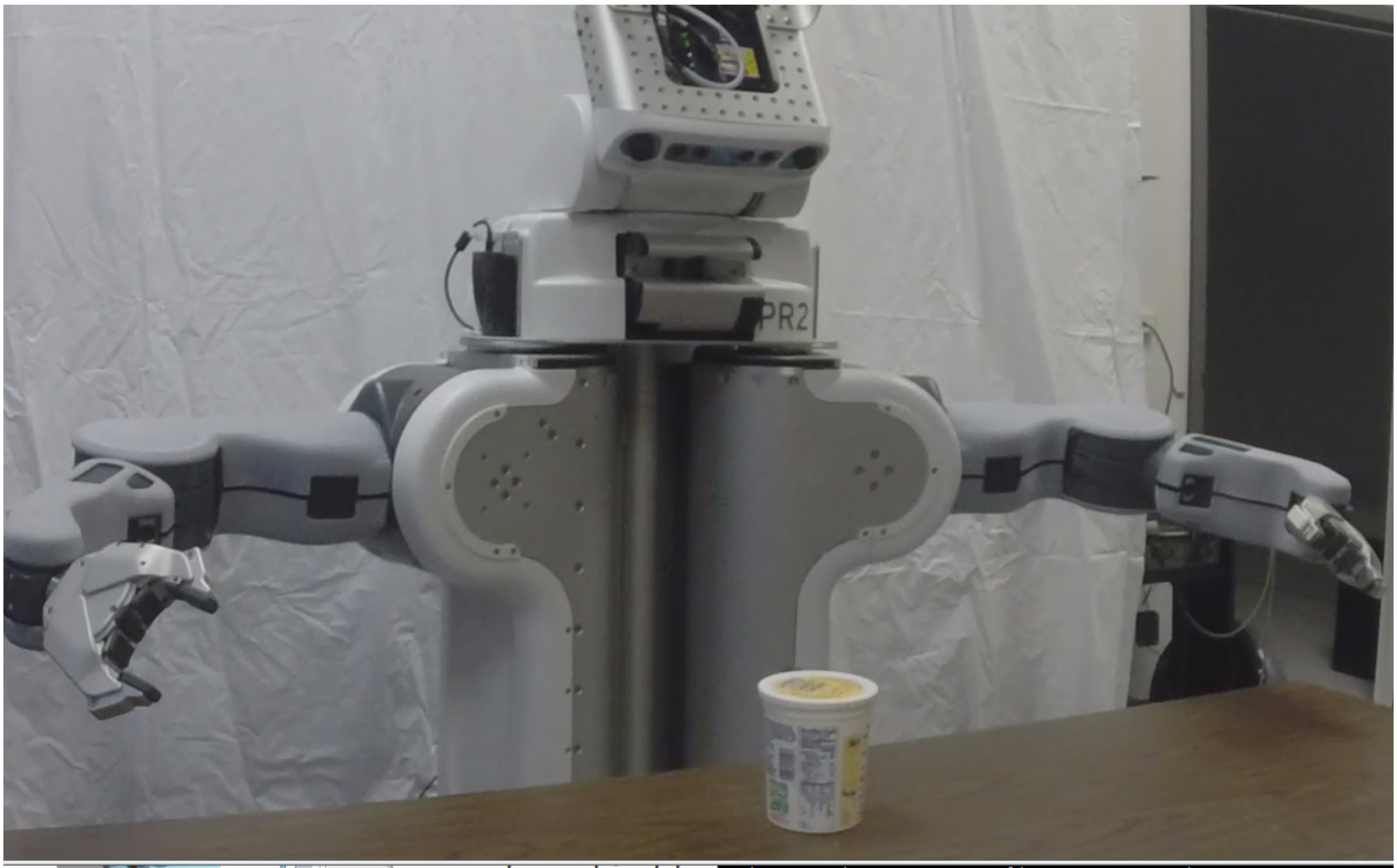


One-shot action learning



One-shot action learning

The generated scripts resulting from object and action generalizations are then immediately executable on the robot



One-shot rule learning

The robot is driving down the hallway as part of its task.

H: What are your orders?
R: I have been ordered to listen for instructions, to go to the end of the hallway, and try to look for a red box.

The robot enters the next room with an open door, searches the room and exits without finding a red box. The robot then drives by a closed door.

H: To go into a room when you are at a closed door, push it one meter.
R: Understood.

At this point, the robot has learned a new plan operator, added the operator to the planner, and instructed the perceptual system to look out for closed doors (in addition to open doors). When the robot then encounters the next room with a closed door, it pushes the door open, enters the room, starts to search it and finds a red box which it reports.

R: There is a red box in Room 1.

One-shot rule learning

Summary and open problems

- Open-world tasks require significant changes to algorithms in almost all components of a robotic architecture
- Integrated one-shot learning mechanisms allow for fast, online acquisition and immediate use of new knowledge, which was demonstrated for one-shot perception, action, and rule learning
- We are currently extending various parts of the architecture to be able to better handle the various uncertainties involved in open-world scenarios
- Open problems include how to learn object manipulations from observations, instructions, and simulations, how to learn complex activities from narrated demonstrations, and overall how to learn and perform team tasks in open worlds in mixed-human robot teams

Acknowledgments and links

- Former students/postdocs: Gordon Briggs (NRL), Rehj Cantrell (Nuance), Matt Dunlap (Charles Rivers), Paul Schermerhorn (Soar Tech), Richard Veale (NIPS Japan)
- Current students/staff: Evan Krause, Brad Oosterveld, Tom Williams, Jason Wilson, Vasanth Sarathy, and various undergraduate students at Tufts
- Collaborators: Rao Kambhampati's group at ASU (open-world planning), and Markus Vincze's group at Vienna (robot vision)
- Funding provided by the US National Science Foundation and the US Office of Naval Research through several grants

- Papers at http://hrilab.tufts.edu/publications/
- Demos at https://www.youtube.com/user/HRILaboratory

References

- We have been working on various aspects of open-world human-robot interactions:
  - **online tasking** through natural language dialogues with new goals involving unknown objects (e.g., ACS 2013)
  - **open-world planning** (e.g., AAAI 2012, ICRA2012)
  - **open-world reference resolution** (e.g., AAAI 2013, IROS 2015, AAAI 2016, HRI 2016)
  - **online learning of new perceptions** (e.g., AAAI 2014)
  - **online learning of new primitive and complex actions** (e.g., RoMan 2012, Cyber 2014)
  - **online learning of new rules and plan operators** (e.g., HRI 2012)
  - **online generalization of learned actions** (e.g., AAMAS 2016)