P1872-1 Robot Task Representation

Chair: Stephen Balakirsky
Stephen.Balakirsky@gtri.gatech.edu

Co-Chair: Craig Schlenoff
Craig.Schlenoff@nist.gov

IEEE Program Manager: Christy Bahn
C.Bahn@ieee.org
Why Another Standard?

- Goal: to develop a knowledge representation that addresses robot task structure, with decomposition into subclasses, categories, and/or relations. It includes attributes, both common across tasks and specific to particular tasks and task types.

- Purpose: to provide a common way of representing tasks across domains and sub-domains, addressing both the relationships between tasks and platforms and the relationships between tasks and users.

- What does this really mean?
  - Would like to task an automated system (i.e. robot) in a similar fashion that you would any human shop.
  - Provide task description, performance characteristics, constraints.
  - System understands if it is capable of accomplishing task.
  - Multiple systems from multiple vendors able to receive same task description.
Use Case: Design New Action From Existing Actions

• Desire to create a new robotic behavior without any robot programming

• Assume that a basis set of commands exist:

• Assume that a database that contains matched tokens and values exists
  - Named location has pose
  - Named offset has 6-DOF values

• Allows for the creation of new behavior by instantiation of commands
  - Goto(robot_1, flask_1, 1cm_above)
  - Goto(robot_1, flask_1, none)
  - Pipette(robot_1, pipette_1, acquire_sample)
• Constraints: The previous slide’s tasks can be executed through many different algorithms
  - How accurate does the goto need to be?
  - How fast?
  - Linear motion or uniform joint velocity?
  - ...

• Results: How do I know that the action was successful?

• Resources: What is required to perform this task?
  - Can it be specified ahead of time so that we know if a system is able to perform it?

• How do I have a common understanding of all of the above?
Common Understanding: Relationship to Other Standards

- **Upper Ontology**: Specifies the meaning of high-level, general terms that are applicable across all of the robotics domain.

- **Middle Ontology**: Specifies the meaning of ubiquitous terms are applicable to many robotics domains. Specializes concepts in the upper ontology.

- **Domain Ontology**: Specifies the meaning of terms that are primarily relevant to a specific domain. Specializes concepts in the middle ontologies.
Task Structure

Constraints specify limitations on actions that are performed in the completion of the task.

The Command Frame defines the job to be done.

Resources present the system with hardware/software resources that may be utilized in the performance of the task.

Actions and Sequencing Frame defines the particular way the job must be accomplished.

Results Frame defines how the robot’s performance of a task is evaluated.
Mission
Autonomously populate the empty wells of a biofluidic device with an organoid.

High-Level: A robotic arm retrieves an organoid from a “flask” (petri dish) and places that organoid in an empty well on a “device”. This process repeats until no empty wells remain.

Low-Level:
Retrieve Organoid From Plate:
• Go to above flask → run organoid detection algorithm → visual servo → go to plate surface → pipette up → go to above plate

Place Organoid In Device:
• Go to above device → go to above empty well → go to empty well surface → pipette down → go to above device

The UR3 using a machine vision algorithm to visual servo and retrieve an organoid from a petri dish (“flask”)
Status Update

- Officially became working group: February 2017
- Number of voting members: 20
- Total membership: 52
- Working status:
  - Outline for document has been created
  - Leadership assignments have been made for sections of outline
  - Bi-weekly meetings to discuss each section and begin writing formal document
- Target date for standard submission: Draft published December 2019