2014 IEEE International Conference on Robotics & Automation (ICRA)
Hong Kong Convention and Exhibition Center
May 31 - June 7, 2014. Hong Kong, China

DARPA Robotics Challenge: Towards a User-Guided Manipulation Framework for High-DOF Robots

Nicholas Alunni*, Halit Bener Suay*, Calder Phillips-Grafflin*, Jim Mainprice*,
Dmitry Berenson*, Sonia Chernova*, Robert W. Lindeman*, Daniel Lofaro†, and Paul Oh†
* Worcester Polytechnic Institute, Worcester, MA, 01609, USA
† Drexel University, Philadelphia, PA, 19104, USA

***Abstract*— Supervision and teleoperation of high degree-of-freedom robots is a complex task due to environmental constraints such as obstacles and limited communication, as well as task specific requirements such as using more than one end-effector at the same time. In this work we present a supervision and teleoperation framework that allows an operator to see the surroundings of a robot in 3D, make necessary adjustments for a dual or single arm manipulation task, preview the task in simulation before execution, and finally execute the task on a real robot. The framework has been applied to the valve turning task of the DARPA Robotics Challenge on the PR2, Hubo2+, and DRCHubo robots.**

High degree-of-freedom (High-DOF) robots requires solving a large set of problems in designing interfaces and systems in a way that will work reliably in real-world conditions. Some of the challenges are: decreasing the cognitive load of the operator(s), handling limited bandwidth communication links, planning in an unstructured environment, assuring the safety of the system, and error-handling. By creating a framework that can operate with real-world challenges and perform industrial manipulation tasks, we hope to decrease robotic response time in disaster situations and mitigate the risk to human disaster relief personnel. [1]

We address these problems with a framework [1], based on ROS[2], that lets an operator see the surroundings of a robot in 3D, find and select an object for manipulation, plan for a dual or single arm manipulation task, preview the task in simulation before execution, and finally execute the task on a real robot while using the graphical user interface (GUI) to monitor for errors. Hubo-Ach [3] is used to interface between ROS and the DRCHubo robot used in the competition.

We use RViz[4], ROS's built-in visualization tool, to display the robot's full body joint states, as well as additional sensor data coming from the robot such as point clouds. Localization Markers, based on ROS's Interactive Markers [5], are used to send commands through RVIZ and interact with the robot in the GUI. In our system, Localization Markers are used to locate and define the shape of the target object in the point cloud of the robot's surroundings. A Localization Marker can either be a disk, quad, or triangle mesh, and the shape and size of the marker can be modified during robot operation. The marker can be manually aligned with a full six-DOF, or the Iterative Closest Point (ICP) algorithm can be used to automatically align the marker to a subset of the point cloud generated by the robot's sensors. Once the marker has been placed, a drop-down menu provides options such as which manipulator to use and what type of manipulation task is required (e.g. clockwise vs. counter-clockwise rotation of a valve).

After defining the object to be manipulated, the user sends a command to the Constrained Bi-Directional Rapidly Exploring Random Tree (CBiRRT) [6] algorithm which can plan whole body paths for the robot while taking into account constraints such as balance and closed chain kinematics. We use OpenRAVE [7] with the CoMPS [6] add-on for path planning. Simulations of the robot are used to preview the generated trajectories to confirm the safety of the robot and its environment. In the event of a problem, the operator can change any of the parameters and then re-generate and simulate a new trajectory before executing on the real robot. Required communication between the robot and operator is reduced by planning on the operator's workstation, and only sending the trajectory to the robot when it is ready for execution. Communication is also reduced by allowing the operator to change the transmission rate of the robot's sensors during operation, allowing limited bandwidth resources to be allocated where needed most.

REFERENCES

[1] Alunni, N.; Phillips-Grafftin, C.; Suay, H.B.; Lofaro, D.; Berenson, D.; Chernova, S.; Lindeman, R.W.; Oh, P., "Toward a user-guided manipulation framework for high-DOF robots with limited communication," Technologies for Practical Robot Applications (TePRA), 2013 IEEE International Conference on , vol., no., pp.1,6, 22-23 April 2013

[2] M. Quigley et al., "ROS: an open-source Robot Operating System," in Proc. Open-Source Software workshop of the International Conference on Robotics and Automation (ICRA), 2009.

[3] D.Lofaro,"Unified algorithmic framework for high degree of freedom complex systems and humanoid robots," Ph.D. dissertation, Drexel University, College of Engineering, Electrical and Computer Engineering Department, May 2013.

[4] D. Hershberger et al. [Online] Available: http://www.ros.org/wiki/rviz

[5] D. Gossow. [Online] Available: http://www.ros.org/wiki/interactive_markers

[6] Dmitry Berenson, Siddhartha Srinivasa, and James Kuffner, "Task Space Regions: A Framework for Pose-Constrained Manipulation Planning," International Journal of Robotics Research (IJRR), Vol. 30, No. 12, October, 2011, pp. 1435 - 1460.

[7] Rosen Diankov and James Kuffner, "OpenRAVE: A Planning Architecture for Autonomous Robotics," tech. report CMU-RI-TR-08-34, Robotics Institute, Carnegie Mellon University, July, 2008

[1]This work was supported in part by the Defense Advanced Research Projects Agency (DARPA) award #N65236-12-1-1005 for the DARPA Robotics Challenge