A Shared Autonomy Interface for Household Devices

[Extended Abstract]

Matthew Rueben
ruebenm@onid.oregonstate.edu

William D. Smart
bill.smart@oregonstate.edu

School of Mechanical, Industrial, and
Manufacturing Engineering
Oregon State University
Corvallis, OR 97331
United States

1. INTRODUCTION

As robots begin to enter our homes and workplaces, they will have to deal with the devices and appliances that are already there. Unfortunately, devices that are easy for humans to operate often cause problems for robots [3]. In teleoperation settings, the lack of tactile feedback often makes manipulation of buttons and switches awkward and clumsy [7]. Also, the robot's gripper often occludes the control, making teleoperation difficult. In the autonomous setting, perception of small buttons and switches is often difficult due to sensor limitations and poor lighting conditions. Adding depth information does not help much, since many of the controls we want to manipulate are small, and often close to the noise threshold of currently-available depth sensors typically installed on a mobile robot. This makes it extremely difficult to segment the controls from the other parts of the device.

In this paper, we present a shared autonomy approach to the operation of physical device controls. A human operator gives high-level guidance, helps identify controls and their locations, and sequences the actions of the robot. Autonomous software on our robot performs the lower-level actions that require closed-loop control, and estimates the exact positions and parameters of controls. We describe the overall system, and then give the results of our initial evaluations, which suggest that the system is effective in operating the controls on a physical device.

2. RELATED WORK

Our approach is an example of what Atherton and Goodrich call "perception by proxy" [1], where a human helps with perception problems that the robot cannot solve autonomously. Nguyen et al. [5], on the other hand, show how novel behaviors can be constructed for a robot by a human assistant. The human gives no help with perception, but constructs hierarchical finite state machines that define tasks that are subsequently performed autonomously by the robot. Srinivasa et al. [8] and Beetz et al. [2] both describe relatively complete systems (i.e., with navigation, perception, and manipulation capabilities), the former focusing on manipulating objects and doors while the latter performs a high-level task: making breakfast. Neither treats household devices or the shared autonomy approach to the extent that this work does.

3. APPROACH AND IMPLEMENTATION

Our implementation was performed on a Willow Garage PR2 robot using the ROS software infrastructure [6], the Point Cloud Library, and OpenCV. All of our software is freely available at our open-source repository.[1] Visual sensing was done using a Microsoft Kinect sensor mounted on the head of the PR2. The implementation was validated with a Yamaha RX-V390 stereo receiver and a typical household light switch, both shown in figure 1. The steps comprising a typical use case are as follows.

1. Identify and Localize the Device. The robot starts in front of the device and sends a 2D image of it to the human. The human selects the front face of the device by drawing a rectangle with a mouse. We then estimate a planar model for the part of the device selected by the human. To continue, the operator selects the device control type: switch, button, or knob.

2. Identify and Localize the Control. We currently detect switches and buttons using the human operator's input, projecting a clicked point into three dimensions using data from the robot's RGB-D camera. Knobs we assume to be cylindrical and perpendicular to the front plane of the device. Using the previously-calculated planar model and another mouse input (a rectangle around the knob) by the operator, we estimate the radius, height, and position of the knob from the point cloud generated by the robot's RGB-D sensor.

3. Perform the Manipulation. Once the position and orientation of the control are known, it can be manipulated. Buttons and switches are manipulated using a Cartesian controller that either presses or flips the control, respectively. Knobs are autonomously grasped, ensuring that the wrist roll axis aligns with the estimated axis of rotation of

HRI'15 Extended Abstracts, March 2–5, 2015, Portland, OR, USA.
ACM 978-1-4503-3318-4/15/03.
http://dx.doi.org/10.1145/2701973.2702061.

[1] `http://github.com/OSUrobotics/pr2_household_devices`

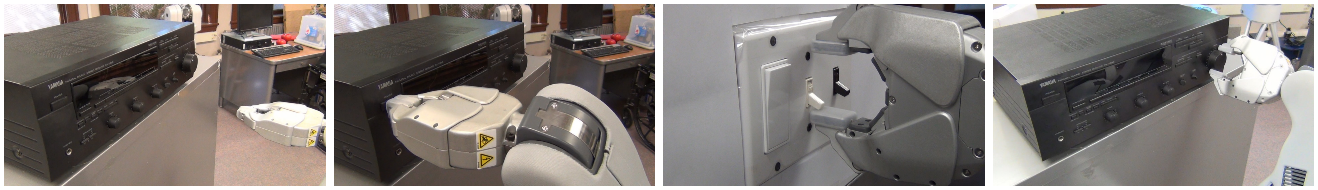

Figure 1: Still images of the ready position (far left) and three manipulations we perform on our two representative devices – a stereo receiver and a household light switch panel.

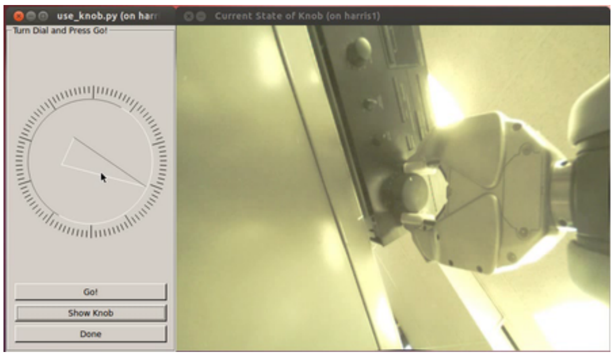

Figure 2: The graphical interface used to select knob positions. The image on the right is from a camera in the forearm of the robot, which typically affords the best view of manipulations.

the knob. Then, a graphical interface, shown in figure 2, is presented to the human operator, allowing them to select the final position of the knob using images from the PR2's forearm camera.

4. INITIAL EVALUATION

Members of our research group tested our shared-autonomy interface against the direct teleoperation interface described by Ciocarlie et al. [3]. Users were asked to use both interfaces to perform the operations shown in figure 1.

We found that people were able to complete the button and switch tasks in approximately half the time with our shared autonomy interface as compared to direct teleoperation. In the button-pushing task, users needed fewer attempts to successfully complete the task using our shared autonomy interface. Misses with our interface were usually due to calibration errors. Our shared autonomy interface performs better because users indicate the position of the control before it is obscured by the end effector.

While less accurate for the button-pushing task, the direct teleoperation interface proved to be competitive with the shared autonomy interface for the knob-turning task. This may be because the knob had enough depth to be clearly visible in the 3D point cloud displayed in the teleoperation interface. Also, teleoperation allowed users to make fine adjustments and multiple attempts in quick succession. In the shared autonomy interface, the user cannot currently make small adjustments midway through the routine, but can only start the entire process over. Nevertheless, users generally felt that the shared autonomy interface was less frustrating, required less effort, and was less mentally demanding, as measured by the NASA Task Load Index [4].

5. FUTURE WORK

In this paper, we have outlined a shared-autonomy system for the operation of device controls that are hard to detect autonomously. We plan to extend the system to other simple controls, such as handles, and to controls that require sequencing of simple manipulations, such as numeric keypads. Currently, pressing buttons and flipping switches is done without explicitly detecting the device control, which can lead to unreliable operation. We plan to augment the system with closed-loop actuation, where the robot explicitly detects the control, and uses this to correct any errors in position.

6. REFERENCES

[1] J. A. Atherton and M. A. Goodrich. Perception by proxy: Humans helping robots to see in a manipulation task. In *Proceedings of the 6th ACM/IEEE International Conference on Human-Robot Interaction (HRI)*, pages 109–110, 2011.

[2] M. Beetz, U. Klank, I. Kresse, A. Maldonado, L. Mosenlechner, D. Pangercic, T. Ruhr, and M. Tenorth. Robotic roommates making pancakes. In *The 11th IEEE-RAS International Conference on Humanoid Robots*, pages 529–536, 2011.

[3] M. Ciocarlie, K. Hsiao, A. Leeper, and D. Gossow. Mobile manipulation through an assistive home robot. In *In Proceedings of the IEEE/RSJ International Conference on Robots and Systems (IROS)*, pages 313–5320, Vilamoura, Portugal, 2012.

[4] NASA. Task load index (TLX), 1986. Version 1.0 Manual.

[5] H. Nguyen, M. Ciocarlie, K. Hsiao, and C. C. Kemp. ROS Commander (ROSCo): Behavior creation for home robots. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, pages 476–474, Karlsruhe, Germany, 2013.

[6] M. Quigley, K. Conley, B. Gerkey, J. Faust, T. Foote, J. Leibs, E. Berger, R. Wheeler, and A. Ng. ROS: An open-source robot operating system. In *Proceedings of the ICRA 2009 Workshop on Open-Source Robotics*, 2009.

[7] B. Sankaran, B. Pitzer, and S. Osentoski. Failure recovery with shared autonomy. In *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 349–355, 2012.

[8] S. S. Srinivasa, D. Ferguson, C. J. Helfrich, D. Berenson, A. Collet, R. Diankov, G. Gallagher, G. Hollinger, J. Kuffner, and M. V. Weghe. HERB: A home exploring robotic butler. *Autonomous Robots*, 28(1):5–20, 2010.