

Multi-Perspective, Multimodal, and Machine Learning for Accessible Robotic Web Interface

INTRODUCTION

The ability to control robots can mean access to the physical world for many people with severe motor impairments. Accessible control of robotic interfaces is not well understood for people with physical disabilities and seniors to use assistive robotics. Although robots have the potential to significantly improve quality of life for people with disabilities and seniors, there remain significant barriers due to accessibility challenges of effectively controlling a high degree of freedom mobile robotic manipulator with a low fidelity interface. Existing interfaces for controlling robots are inefficient and inaccessible.

Our goal is to design and develop accessible robotic web interfaces to demonstrate the feasibility of tele-operating any given robotic manipulator for achieving physical independence. For any robotic mobile manipulator, we propose to create accessible low fidelity 3D Web interfaces for manipulating a high degree-of-freedom robot to assist with household activities and safely reposition the human body and limbs. This paper will show five robotic web interfaces and explain how we plan to analyze the effective accessibility of the interfaces.

BACKGROUND

Though teleoperation has been a part of robotics as long as robots have been around, the current teleoperation interfaces available for general manipulation tasks are exceedingly slow compared to able-bodied human manipulation. There are many degrees of freedom on a mobile robotic arm but persons with disabilities may not have access to effectively control that many degrees at one time. The control of 3D position and orientation of reaching a target point in the real-world and grasping an object with the end-effector is a grand challenge.

Much research has been done to expedite teleoperation by introducing greater autonomy into the system [1]. There is limited research to develop interfaces that provide accessible control of a robot's high degree of freedom (DOF) end-effector. At Georgia Institute of Technology, an accessible mobile manipulation interface was developed and tested by an expert user with severe motor impairments who was able to complete the Action Research Arm Test (ARAT) earning a score of 10 when each task was limited to one minute, and a score of 19 when the time limit was removed [2]. When the time constraint was removed however, the average time required to complete each task was over 3 minutes, a clear indicator of efficiency lack.

The interface used to control the robot dramatically impacts the user's teleoperation experience, and the amount of time required for the user to ascertain a strong mental model of the robot, and the environment that it is in. The interface developed at Georgia Institute of Technology utilizes a video-centric approach which allows the user to see the world from the robot's head-mounted camera. Though this is helpful for recognizing objects and understanding the layout of the environment, it does not clearly depict the distance of various objects in the environment to each other, because there is no understanding of depth when only one image is available.

Despite the advancements in robotics technology, several prominent challenges for people with disabilities and seniors include interface accessibility to control the robotic devices, real-time performance, and safety [3]. Providing users with disabilities and seniors access to robotic technologies requires the understanding of their needs, preferences, environment, and range of capabilities for input devices to issue commands. Given the proper technology, robotics can be made accessible, safe, affordable, effective, and efficient for long-term physical assistance and life-long care.

METHODS

Our primary goal is to develop accessible interfaces for any given robotic mobile manipulator to assist people perform self-care and household tasks safely and effectively. For example, by using a computer and a web browser a robot can help users independently pick up and place objects, get something to drink, and reposition the person's body and limbs. On all of our interfaces, the front-end is built on ROSLIBjs, and subscribes to our ROS back-end using Rosbridge Websocket. For building web-based interfaces to ROS systems we used Robot Web Tools: <http://robotwebtools.org> and the roslib.js javascript library. This package sets up a python ROS node

on the system which relays information to a web client in roslib.js over a websocket for consistent, low-latency communication. RViz is used to render the point cloud, and contains virtual cameras both above the robot and to the side of the robot that capture a video stream of the robot and its environment. Because the back end is sending a few video streams to the web client, the data transferred is far less than if the web client was sent the point cloud of the robot's environment. When the user uses one of the available controls to manipulate the robot's end effector, the commands are sent to the ros backend using ROSLIBjs, and are subsequently planned by MoveIt! and executed.

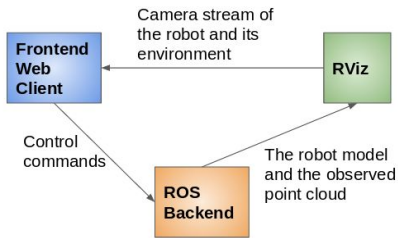


Figure 1. An architectural diagram of the system

We will perform a IRB approved user study with the ARAT test and our five design interfaces to analyze the effectiveness and accessibility of the interface with 30 subjects. Participants in our study will be asked to complete a demographic survey, a mouse-pointing task in a web browser for baseline data collection, operate a robot remotely using a web browser using their computer to complete a modified version of the ARAT test, and a survey to collect their experiences. We will analyze the trade-offs of multimodal control, switching between different modes, and accessibility. Our interfaces will provide multi-view perspective, multimodal interaction, and machine learning algorithms to model and observe interaction behaviors to recognize and suggest predictive user intentions. Our research seeks to retain the accessible aspects of the Georgia Tech interface, while making it more efficient, so that a novice user could score more than 19 points on the ARAT while still maintaining the one minute time limit per task.

Multi-perspective

The goal is to build an accessible interface for a mobile robotic arm by displaying the three different viewpoints from the top (X-Y plane) and side (Y-Z plane) perspectives and first person view of the robot in three separate windows on the interface. Specifically, the goals of the interface are to enable 3D manipulation of a mobile robotic arm in a 2D environment. By providing an overlay on the interface with augmented and virtual reality trajectories of the projected path, navigation control and robotic end-effector manipulation are achievable in 2D screen space with mouse gestures emulating the direction. The projected trajectory is created from the center point of the end-effector and to the position of the mouse release on the screen space. When the end-effector needs to reach a target location, the inverse kinematics from the current position of the end effector to the position on the projected trajectory is computed. The geometry of the scene will provide the solution to inverse kinematics.

Multimodal

Our design principle is to provide universal accessibility with multimodal control with voice, mouse, keyboard, or any other choice of input preferred by the user. What varies between our interfaces to the next is the method used to control the end effector. The current controls we plan to explore so far are (1) mouse point and click, (2) directional buttons, (3) speech recognition to give coordinates of a mouse click, and (4) a virtual joystick.

Machine Learning

To increase accessibility, we can minimize the number of interaction steps by learning and predicting actions on the interface using only the history patterns of user mouse movements and dynamics. Therefore, we will use machine learning for the classification of mouse actions depending on the screen region and specific time. This information can be used to assist the user in suggesting mouse cursor actions and mouse cursor location values.

Categories of mouse actions

1. Silence: no movement and no click
2. Point Single Left Mouse Click: mouse movement followed by a single left mouse click

3. Point Double Left Mouse Click: mouse movement followed by a double left mouse click
4. Point Single Right Mouse Click: mouse movement followed by a single right mouse click
5. Point Double Right Mouse Click: mouse movement followed by a double right mouse click
6. Drag-and-Drop Left Mouse: left mouse button-down, mouse movement, and then followed by left mouse button-up
7. Drag-and-Drop Right Mouse: right mouse button-down, mouse movement, and then followed by right mouse button-up

We will use the following data from the collected mouse dynamics over time user for training the learning model.

Mouse Dynamics Data

1. Category of mouse action
2. Distance (in pixels): length between current mouse cursor coordinate location to the new mouse cursor coordinate location
3. Elapsed Time (in seconds): time duration from start to finish of mouse action
4. Movement Direction: angle of mouse movement in the categorical directions of top, top-left, left, down-left, down, down-right, right, and top-right from start to finish of mouse action separated by 45° clockwise from the top direction
5. Interface resolution: dimensions of the displayed interface corresponding with the mouse actions
6. Interface mouse speed: system settings from mouse cursor speed and mouse cursor acceleration.
7. Mouse configuration: specific type of mouse
8. Hardware configuration: specific type of mouse

We may also be able to detect anomalies based on the observing the patterns of mouse dynamics and keyboard stroke behavior. During speech recognition, we can use audio events for context-aware recognition to switch between limb repositioning tasks and household activities with speech recognition that can be tailored to context by providing a separate set of grammar words that can utilize advanced deep learning neural network algorithms to match with the user's audio for speech recognition

Novel Robotic Interfaces

Simple Buttons

There are navigational buttons on each camera view for gripper movement.

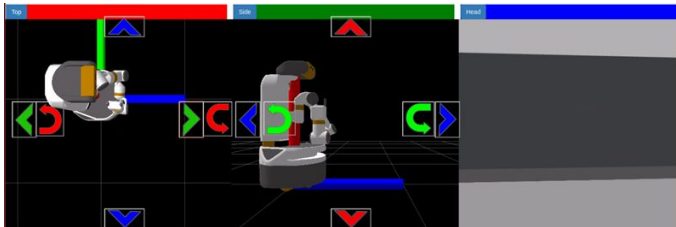


Figure 2. Screenshot of Simple Buttons Interface

Click and Orient

Allows the user to click a location on the screen that the gripper will move to, and then drag the mouse to determine the orientation of the gripper.

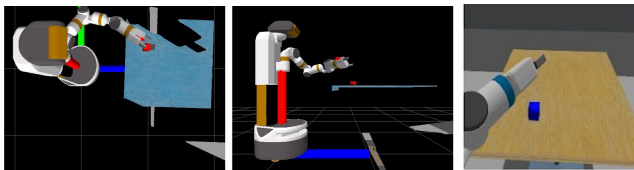


Figure 3. Screenshot of Click and Orient Interface with the red arrow indicating the location of mouse click and drag.

One Touch

Similar to Click and Orient but does not require a mouse drag. Once you click on the camera view, the gripper will follow the mouse movement.

Drag and Rotate

Similar to Click and Orient and provides additional support to orient the gripper. There is a circular ring around the gripper in both camera views, and users can “rotate” this ring to rotate the gripper.

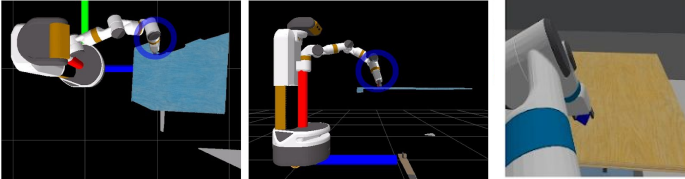


Figure 4. Screenshot of Drag and Rotate Interface with blue circular ring for orientation.

Virtual Joystick

Use a virtual joystick to move the gripper in either camera stream

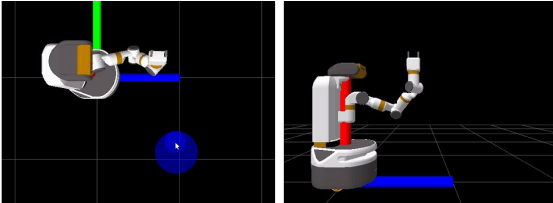


Figure 5. Screenshot of Virtual Joystick Interface with blue circle as virtual joystick.

PRELIMINARY RESULTS

Our preliminary results have shown that using two static orthogonal views to complete a manipulation task is faster than having one dynamic view which requires constant adjustment to gain an understanding of the environment. We completed a preliminary study as exploratory research with a small group of three test subjects in which the subject completed manipulation tasks using RViz interactive markers with either one or two views. This study found that the two-view interface consistently outperformed the one-view interface over user preference. We will be completing a more rigorous study with more subjects that use the interfaces to complete the ARAT test with our developed interfaces once they have been completely implemented.

CONCLUSION AND FUTURE DIRECTIONS

Our interfaces illustrates our approach to accessible control of robots as a proof-of-concept. There is a dire need for more work in this area to extend the autonomy of persons with disabilities and seniors via assistive robotics. Ultimately, our technique will promote increased physical HRI with accessible robotic interfaces.

REFERENCES

- [1] Kent, D., Saldanha, C., & Chernova, S. (2017, March). A Comparison of Remote Robot Teleoperation Interfaces for General Object Manipulation. In *Proceedings of the 2017 ACM/IEEE International Conference on Human-Robot Interaction* (pp. 371-379). ACM.
- [2] Grice, P. M., & Kemp, C. C. (2016). Assistive mobile manipulation: Designing for operators with motor impairments. In *RSS 2016 Workshop on Socially and Physically Assistive Robotics for Humanity*.
- [3] Krishnaswamy, K., Moorthy, S., & Oates, T. (2017, June). Survey Data Analysis for Repositioning, Transferring, and Personal Care Robots. In *Proceedings of the 10th International Conference on Pervasive Technologies Related to Assistive Environments* (pp. 45-51). ACM.