Comparing Robot Grasping Teleoperation across Desktop and Virtual Reality with ROS Reality

David Whitney, Eric Rosen, Elizabeth Phillips, George Konidaris, Stefanie Tellex

Abstract Teleoperation allows a human to remotely operate a robot to perform complex and potentially dangerous tasks such as defusing a bomb, repairing a nuclear reaction, or maintaining the exterior of a space station. Existing teleoperation approaches generally rely on computer monitors to display sensor data and joysticks or keyboards to actuate the robot. These approaches use 2D interfaces to view and interact with a 3D world, which can make using them difficult for complex or delicate tasks. To address this problem, we introduce a virtual reality interface that allows users to remotely teleoperate a physical robot in real-time. Our interface allows users to control their point of view in the scene using virtual reality, increasing situational awareness (especially of object contact), and to directly move the robot's end effector by moving a hand controller in 3D space, enabling fine-grained dexterous control. We evaluated our interface on a cup-stacking manipulation task with 18 users, comparing the relative effectiveness of a keyboard and mouse interface, virtual reality camera control, and positional hand tracking. Our system reduces task completion time by 101 seconds (a reduction of 66%), while improving subjective assessments of system usability and workload. Additionally, we have shown the effectiveness of our system over long distances, successfully completing a cup stacking task from over 40 miles away. Our paper contributes a quantitative assessment of robot grasping teleoperation across desktop and virtual reality interfaces.

David Whitney
Humans To Robots Lab, Brown University, e-mail: david_whitney@brown.edu

Eric Rosen
Humans To Robots Lab, Brown University, e-mail: eric_rosen@brown.edu

Elizabeth Phillips
Humanity Centered Robotics Initiative, Brown University, e-mail: elizabeth_phillips1@brown.edu

George Konidaris
Intelligent Robot Lab, Brown University, e-mail: gdk@cs.brown.edu

Stefanie Tellex
Humans To Robots Lab, Brown University, e-mail: stefie10@cs.brown.edu

1 Introduction

Whether navigating a nuclear reactor station, defusing a bomb, or repairing the International Space Station from the outside, robots have the ability to be in places where humans cannot or should not go. Deft manipulation in those places could save lives. Since even the most advanced autonomous robots are unable to perform tasks that require grasping and manipulation [13], human teleoperation is often a practical alternative—importing the dexterity, expertise, and wealth of background knowledge of a human operator without requiring them to be physically present.

In order to perform manipulation tasks, human operators need high-fidelity control over a robot's actuators and an accurate visualization of its environment. State-of-the-art teleoperation systems require the operator to both manage their view of the scene and, separately, command the robot's actuators. This is typically performed with keyboard and mouse interfaces, as was the case for the DARPA Robotics Challenge [13].

The recent renewed interest in, and lowered prices of, virtual reality (VR) devices have raised the possibility of using VR as an interface for robot teleoperation. VR promises a user experience that is both immersive and detailed, coupled with complete freedom of viewpoint and a natural method of expressing robot action. Indeed, previous research has found that immersive interfaces leads to significant benefits in therepeutic applications [12, 8]. While there have been a few systems that enable robot teleoperation using VR [4, 3, 2], these systems did not provide the fluid camera control and handtracking movement, and their performance has not been evaluated empirically.

Fig. 1: Using ROS Reality, our virtual reality interface, to teleoperate a robot arm to perform a cup stacking task. The user is able to see a 3D model of the robot, an overlaid point cloud, and a camera feed from the wrist camera of the robot.

We present a VR interface that allows an untrained user to control a robot arm to carry out fine-grained manipulation tasks. Using VR camera control, the operator can quickly obtain situational awareness by moving their head and body around the scene; the point of view follows. The operator can also directly control the arm's

end effector position by simply moving their own hand, via a combination of positional hand tracking and combined with autonomous collision avoidance. These two capabilities allow the user to effectively carry out fine-grained tasks. Figure 1 shows a user controlling the Baxter robot using our system, as well as the user's view. The robot's sensors—in this case a calibrated 3D point cloud, its joint sensors, and wrist camera—are used to visualize the robot's environment. The person can move the robot's end effector by dragging it in the virtual space, causing the real robot to move.

We performed a quantitative assessment of the system's ability to improve robot pick-and-place teleoperation a grasping and manipulation task using a Baxter robot. Eighteen subjects used our VR system to teleoperate the robot to perform a variant of the YCB [7] cup stacking task. Our system reduced task completion time by 101 seconds on average (a reduction of 66%) while improving subjective assessments of system usability. We have released our system as a ROS package, ROS Reality,[1] which includes integration with the HTC Vive and the widely-used Unity game engine, and a URDF parser that allows new robots to be quickly imported into Unity.

2 Related Work

Much of the robotic research community has settled on Robot Operating System (ROS) as their software of choice [14]. In ROS, teleoperation is generally done with the joint use of the built in visualization software RViz [10] and the ROS Interactive Manipulation (IM) stack [1]. The IM stack uses a point and click interface using a computer monitor, which is cumbersome and slow compared to our interface.

In 2013, Willow Garage released an RViz plugin for the Oculus Rift DK1, a developer pre-production virtual reality headset [15]. This led to the creation of several VR-teleoperation packages [2, 4]. Unfortunately, the Rift did not yet support positional head or hand tracking, limiting the usefulness of systems. In order to track hands, the packages relied on third-party hand trackers. Additionally, RViz is meant to be run on the same Local Area Network (LAN) as the robot, and in our experience, suffers from latency issues if large amounts of data, like a point cloud, are streamed over the Internet. Our system is able to mitigate this issue by deconstructing the point cloud, sending separate compressed depth and color images, and using a custom GPU shader to reconstruct the point cloud on the VR computer.

One similar system to ours is Mind Meld [3]. Designed by McCarthy et al., it is a VR-teleoperation system that uses Unity render data from a PR2 robot. Hands are tracked using custom-made 3D-printed grippers, and the scene is viewed using the previous generation Oculus DK2 headset. Mind Meld is not publicly available, and unlike our system, is not designed to work over long distances. Additionally, our work provides an empirical evaluation of the system for object manipulation.

[1] Our code is at `https://github.com/h2r/ros_reality`.

The DARPA Robotics Challenge (DRC) [13] is a challenge motivated by DARPA's goal of developing human-supervised robots to perform dangerous, complex tasks. Robotic teams competed to create robots that could drive, move through rubble, turn valves, and climb stairs. The robots were semi-autonomous, and the winning team, HUBO, stated that towards the end of the challenge the team had a "strong focus on human teleoperation" [16]. The teleoperation interface was RViz and a variant of the Interactive Manipulation stack. Virtual reality could be a useful tool for helping robots perform DRC-like tasks due to the superior scene understanding it enables. Since the DRC bandwidth limited entrants, it is important to note that VR interfaces do not require more bandwidth from the robot than a keyboard and monitor interface.

2.1 Contributions

The main contributions of this paper are the system ROS Reality, as well as a user study showing the improved performance of virtual reality interfaces compared to more traditional interfaces. ROS Reality is the first publicly available package connecting the current generation of commercially available virtual reality hardware (HTC Vive and Oculus Rift) to a ROS network. These VR systems are (relatively) low cost and have highly accurate tracking. ROS Reality allows designers to import arbitrary robot URDFs to build virtual robot models, as well as send and receive multiple ROS topics across the Internet. The system currently has visualization tools for camera, point-cloud, and TF topics. Point-clouds in particular are broken down into separate compressed depth and color images, and reconstructed on the VR computer using a custom shader to keep a low latency.

The user study presented here demonstrates a very large reduction in task completion time for VR systems compared to keyboard and monitor interfaces. From our results, it seems hand-tracking is the key component that increases task speed, and the virtual reality scene causes the large gains in workload, usability, and likability. The evaluation and discussion may help guide future studies on assessing similar interfaces for teleoperation.

3 Technical Approach

Our aim is to design a virtual reality interface that allows human operators to 1) effectively perceive the robot's environment and 2) effectively control the robot's effectors to carry out fine-grained tasks. To achieve our first goal, we created a virtual environment that captures and legibly displays the state of the robot's environment. The movement of the camera in the scene is controlled by the movement of a virtual reality headset worn by the user. To achieve our second goal, we designed a positional tracking system that maps the movements of a tracked controller to de-

sired end-effector movement. This interface allows the operator to move the robot's effector in real space by moving their own hands. The architecture of our system, ROS Reality, is shown in Figure 2.

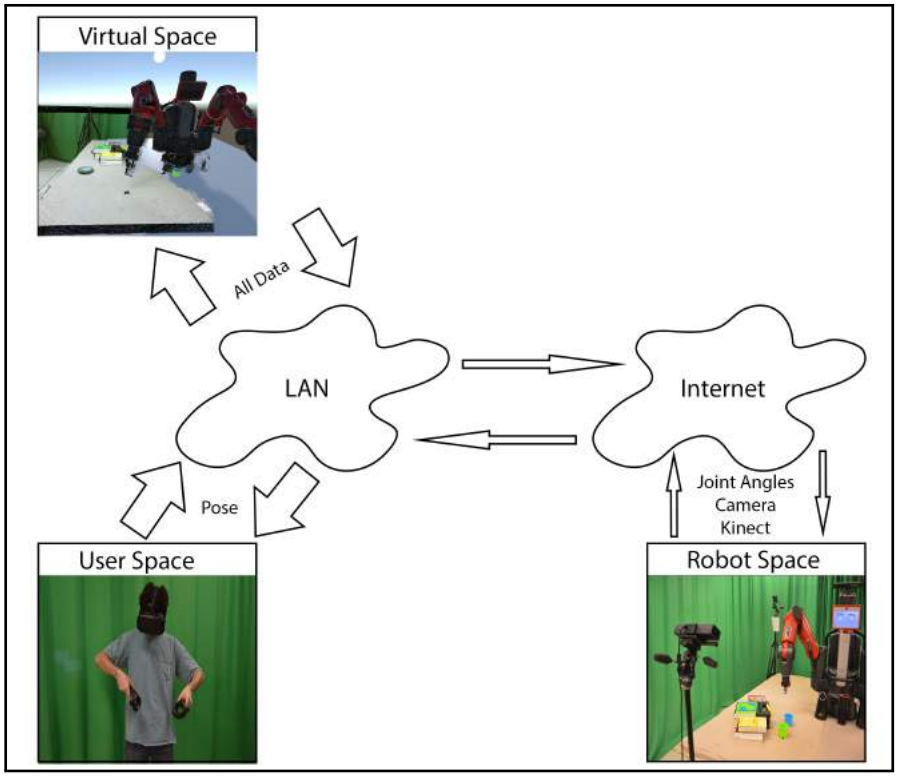

Fig. 2: A visual representation of the abstraction layers of ROS Reality. Sensory data is sent from the robot to the user, and movement commands are sent from the user to the robot. Arrow thickness corresponds to bandwidth.

We chose the HTC Vive as our VR interface because it is capable of tracking the pose of both the headset and the hand controllers (with the caveat that the user is confined to a $5m^2$ play-area). The Vive provides users with the ability to walk around a 3D virtual environment to attain strong situational awareness through multiple perspectives.

3.1 Scene Rendering and Head Tracking

The core task of a VR visualization system is creating a realistic and natural visualization of the environment—fusing sensor data, joint positions, and available object and robot models—that responds to the operator's movements in real time, without disorienting lag. This requires a high-performance rendering system. Our strategy in developing that system was to stream sensor data to a powerful local computer, which built and displayed a local model of the world using Unity. Rendering thus takes place close to the user and the VR headset. User movements—captured by the Vive's head-tracking system—caused the rendering point of view to change within that local model, and updates from the robot's sensors modified the local model asynchronously.

The virtual scene in Unity consisted of three primary informational components: 1) a 3D model of the robot, obtained by importing a description of the robot in URDF format; 2) a 3D point-cloud of the scene, obtained by a calibrated Kinect v2 sensor mounted near the robot; and 3) a display of the robot's wrist camera (a 1280×800 pixel RGB camera, downscaled to 400×600, showing a very useful high resolution live image of the environment immediately forward of the manipulator). Unity's high quality rendering resulted in a generated virtual robot that looked realistic, and movement with no perceived lag.

3.2 Positional Hand Tracking

A key reason for our selection of the HTC Vive is that it includes two hand controllers, which are tracked in the same way that the headset pose is tracked. The hand controllers include a few buttons, which we used to allow the operator to indicate when moving the hand controllers should also move the robot's end-effector, and when the gripper should open or close. The hand controllers are visualized in the scene (in the same way that the robot is visualized in the scene), which allows the operator to move the robot's end effector by virtually dragging it through space. To achieve this, we used Baxter's built-in collision detection and inverse kinematics to generate trajectories corresponding that follow the user's motion.[2] This provided the operator with fine-grained, full-pose control of the robot end-effector.

4 Evaluation

Our evaluation assesses the effectiveness of VR camera control and positional hand tracking as teleoperation interfaces. To do so, we asked novice users to teleoperate

[2] If a requested end effector pose is not possible to attain due to collision or not being in the robot's work space, then the robot does not move.

a Baxter robot to perform a cup-stacking task in four ways: directly manipulating the arm, and using three different teleoperation interfaces: keyboard and monitor, positional hand tracking and monitor, and positional hand tracking with VR camera control. We report task completion time as an objective metric, as well as subjective assessments of system usability, likability, and workload.

As an additional demonstration of effectiveness, we completed a larger cup stacking task from 41 miles away, stacking 10 cups in a row with a Baxter robot in Cambridge, MA from Providence, RI. A video of this demonstration can be found here: `https://www.youtube.com/watch?v=HlRJZYNNndI`

4.1 Task

Each user was given the task of assembling three cups—all located on a table in front of the robot—into a single stack, by controlling a Baxter robot's right arm to first place the blue cup into the green cup, and then the blue-green stack into the yellow cup. The blue and green cups were placed flat on the table, while the yellow cup was propped up at a 45-degree angle. The cups were taken from the group of stacking cups in the YCB Object set [7]. The task is shown in Figure 3. During teleoperation, the users controlled the robot from a computer across the room, and a divider blocked their line of sight.

Fig. 3: Pictures of the cup-stacking task: (left) the initial configuration, (middle) the blue cup stacked in the green, and (right) the blue-green stack into the yellow cup.

This task was designed to be difficult. The cups fit snugly into each other, with a clearance of under two millimeters. The blue and green cups were not secured to the table, and were liable to be knocked over if bumped. The angle of the yellow cup required the operator to rotate the robot arm about two of its axes, a dexterous task that forced the operator to consider the arm's orientation and position simultaneously.

4.2 Interfaces

Our experiment compared four interfaces:

4.2.1 Direct Manipulation (Direct)

Users physically moved the arm in order to complete the task. We chose this interface as the lower bound, best-case baseline for the task. The users were able to directly view the cups and move the arm. An ideal teleoperation system would be as fast and accurate as direct manipulation.

4.2.2 Keyboard and Monitor (KM)

Users viewed the scene using a 1080p 23" computer monitor. The users could move the camera through the scene using a mouse, and control the robot's gripper using a keyboard interface,[3] in a manner typical of software interfaces such as RViz [14] and Gazebo [11]. The robot's end effector was controlled via a keyboard interface.

4.2.3 Positional Hand Tracking with Monitor (PM)

Users view the scene and control the camera as in the keyboard and monitor interface, but control the arm with the positional tracking interface. This interface allows us to study the effect of positional tracking—a relatively new aspect of VR headsets—without virtual reality camera control.

4.2.4 Positional Hand Tracking with Virtual Reality Camera Control (PV)

Users viewed the scene using an HTC Vive virtual reality headset, and controlled the arm using an HTC Vive hand controller. The VR headset allowed the user to move about the scene at will, and the hand controller controlled the gripper using the positional hand tracking technique described in section 3.2. This is the complete version of our system.

4.3 Experimental Procedure

Users teleoperated the robot to perform the cup-stacking task with each interface. Direct manipulation was always done first, to gain familiarity with the robot. Next, they performed each of the three teleoperation schemes in random order. Each of the six possible random orderings was performed by three different users. After using each interface, users filled out subjective evaluations for that interface. After using all interfaces, users filled out a form asking for further subjective measures, such choosing their favorite interface, and basic demographic information.

[3] The `WASD` keys governed horizontal movement, Q and E moved the arm down and up, and R and F opened and closed the grippers. The shift key switched translational movement to rotational.

For each interface, we instructed the user how to move the robot and view the scene. We asked users to complete the task as quickly as possible. They were then given as many attempts as they liked to complete the task. For each task attempt, the experimenter gave a countdown and then started a stopwatch. The experimenter stopped the stopwatch once all three cups were completely stacked. If the user knocked over a cup or otherwise made the task impossible, an experimenter recorded the time, reset the objects, and restarted the attempt.

4.4 Participants

Our evaluation used 18 participants (11 male, 7 female) with ages ranging from 18 to 22 ($M = 19.78$, $SD = 1.17$). Video game usage at peak varied between users from 0 to 30 hours per week ($M = 8.36$, $SD = 8.76$).

4.5 Measurements

In our experiment, the independent variable was the choice of interface. Our objective dependent variable was the time to completion of the task. For this measure, we took each users' fastest time for each interface. Five of the eighteen users were unable to complete the task with the keyboard and monitor interface and two users were unable to complete the task with the positional tracking and monitor interface. For those users, we chose the attempt in which the user was closest to completing the task.

Our subjective dependent variables were user workload as measured by the NASA Task Load Index (NASA-TLX) [9], system usability as measured by the System Usability Scale (SUS) [6, 5], and system likability as measured by several Likert scale questions. Each of these measures were collected via questionnaires at various points throughout the experiment.

The NASA-TLX is a widely used assessment tool that measures perceived workload of a particular task [9]. It measures global workload across six sub-scales: mental demand, physical demand, temporal demand, effort, frustration, and performance. Users were asked to provide a rating of their perceived mental workload along each of the six dimensions via a scale ranging from 0 (Low) to 100 (High) for the first five dimensions and 0 (Perfect) to 100 (Failure) for the performance dimension. For this evaluation, the weighted measure of paired comparisons among the sub-scales was not included. The workload score is calculated as the average of the six sub-scales. Therefore, the best workload score is 0 and the worst is 100.

Users assessed each interface on overall usability by filling out a System Usability Scale (SUS) questionnaire [6, 5]. The SUS questionnaire asks users to rate ten sentences on a 7-point Likert scale ranging from "strongly disagree" to "strongly agree." The sentences cover different aspects of the system, such as complexity,

consistency, and cumbersomeness. Like the NASA-TLX, the SUS is measured on a scale from 0 to 100. For the SUS, however, 0 is the worst score, and 100 is the best.

For our final subjective measure, we asked each participant to rate the various interfaces in terms of likability on a Likert scale from 1 to 7.

As a covariate measure, we asked participants how many hours of video-games they played per week at their peak.

4.6 Hypotheses

We expected that users would show the best performance (i.e., the fastest completion times, lowest levels of mental workload, highest usability and likability scores) in the Direct Manipulation Interface condition, followed by the Positional Hand Tracking with VR condition, and then the Positional Hand Tracking with Monitor condition. Finally, we posited that the Keyboard and Monitor condition would be associated with the lowest levels of performance.

Specifically, we had 3 hypotheses:

- **H1:** The Direct Manipulation Interface condition will be associated with the best performance of the four conditions, as demonstrated by (a) the fastest task completion times, (b) the lowest levels of mental workload, (c) the highest usability scores, and (d) the highest likability ratings.
- **H2:** The Positional Tracking with Virtual Reality Interface condition will be associated with the best performance out of the teleoperated conditions.
- **H3:** Of the remaining teleoperated conditions, the Positional Hand Tracking with Monitor condition will be associated with better performance than the Keyboard and Monitor condition.

The first hypothesis reflects our idea that Direct Manipulation is the easiest interface for completing the cup stacking task. The remaining hypotheses reflect our thought that using the Vive HUD would offer environmental perception that leads to quicker task completion than looking at a monitor, and that having position/pose-tracking hand controllers will make it faster and more intuitive to control the robot than a keyboard.

4.7 Results

To analyze the three hypotheses, four Analyses of Covariance (ANCOVAs) were used to look for significant differences between the conditions on the four dependent measures (i.e., task completion times, NASA-TLX, SUS, and Likability measure). Planned contrasts were conducted to test for significant differences between individual conditions. Specifically, planned contrasts were conducted to look for significant differences on the dependent measures between the Direct Manipulation

Table 1: Results

(a) Table of means, standard deviations, and significant contrasts between experimental conditions on the time to completion dependent measure. †
ANCOVA $F(3,14) = 37.840$, $p < .001$, partial $\eta^2 = .890$, $N = 18$, LSD Significant between two conditions at $p < .05$

	Time to complete task		
Condition	Mean	SD	Significant Contrast
Direct	8.15	2.68	KM,PM,PV
KM	153.43	44.37	Direct,PM,PV
PM	79.81	39.09	Direct, KM
PV	52.56	37.16	Direct, KM

(b) Table of means, standard deviations, and significant contrasts between experimental conditions on the NASA-TLX dependent measure. †
ANCOVA $F(3,13) = 12.289$, $p < .001$,partial $\eta^2 = .739$ $N = 17$, LSD Significant between two conditions at $p < .05$ *Contrast marginally significant at $p = .058$

	NASA-TLX Measure		
Condition	Mean	SD	Significant Contrast
Direct	29.31	12.54	KM,PM,PV
KM	56.37	13.71	Direct,PM*,PV
PM	51.08	15.90	Direct, KM*, PV
PV	44.95	20.53	Direct, KM

(c) Table of means, standard deviations, and significant contrasts between experimental conditions on the SUS dependent measure. †
ANCOVA $F(3,12) = 6.847$, $p = .006$,partial $\eta^2 = .631$, $N = 16$, LSD Significant between two conditions at $p < .05$ *Contrast marginally significant at $p = .056$

	System Usability Scale		
Condition	Mean	SD	Significant Contrast
Direct	71.25	9.97	KM,PM
KM	37.29	19.13	Direct,PM,PV
PM	55.94	21.01	Direct, KM, PV*
PV	71.46	19.61	KM, PM*

(d) Table of means, standard deviations, and significant contrasts between experimental conditions on the Likability dependent measure. †
ANCOVA $F(3,14) = 24.679$, $p < .001$, partial $\eta^2 = .894$, $N = 18$, LSD Significant between two conditions at $p < .05$

	Likability Measure		
Condition	Mean	SD	Significant Contrast
Direct	5.61	1.61	KM,PM
KM	2.06	1.35	Direct,PM,PV
PM	4.28	1.71	Direct, KM, PV
PV	6.11	1.41	KM, PM

condition and each of the teleoperation conditions (i.e., Condition 1 vs. 2, 3, and 4, independently). Planned contrasts were also conducted to look for differences on the dependent measures between the VR condition and each of the other teleoperated conditions (i.e., Condition 4 vs. 2 and 3), and planned contrasts were conducted to look for significant differences on the dependent measures between the Positional Hand Tracking with Monitor condition and the Keyboard and Monitor condition (i.e., Condition 3 vs 2).

A one-way repeated measures ANCOVA with task completion times for each experimental condition as the within-subjects variable, and scores on the measure of peak video game hours as the covariate, was used to test for significant differences in mean teleoperation task completion times across the interface conditions. The test revealed that there was a significant difference in mean task completion times across the four interface conditions, Wilks $\Lambda = 0.110$ $F(3,14) = 37.840$, $p < 0.001$, $\eta^2 = 0.890$. Planned contrasts using the LSD method were conducted to test for significant differences in task completion times between conditions. The means,

standard deviations, and statistically significant contrasts between conditions are presented in Table 1a.

The Direct Manipulation condition resulted in statistically significantly faster task completion times than any of the other conditions. This result supports Hypothesis H1, which stated that the Direct Manipulation condition would be associated with the best performance on the task completion time measure. Further, of the teleoperated conditions, the PV condition was associated with the fastest task completion times. However, the PV condition was only statistically significantly faster than the KM condition, but not the PM condition. These findings only lend partial support for Hypothesis H2, which stated that the PV condition would be associated with significantly faster task completion times than both the PM and KM conditions . Finally, the PM condition was statistically significantly faster than the KM condition, which supports Hypothesis H3.

A one-way repeated measures ANCOVA with scores on the NASA-TLX for each experimental condition as the within-subjects variable, and scores on the measure of peak video game hours as the covariate, was used to test for significant differences in users' subjective mental workload across the conditions. The test revealed that there was a significant difference in mean NASA-TLX scores across the four interface conditions, Wilks $\Lambda = 0.261$ $F(3,13) = 12.298$, $p < 0.001$, $\eta^2 = 0.739$. Planned contrasts using the LSD method were conducted to test for significant differences in NASA-TLX scores between conditions. The means, standard deviations, and statistically significant contrasts between conditions are presented in Table 1b.

For the NASA-TLX measure, the Direct Manipulation condition resulted in statistically significantly lower subjective workload scores than any of the other conditions. This result supports Hypothesis H1, which stated that the Direct Manipulation condition would be associated with the lowest levels of workload among the four conditions. Further, of the teleoperated conditions, the PV condition was associated with the lowest levels of subjective workload. However, workload scores in the PV condition were only statistically significantly lower than the KM condition, but not the PM condition. These findings lend only partial support for Hypothesis H2, which stated that the PV condition would be associated with significantly lower workload scores than both the PM and KM conditions. Finally, the difference in workload scores between the PM condition and the KM condition was not statistically significant at the $p = 0.05$ level, instead the difference between the two conditions approached significance at $p = 0.058$. This finding only lends partial support for Hypothesis H3.

A one-way repeated measures ANCOVA with scores on the SUS for each experimental condition as the within-subjects variable, and scores on the measure of peak video game hours as the covariate, was used to test for significant differences in subjective assessments of the usability of each interface across the conditions. The test revealed that there was a significant difference in mean SUS scores across the four interface conditions, Wilks $\Lambda = 0.369$ $F(3,12) = 6.847$, $p = 0.006$, $\eta^2 = 0.631$. Planned contrasts using the LSD method were conducted to test for significant differences in SUS scores between conditions. The means, standard deviations, and statistically significant contrasts between condition's are presented in Table 1c.

The Direct Manipulation condition was associated with higher SUS scores than all of the other conditions except the PV condition. Thus, Hypothesis H1 which stated that the DM condition would be associated with the highest SUS scores of all of the conditions was not supported. Of the teleoperated conditions, however, the PV condition was associated with the highest SUS scores out of any of the conditions, strongly supporting H2. Finally, the difference in SUS scores between the PM condition and the KM condition was not statistically significant at the $p = 0.05$ level, instead the differences between the two conditions approached significance at $p = 0.056$. This finding only lends partial support for Hypothesis H3.

A one-way repeated measures ANCOVA with scores on the Likability measure for each experimental condition as the within-subjects variable, and scores on the measure of peak video game hours as the covariate, was used to test for significant differences in assessments of how much users liked interacting with each interface. The test revealed that there was a significant difference in mean Likability scores across the four interface conditions, Wilks $\Lambda = 0.159$ $F(3,14) = 24.679$, $p < 0.001$, $\eta^2 = 0.841$. Planned contrasts using the LSD method were conducted to test for significant differences in Likability scores between conditions. The means, standard deviations, and statistically significant contrasts between condition's are presented in Table 1d.

Similar to the SUS results, on the likability measure, the Direct Manipulation condition was associated with higher SUS scores than all of the other conditions except the PV condition. Thus, Hypothesis H1 which stated that the DM condition would be associated with the highest Likability scores across all of the conditions was not supported. Instead, the PV condition had the highest Likability scores in comparison to all the other conditions, again lending strong support for Hypothesis H2. Finally, the difference in Likability scores between the PM and KM condition was statistically significant, where users rated liking interacting with the PM interface more than the KM interface, supporting Hypothesis H3.

5 Discussion

Overall, we found that the full VR interface was significantly better in both the objective and subjective metrics compared to the keyboard and monitor interface. It was faster, with an average improvement of 101 seconds (66% improvement), and was rated as having a lower workload and higher usability, as measured by the NASA-TLX and SUS measures, respectively. Additionally, the full VR interface was much more liked, with an average likability score of 6.11 (out of 7), compared to 2.06. This result supports Hypothesis H2 and is encouraging, as it implies that a user performing VR teleoperation tasks would be both faster and happier than if they were using a keyboard and monitor interface.

Interestingly, while the full VR interface was on average faster than the positional hand tracking with monitor interface, it was not significantly so. This implies that the positional hand tracking was more important to the task speed than the

VR camera control. The workload was also not significantly different. The system usability, however, was highly significantly different. The full VR interface scored much higher on the SUS test, $M = 71.46$ compared to $M = 55.94$. This implies that although users were able to complete the task with the monitor, they found it more difficult to use than the VR interface, further supporting Hypothesis H2.

As expected, the VR interface was slower than direct manipulation of the arm. Direct manipulation allows the user to see the cups with their own eyes and move the robot with their own hands. The fastest time recorded for direct manipulation was 5.5 seconds, which we believe approaches the physical limit of the task. The workload score was also significantly lower, which may be due to the shorter times the users achieved with direct manipulation. Both the fast time to complete the teleoperation task and the low workload scores strongly supported H1. Surprisingly, however, the VR interface actually had a marginally higher SUS score compared to direct manipulation, $M = 71.46$ to $M = 71.25$. We believe this is because SUS measures the complexity, consistency, and ease of use of a system, not physical effort or objective success.

Users failed the task when a cup was knocked over or dropped, leading it to roll out of reach of the robot. This happened the most with the keyboard and monitor interface. Five of the eighteen users were never able to complete the task with the keyboard interface. Two users were never able to complete the task with the positionally tracked controller and monitor, and all users completed the task with the VR interface at least once.

6 Conclusions and Future Work

This paper presents a novel, virtual-reality-based interface for remote robot teleoperation. This interface allows novice users to complete a manipulation task faster than a keyboard and mouse interface, with lower reported workload and higher usability.

In the future, we aim to use ROS Reality as a basis for further research into VR-based interfaces. One challenge is extending our system to deal with much more complex tasks—involving both more dexterous manipulation and combining navigation with manipulation—which will require more and higher resolution sensors, with all the bandwidth and processing requirements that entails. We also plan to explore mixed-initiative autonomy, where the robot behaves autonomously for some parts of the tasks, and seeks human input (in the form of guidance or commands) for the remainder. Virtual reality interfaces offer a teleoperation modality that is both immersive and intuitive, with the opportunity to substantially extend the range of tasks that can be successfully completed by remotely operated robots.

Acknowledgements

This material is based upon work supported by the Defense Advanced Research Projects Agency (DARPA) under Grants No. W911NF-15-1-0503, YFA: D15AP00104, YFA: GR5245014, and D15AP00102, as well as NASA under Grants No. GR5227035.

Any opinions, findings, and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of DARPA.

References

[1] Interactive Manipulation im description. `http://projects.csail.mit.edu/pr2/wiki/index.php?title=Interactive_Manipulation`.

[2] IVRE - An Immersive Virtual Robotics Environment. `https://cirl.lcsr.jhu.edu/research/human-machine-collaborative-systems/ivre/`.

[3] Mind Meld. `https://www.youtube.com/watch?v=kZlg0QvKkQQ`.

[4] PR2 Surrogate. `http://wiki.ros.org/pr2_surrogate`.

[5] A. Bangor, P. T. Kortum, and J. T. Miller. An empirical evaluation of the system usability scale. *Intl. Journal of Human–Computer Interaction*, 24(6):574–594, 2008.

[6] J. Brooke et al. Sus-a quick and dirty usability scale. *Usability evaluation in industry*, 189(194):4–7, 1996.

[7] B. Calli, A. Singh, A. Walsman, S. Srinivasa, P. Abbeel, and A. M. Dollar. The ycb object and model set: Towards common benchmarks for manipulation research. In *Advanced Robotics (ICAR), 2015 International Conference on*, pages 510–517. IEEE, 2015.

[8] Y.-p. Chen, S.-Y. Lee, and A. M. Howard. Effect of virtual reality on upper extremity function in children with cerebral palsy: a meta-analysis. *Pediatric Physical Therapy*, 26(3):289–300, 2014.

[9] N. H. P. R. Group et al. Task load index (nasa-tlx) v1. 0 computerised version. *NASA Ames Research Centre*, 1987.

[10] H. R. Kam, S.-H. Lee, T. Park, and C.-H. Kim. Rviz: a toolkit for real domain data visualization. *Telecommunication Systems*, 60(2):337–345, 2015.

[11] N. Koenig and A. Howard. Design and use paradigms for gazebo, an open-source multi-robot simulator. In *Intelligent Robots and Systems, 2004.(IROS 2004). Proceedings. 2004 IEEE/RSJ International Conference on*, volume 3, pages 2149–2154. IEEE, 2004.

[12] K. R. Lohse, C. G. Hilderman, K. L. Cheung, S. Tatla, and H. M. Van der Loos. Virtual reality therapy for adults post-stroke: a systematic review and meta-analysis exploring virtual environments and commercial games in therapy. *PloS one*, 9(3):e93318, 2014.

[13] G. Pratt and J. Manzo. The darpa robotics challenge [competitions]. *IEEE Robotics & Automation Magazine*, 20(2):10–12, 2013.

[14] M. Quigley, K. Conley, B. Gerkey, J. Faust, T. Foote, J. Leibs, R. Wheeler, and A. Y. Ng. Ros: an open-source robot operating system. In *ICRA workshop on open source software*, volume 3, page 5. Kobe, 2009.

[15] D. Serrano. Introduction to ros–robot operating system–.

[16] M. Zucker, S. Joo, M. X. Grey, C. Rasmussen, E. Huang, M. Stilman, and A. Bobick. A general-purpose system for teleoperation of the drc-hubo humanoid robot. *Journal of Field Robotics*, 32(3):336–351, 2015.