Manipulation Assist for Teleoperation in VR

Christian Barentine,[1] Andrew McNay,[1], Ryan Pfaffenbichler,[1] Addyson Smith,[1] Eric Rosen, [2], and Elizabeth Phillips[3]
[Christian.Barentine,C21Andrew.McNay,
C22Ryan.Pfaffenbichler,C21Addyson.Smith]@afacademy.af.edu,eric_rosen@brown.edu,ephill3@gmu.edu
[1]United States Air Force Academy, [2]Brown University, [3] George Mason University
Air Force Academy, Colorado

ABSTRACT

VR interfaces are promising for robotics for several reasons, including that they may be suitable for resolving many of the human performance issues associated with traditional robot teleoperation interfaces used for robot manipulation. In this systems-focused paper, we introduce and document the development of a VR-based robot control paradigm with *manipulation assist* control algorithm, which allows human operators to specify larger manipulation goals while leaving the low-level details of positioning, manipulation and grasping to the robot itself. For the community, we also describe system design challenges to our progress thus far.

CCS CONCEPTS

• **Human-centered computing** → **Virtual reality**.

KEYWORDS

Virtual reality, robot, predictive control

ACM Reference Format:
Christian Barentine, Andrew McNay, Ryan Pfaffenbichler, Addyson Smith, Eric Rosen, & Elizabeth Phillips. 2021. A VR Teleoperation Suite with Manipulation Assist. In *Companion of the 2021 ACM/IEEE International Conference on Human-Robot Interaction (HRI'21 Companion), March 8-11, 2021, Boulder, CO, USA.* ACM, NY, NY, USA, 5 pages. https://doi.org/10.1145/3434074.3447210

1 INTRODUCTION AND PRIOR WORK

Robots are capable of fast, repetitive, and precise manipulation of objects both large and small, but are often limited by perception, planning, and control for manipulation tasks that require flexibility and re-planning. As a result, taking advantage of robot strengths while mitigating their drawbacks is a main appeal of teleoperation, or the direct remote control of robots by human operator(s). However, in order for humans to teleoperate robots well, they need high-fidelity control over the robot's actuators and an accurate and rich visualization of the robot's environment.

Advances in the capabilities of virtual technologies, as well as their rapid proliferation at consumer price points, have made it much easier to integrate them into existing robotic frameworks[4–6, 9, 10]. For instance, prior work showed that using a virtual reality (VR) interface which allowed users to teleoperate a Baxter robot's manipulators using waypoint-like control (called positional control), was faster and more accurate for both gross and fine motor manipulation tasks than a VR interface that mimicked directly "clicking and dragging" a Baxter's manipulators (called trajectory control) [2]. A study by Whitney et al. [8] showed that a VR interface allowed non-expert users to teleoperate a robot to complete a number of dexterous manipulation tasks faster and with lower cognitive workload than than traditional 2-D keyboard and monitor interfaces. The researchers also found that participants rated the VR interface more usable and assigned higher satisfaction scores to the VR interface than the keyboard and monitor interface. Lipton [3] conducted an informal user evaluation of a VR-robot teleoperation paradigm that employed a homunculus control interface. The homunculus model of the robot virtually embedded users in a "control room" inside the robot's "mind." The researchers asked users to control the robot to engage in a number of manufacturing and pick and place tasks with objects of different shapes and compliance.

Because several researchers have demonstrated the promise of using consumer grade VR hardware for teleoperated control of robots, there is value in continuing to develop and test VR-based interfaces to improve human control paradigms of robots and ultimately human-robot interactions. As a result, we are in the process of building upon the waypoint-like interface by creating a VR robot control paradigm with *manipulation assist*, a control assist algorithm, which allows a human operator to specify larger manipulation goals while leaving the details of positioning and manipulation to the robot itself. The purpose of this paper is to document the progress and report the system details of the manipulation assist VR system. We also describe for the community system design challenges to our progress thus far.

2 SYSTEM OVERVIEW

Due to the nature of teleoperation, our system has significant hardware and software components that need to operate in concert for successful execution documented in the following sections.

2.1 Hardware

The physical components of the system are:

- A Rethink Baxter Robot: Two armed pick and place
- An Oculus Quest VR headset and controllers
- A WLAN router, to connect to the Headset
- A network switch, to connect system components
- Multiple printed ArUco tags for object localization

HRI '21 Companion, March 8–11, 2021, Boulder, CO, USA

ACM ISBN 978-1-4503-8290-8/21/03…$15.00
https://doi.org/10.1145/3434074.3447210

- At least one Logitech USB Webcam, to recognize tags
- External laptop running computationally intensive packages

Figure 1: The Baxter robot used for the TagUp system.

2.2 Software

The software components of the system are:

- ROS Indigo for running other packages
- Ubuntu 14.04 for Indigo compatability
- TagUp VR Control Program, running on the Quest's onboard computer
- Unity Game Engine, to develop TagUp
- ZeroMQ sockets, bridge ROS and the Quest
- ArUco Tag recognition package
- MoveIt Inverse Kinematics solver

2.3 Object Pose Tracking

For the robot to assist with object manipulation, it must first be able to recognize what the object is and where in 3D space the object is relative to the robot itself. To meet this requirement, our system uses fiducial tags. Fiducial tags are high contrast patterns, similar in concept to a QR or bar code, though with more data redundancy built in. Each tag has an ID which the recognition software decodes in addition to the tag pose relative to the detecting camera. Our system uses a dictionary of tag IDs to map a tag's camera relative pose to its associated virtual object.

2.4 Manipulation assist algorithm

Once the system knows of an object's position in the world, the process of a manipulation assist is conceptually similar to following a set of waypoints. The VR control program contains a list of all the objects that have tags on them, as well as lists of what sorts of assists are possible for each object, and crucially, waypoints for the assist, stored relative to the object's center of mass.

When a user presses the assist button within range of an object that has possible assists, the control program first checks to see whether the permission mode is by *exception* or by *permission*. If it is by exception, the program selects the first valid assist for the closest object to the user's hand, and begins execution, pending an interupt signal from the user. If it is by permission, the program creates a menu in the user's field of view, prompting them to select a valid assist from the list presented. The robot begins execution after the selection is made. The actual process of an assist is the same in both cases. The robot stops listening to the user's position commands, and moves its end effector to each waypoint in the assist in turn, until all positions have been reached. After moving to the last waypoint, control is returned to the user.

2.5 VR environment

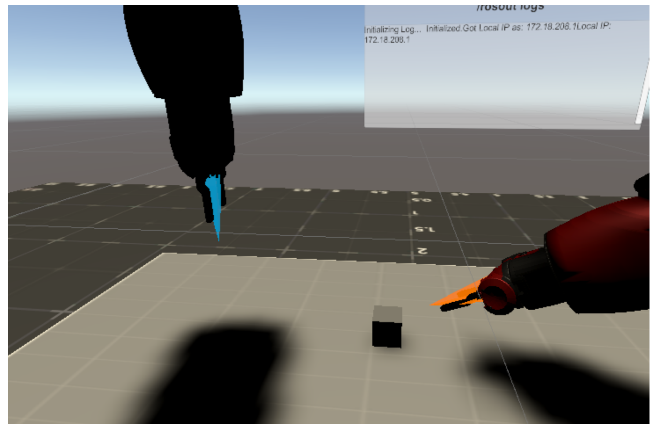

Figure 2: What the user sees as they control the Baxter. Note that the user has taken an egocentric position.

Currently, most of the VR environment is hard coded. A model of Baxter sits at the center of the space, with a grey box representing the real Baxter's work table. Assist targets are stored in a hard coded list, with an ID assigned to each. When one of the cameras in the work space detects a tag, the ID is looked up, and the corresponding object is placed in the scene, relative to the camera that detected it. If multiple cameras detect the same object, then an average of the positions is assigned as the object's position. These objects are visually represented by low polygon models of the corresponding physical objects. If a tag has not been detected after a certain timeout value, the virtual object is disabled until tracking is re-established.

3 CHALLENGES AND FUTURE WORK

While the *manipulation assist* of our system is quite promising, we struggled with effective means to reliably track object poses. While there are high performance tag systems available, such as LFTag[7] and STag[1], ArUco is the only package were were able to integrate into our system. In the future, we aim to use AI for our object pose estimation.

4 CONCLUSIONS

VR interfaces are promising for allowing users to more effectively teleoperate robots, especially for manipulation tasks. In this paper, we presented system details for a VR-based robot control system with *manipulation assist*, which allows users to specify high level manipulation goals while leaving the low-level goals to the robot itself. For the community, we detail the individual components of the system (hardware, software, and algorithm details) and progress and challenges to its development to date, as well as planned future development details.

5 ACKNOWLEDGMENTS

This research was funded by AFOSR Grant 16RT0881.

REFERENCES

[1] Burak Benligiray, Cihan Topal, and Cuneyt Akinlar. 2017. STag: A Stable Fiducial Marker System. *CoRR* abs/1707.06292 (2017). arXiv:1707.06292 http://arxiv.org/abs/1707.06292

[2] Rebecca Hetrick, Nicholas Amerson, Boyoung Kim, Eric Rosen, Ewart J de Visser, and Elizabeth Phillips. 2020. Comparing Virtual Reality Interfaces for the Teleoperation of Robots. In *2020 Systems and Information Engineering Design Symposium (SIEDS)*. IEEE, 1–7.

[3] Jeffrey I Lipton, Aidan J Fay, and Daniela Rus. 2018. Baxter's Homunculus: Virtual Reality Spaces for Teleoperation in Manufacturing. *IEEE Robotics and Automation Letters* 3, 1 (2018), 179–186.

[4] E. Rosen, D. Whitney, E. Phillips, G. Chien, J. Tompkin, G. Konidaris, and S. Tellex. 2017. Communicating robot arm motion intent through mixed reality head-mounted displays. In *International Symposium On Robotics Research*.

[5] Daniel Szafir, Bilge Mutlu, and Terrence Fong. 2014. Communication of intent in assistive free flyers. In *Proceedings of the 2014 ACM/IEEE international conference on Human-robot interaction*. ACM, 358–365.

[6] Daniel Szafir, Bilge Mutlu, and Terry Fong. 2015. Communicating directionality in flying robots. In *Proceedings of the Tenth Annual ACM/IEEE International Conference on Human-Robot Interaction*. 19–26.

[7] Ben Wang. 2020. LFTag: A Scalable Visual Fiducial System with Low Spatial Frequency. arXiv:2006.00842 [cs.CV]

[8] David Whitney, Eric Rosen, Elizabeth Phillips, George Konidaris, and Stefanie Tellex. 2020. Comparing robot grasping teleoperation across desktop and virtual reality with ROS reality. In *Robotics Research*. Springer, 335–350.

[9] Tom Williams, Daniel Szafir, Tathagata Chakraborti, and Heni Ben Amor. 2018. Virtual, augmented, and mixed reality for human-robot interaction. In *Companion of the 2018 ACM/IEEE International Conference on Human-Robot Interaction*. 403–404.

[10] Tom Williams, Daniel Szafir, Tathagata Chakraborti, and Elizabeth Phillips. 2019. Virtual, augmented, and mixed reality for human-robot interaction (vam-hri). In *2019 14th ACM/IEEE International Conference on Human-Robot Interaction (HRI)*. IEEE, 671–672.

,