Towards Multimodal Interactions: Robot Jogging in Mixed Reality

Enrico Sita
University of the West of England
Bristol, United Kingdom
enrico2.sita@live.uwe.ac.uk

Matthew Studley
University of the West of England
Bristol, United Kingdom
Matthew2.Studley@uwe.ac.uk

Farid Dailami
University of the West of England
Bristol, United Kingdom
Farid.Dailami@uwe.ac.uk

Anthony Pipe
University of the West of England
Bristol, United Kingdom
tony.pipe@brl.ac.uk

Trygve Thomessen
PPM AS
Trondheim, Norway
trygve.thomessen@ppm.no

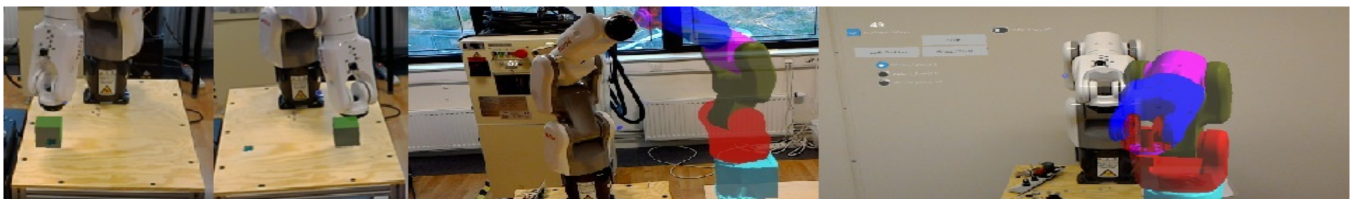

Figure 1: The view of the 3D models from the user point of view (through HoloLens). On the left, a cube that is moved by the real robot (NACHI MZ-04) through the definitions of waypoints. The 3D model of the NACHI MZ-04 that can be jogged with the user gestures (Middle). The real robot will then assume the target configuration.

ABSTRACT

The recent progress made in the field of Augmented Reality/Mixed Reality (AR/MR) has opened new possibilities and approaches to research areas that can benefit from 3D visualization of digital content in the real world.In fact, human-robot interaction design and the design of user interfaces have very much to gain from MR technologies. Nonetheless, designing the user-robot interaction and processing multimodal feedbacks are very challenging tasks. In this paper we focus in particular on interactions in mixed reality.

The main contribution of this paper is the implementation of a control system for an industrial manipulator through the user's interactions with MR content displayed with the Microsoft HoloLens. The system is based on the communication between Unity3D (used to design the user experience) and ROS, therefore extendible to any ROS-compatible robotic hardware.

CCS CONCEPTS

• **Human-centered computing → Mixed / augmented reality**; *Graphical user interfaces*;

KEYWORDS

Mixed Reality, Multimodal Interactions, Robotics, Unity3D, ROS

VRST '17, November 8–10, 2017, Gothenburg, Sweden

ACM ISBN 978-1-4503-5548-3/17/11.
https://doi.org/10.1145/3139131.3141200

ACM Reference format:
Enrico Sita, Matthew Studley, Farid Dailami, Anthony Pipe, and Trygve Thomessen. 2017. Towards Multimodal Interactions: Robot Jogging in Mixed Reality. In *Proceedings of VRST '17, Gothenburg, Sweden, November 8–10, 2017,* 2 pages.
https://doi.org/10.1145/3139131.3141200

1 INTRODUCTION & RELATED WORK

The number of research papers involved with AR/VR technologies and investigating their potential for human robot interactions is growing very rapidly. Despite some early attempts of using AR with multimodal feedbacks such as haptic data [Cha et al. 2005], most of the research in this field has mainly focused on enhancing visual feedbacks. In [Kruckel et al. 2015] for example, Kruckel et al. successfully provide a framework to teleoperate unmanned guided vehicles (UGVs) using an HMD, displaying the free-look camera view in virtual reality. Other researchers have been working on the integration of Unity3D and ROS ([Peppoloni et al. 2015]) to achieve more immersive teleoperation solutions. Related to grasping control techniques, another strategy based on the LMC has been presented by Jin et al. in [Jin et al. 2015] and most recently by Krupke et al. in [Krupke et al. 2016], where they developed a real-time gripper control with LMC in a Unity3D/ROS environment in virtual reality. We adopt an approach similar to [Peppoloni et al. 2015] which relies on the definition of waypoints, but we define the positions on the 3D model of the workpiece. Moreover, it is possible to interact with the 3D model of the robot and jog it to a custom position: we call this mode of operation *mixed reality teaching*. Once the configuration is satisfactory, using the UI or voice commands the robot will assume the target configuration.

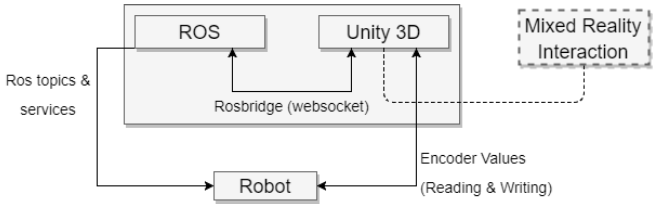

Figure 2: General architecture of the system.

We thus present a framework that allows the user to control a robot manipulator by interacting with MR 3D models displayed with the HMD.

2 SYSTEM DESCRIPTION

The current system architecture is sketched in Fig.2 and allows the user to interact with the robot in two ways. In MR teaching mode, the user can jog the robot by moving a target placed at the end effector of the 3D model of the robot. Alternatively, we can send the robot point to point instructions by interacting with a 3D digital model so that the robot can potentially reproduce the profile of such object. Although the naming may be deceiving, *teaching* here means that the user can instruct new positions to the robot by jogging it in MR. It does not refer to teaching the user how the robot can be controlled. The robot used in this work is a NACHI MZ-04 which although natively not ROS compatible, has been configured with a custom user application to be able to communicate internal variables to ROS.The user's inputs are handled by Unity3D, while the robot communication is made of two components that have been interfaced with Unity: the first is ROS-Unity, that manages inputs like 3D coordinates or gripper inputs; the second instead provides the robot's encoder values in real time through UDP communication.The communication with ROS is based on ROSbridge, which is maintained by the community. However, to interface Unity3D with ROS an additional module is necessary. In our case we adopt a C# websocket based communication similarly adopted in. In Unity3D instead, we designed the user's possible interactions: with the HoloLens the user can walk around a digital object, in our case a cube, and use the equivalent gesture of a "click" on its surface to issue a movement command to the robot. The 3D model of the robot is animated by a C# IK solver which is based on the work made by Starke in [Starke et al. 2016]. The user can drag a small cube that serves as target for the 3D robot, which will move in order to minimize the displacement between the end effector and the target.

3 IMPLEMENTATION

The HoloLens establish a UDP communication with the robot to access the robot's encoder values. The values are used to update the 3D model and send the correct configuration when the model is used to jog the real robot. Concretely, once the HoloLens application receives the encoder values, it operates the conversion to obtain the rotation of the joints in degrees to then apply it to the 3D model in augmented reality. The conversion from encoder representation to degrees is performed with the encoders' pulse and offset constants, which are typically located in the robot's internal storage or indicated in the manual. In waypoint mode, the space coordinates in the HoloLens are published to ROS as topics and interpreted by the robot as relative movements, which are added to the current pose. The C# implementation of ROSbridge running on the HoloLens is in charge of handling JSON strings used for the communication.

4 RESULTS AND DISCUSSION

The first implementation of the framework has been preliminarily tested. The parameter under test were the general performance in terms of frame rate of the application which resulted in a fairly constant rate of 60 FPS. It is important to notice that the HoloLens limits the application's maximum frame rate at exactly 60 FPS, putting a constraint on the maximum achievable update frequency of the encoder values. The preliminary test was conducted with the robot, the HoloLens and the PC serving as ROS server connected wireless on a 100Mb network. Further tests will aim at evaluating the usability of the framework, in terms of the accuracy with which the robot tracks the configuration defined on the 3D model and the waypoints defined.

5 CONCLUSION AND FUTURE WORK

In this paper we presented the preliminary results of a control system based on user's interactions with a digital model in mixed reality. The robot can operate in two modes: through point to point instructions stored into ROS topics and updated by the user with the HMD (HoloLens), or through direct manipulation of a target which the robot will track. We intend to work on the synchronization of the digital model on top of the real robot. In this way, we hope to achieve a more immersive interaction with the system. The digital robot will only appear as feedback once the user starts interacting with the 3D target, which will be rendered in correspondence of the real end effector.

ACKNOWLEDGMENTS

This research was partially funded by the Norwegian Research Council.

REFERENCES

Jongeun Cha, Ian Oakley, Junhun Lee, and Jeha Ryu. 2005. An AR System for Haptic Communication. In *Proceedings of the 2005 International Conference on Augmented Tele-existence (ICAT '05)*. ACM, New York, NY, USA, 241–242. https://doi.org/10.1145/1152399.1152444

Haiyang Jin, Liwei Zhang, S. Rockel, Jun Zhang, Ying Hu, and Jianwei Zhang. 2015. A novel optical tracking based tele-control system for tabletop object manipulation tasks. In *2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. 636–642. https://doi.org/10.1109/IROS.2015.7353439

Kai Kruckel, Florian Nolden, Alexander Ferrein, and Ingrid Scholl. 2015. Intuitive visual teleoperation for UGVs using free-look augmented reality displays. 4412–4417. http://opus.bibliothek.fh-aachen.de/frontdoor/index/index/docId/7412

Dennis Krupke, Lasse Einig, Eike Langbehn, Jianwei Zhang, and Frank Steinicke. 2016. Immersive Remote Grasping: Realtime Gripper Control by a Heterogenous Robot Control System. In *Proceedings of the 22Nd ACM Conference on Virtual Reality Software and Technology (VRST '16)*. ACM, New York, NY, USA, 337–338. https://doi.org/10.1145/2993369.2996345

L. Peppoloni, F. Brizzi, C. A. Avizzano, and E. Ruffaldi. 2015. Immersive ROS-integrated framework for robot teleoperation. In *2015 IEEE Symposium on 3D User Interfaces (3DUI)*. 177–178. https://doi.org/10.1109/3DUI.2015.7131758

S. Starke, N. Hendrich, S. Magg, and J. Zhang. 2016. An efficient hybridization of Genetic Algorithms and Particle Swarm Optimization for inverse kinematics. In *2016 IEEE International Conference on Robotics and Biomimetics (ROBIO)*. 1782–1789. https://doi.org/10.1109/ROBIO.2016.7866587