Enabling Human-Robot-Interaction for remote robotic operation via Augmented Reality

Chung Xue Er (Shamaine)
Computer & Software Engineering
Faculty of Engineering & Informatics
Athlone Institute of Technology
Athlone, Ireland
xechung@research.ait.ie

Yuansong Qiao
Software Research Institute
Faculty of Engineering & Informatics
Athlone Institute of Technology
Athlone, Ireland
ysqiao@research.ait.ie

Niall Murray
Computer & Software Engineering
Faculty of Engineering & Informatics
Athlone Institute of Technology
Athlone, Ireland
nmurray@research.ait.ie

***Abstract*— Human-Robot Interaction (HRI) will be a crucial component of smart factories of the future (FoF). This demo presents a reliable and cost-effective HRI system based on Augmented Reality (AR). In this demonstration, we offer concepts and methods currently being developed to enable high-level human-robot collaboration and interaction. The user interaction and instructions are captured via AR and communicated to a Robot Operating System (ROS) powered robotic arm.**

Keywords*— *Augmented Reality, Unity3D, Robot Operating System, Human-Robot Interaction

I. Introduction

Human-Robot Interaction (HRI) is the study of how a human communicates with a robotic system. It informs us of how to understand, design, evaluate, and implement robotic systems that can aid human operators in collaborative chores [1],[2]. In certain circumstances, to achieve effective HRI, the robot needs to be positioned relative to a human while performing manipulation and execution actions. Current research in HRI seeks to discover less training-intensive, more direct, safe, effective, natural, and stress-free interactions [2].

A novel approach of using Augmented Reality (AR) creates a new design space and opportunity to facilitate more naturalistic teleoperation [3]. On account of the ability of AR to augment the operator view and visualise the corresponding robotic task space, the operator can retrieve the robot's sensor state information (temperature level, position, joint angle, etc.) via graphic overlays. This results in an AR interface with great potential for robotic teleoperation [3].

In this context, this demo paper outlines an early-stage work, where AR technologies can be utilised by operators with limited background knowledge in robotic programming to visualise and control the simulated robot arm from a safe distance. We are researching applications of this interface for smart manufacturing. Instead of implementing the AR system on a real robotic arm, the HRI system is tested on a virtual robotic arm in the Gazebo simulator [4] via Robot Operating System [5]. Our main objectives are: to develop and evaluate this AR-based HRI; to reduce the cost of developing and performing real robot experiments; to identify its constraints and capability through user QoE.

II. Related Work

Industrial safety standards and regulations require an operator to control the robot at a safe distance. Therefore, a considerable amount of research has been carried out to develop a less training-intensive, more direct, safe, effective, natural, and stress-free interaction [6]. In [7], Chacko et al. presented a smartphone-based AR approach to detect and localise a 4 DOF manipulator robot on a planar surface by using Google's ARCore framework. It successfully performs a simple pick-and-place task within the robot's workspace with the aid of a 2D marker. However, this system was constrained in that the user needs to perform the gestures with one hand whilst holding the mobile phone in the other. The work also highlighted challenges in identifying complex shapes or picked object-orientation due to limitations of the mobile phone camera.

Similar to our approach, in [8] a user could execute a robot trajectory through free space or in contact with a surface using the Microsoft HoloLens (in our case, we will deploy the HoloLens 2). The user can move the 7DOF robot's end-effector along a path with an MYO armband and speech recognition. The user sets multiple waypoints by altering the orientation of their head and speech command. Although the system required less physical effort to train, the AR robotic interface needed a higher mental effort. The poor 3D environmental reconstruction from HoloLens resulted in difficulty for accurate depth estimation and higher registration error after constant user position displacements.

III. Technologies and System Architecture

The proposed HRI system architecture is presented in Fig. 1. It includes Gazebo, a robotic simulator running on a virtual Ubuntu 16.04 LTS ROS Kinetic machine. Our AR system deployed Unity 2018.4.12f1 (64-bit) as our base architecture. It has been developed into a very versatile AR development tool used by different devices and platforms, ranging from mobile devices to Windows Mixed Reality headset (HoloLens). The Microsoft HoloLens 2 AR Head Mounted Display (HMD) is the interface for users to visualise and interact with the hologram of the 3D robotic arm model. The integration of ROS and Unity 3D is enabled via ROS-Sharp[9], through a WebSocket connection on port 9090.

978-1-7281-7374-0/20/$31.00 ©2020 IEEE
DOI 10.1109/WoWMoM49955.2020.00046

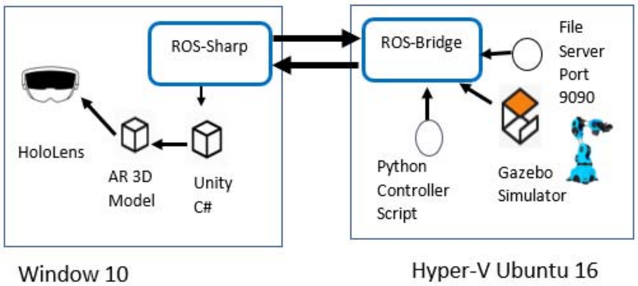


Fig. 1. End-to-end HRI System Architecture

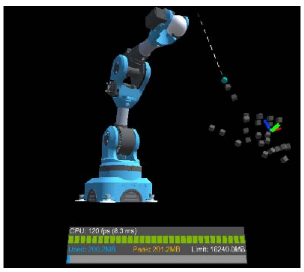


Fig. 2. Game View of Virtual robot arm with MRTK in Unity 3D

The original robot arm URDF model was imported from ROS through ROSBridge Client while the UrdfImporter converts the arm's Universal Robotic Description Format (URDF) to a Unity game object. In this context, ROS act as a publisher while Unity as a subscriber. Unity created a topic with pose stamped publisher script to send end-effector pose (position and orientation) in real-time to ROS over ROSBridge. Unity achieved this by converting C# data structures into JSON messages and passing the pose data across the bridge. The ROSBridge library, which contains core ROSBridge package, will then retrieve the pose data through the same topic by converting the JSON message back to geometry messages. In this case, Unity will act as a publisher while the ROS simulator as a subscriber.

A. Unity3D

Unity3D real-time development game engine is deployed. Powerful tools are provided by Unity to create appealing AR experiences such as the Mixed Reality Toolkit (MRTK)[9]. This toolkit is used to develop HoloLens applications. The Microsoft HoloLens ARHMD is employed because of its built-in constant simultaneous localisation and mapping (SLAM) ability in addition to its self-supporting nature. The ROS sharp folder [10] contains a RosBridgeClient and UrdfImporter. The Forward and Backward Reaching Inverse Kinematics (FABRIK) algorithm has been used in Unity virtual robotic arm to solve its inverse kinematics. FABRIK is a new method discovered by [11] in 2011 to solve the problem of inverse kinematics. A FinalIK plugin in Unity3D created by Pärtel Lang has provided complete documentation of FABRIK algorithm for Unity developers to integrate FABRIK algorithm easily into their project [12].

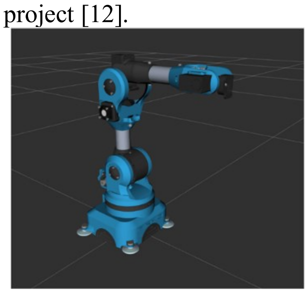


Fig. 3. Side View of Gazebo simulation scene

B. Robot Operating System

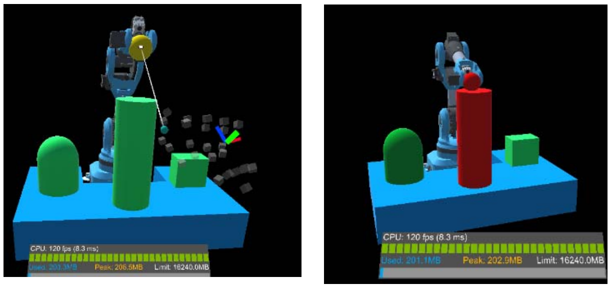
ROS is a versatile, multilingual middleware that provides a collection of tools and libraries for robotic developers to simplify their effort of designing complicated and robust robot actions [5]. ROS is open-source and is language agnostic. A developer can write "a node" in any programming language, although it is primarily developed in C++ and Python [13]. A node in ROS represents a separate program running a specific task. Each standalone task can be divided into different nodes that connect each other across channels. These so-called channels are described as topics in [13]. A publisher allocates data into a ROS topic, whilst a subscriber retrieves the data from a topic.

MoveIt is a ROS-based motion planning framework that provides an inverse kinematics solver using inverse Jacobian methods [14]. It allows the user to plan a collision-free path of the arm from point A to point B. We developed a Python script that subscribes to the same topic where Unity publishes the end-effector pose to the simulated robot across the ROSBridge. We run the script and wait for the Unity to publish the pose messages in geometry format (position translation x, y and z, and quaternion rotation x, y, z, and w). Gazebo serves as an open-source robotics simulator while ROS as an interface for the robot. The benefits of using Gazebo is its ready integration with ROS [4].

IV. DEMONSTRATION

The current capability of this system can be divided into two categories. First and foremost, it can publish the robot's real-time pose information from Unity to ROS robot arm. A yellow target sphere object is created and attached to the robot arm end effector in Unity3D. When the user clicks on the play button and drags the target sphere in 3D space, the robot arm in Unity and ROS will follow the sphere direction in real-time.

Second, a sophisticated 3D work environment is created by modelling separate parts of the 3D game object and merging them together, as shown in Fig. 4(a). A script was written in C sharp to detect collision. When the user drags the sphere and accidentally collide with a 3D obstacle, both obstacle and the target sphere will turn red, as depicted in Fig. 4(b). This condition indicates that a collision has been detected. This feature is useful for AR path planning preview where the user would be able to preview and evaluate the motion planned before sending the pose information to a physical robot arm for execution.



(a) (b)

Fig. 4. Experimental conditions: (a)Original simulation scene without a collision b) Obstacle and sphere turn red when a collision is detected

V. Conclusion and Future Work

In this demo, we showcase an early design and development of AR system prototype based on the idea of safe human-robot teleoperation. An early stage of collision detection, as shown in Fig.4(b) is considered as one of the most sensible solutions to ensure smooth trajectories along specified paths for real-world industry robot. Through this work, we explore and develop the compatibility solution for the latest Hololens2 HMD application with the open-source MRTK v2 toolkits [9] and external libraries. The results of the simulation experiment demonstrate that our proposed method can be further improved through the implementation of image registration.

In the future, four buttons labelled with “Plan”, “Preview”, “Reset” and “Execute” that has interactive behaviour attached to it will trigger the main actions of the virtual robot arm. The “Plan” button consists of a drop-down bar which allows the operator to choose which waypoint systems to use. Once the desired waypoint system has been selected, the operator could start to plot the end-effector’s start and endpoint. The “Preview” button will let the operator visualise the planned robot motion trajectory. If the operator is satisfied with the robot motion trajectory, he or she could press the “Execute” button. “Execute” button will trigger the communication bridge by sending the pose of the Unity robot arm to the ROS robot arm. Once complete and inspired by [15][16][17], the QoE framework will be employed to execute a suite of user studies to identify system constraints and capability.

ACKNOWLEDGEMENT

This publication has emanated from research conducted with the financial support of Science Foundation Ireland (SFI) under grant number SFI/16/RC/3918, co-funded by CONFIRM Smart Manufacturing and Robotics & Drives Ireland

References

[1] D. Feil-Seifer and M. Mataric, “Human-Robot Interaction,” Encyclopedia of Complexity and Systems Science, 2019, pp4643-4659, doi: 10.1007/978-0-387-30440-3_274.

[2] M. A. Goodrich *et al.*, "Human-robot interaction: a survey: Foundations and Trends in Human-Computer Interaction: Vol 1, No 3", Dl.acm.org, 2020. [Online]. Available: https://dl.acm.org/doi/10.1561/1100000005

[3] S. Green, M. Billinghurst, X. Chen and J. Chase, "Human Robot Collaboration: An Augmented Reality Approach A Literature Review and Analysis," Proceeding of MESA 07:3RD International Conference on Mechatronics and Embedded System and Applications, Nevada,USA,2007.

[4] M. Santos Pessoa de Melo, J. Gomes da Silva Neto, P. Jorge Lima da Silva, J. M. X. Natario Teixeira and V. Teichrieb, "Analysis and Comparison of Robotics 3D Simulators," 2019 21st Symposium on Virtual and Augmented Reality (SVR), Rio de Janeiro, Brazil, 2019, pp.242-251.

[5] "ROS.org | About ROS", Ros.org, 2020. [Online]. Available: https://www.ros.org/about-ros/

[6] Przemyslaw A. Lasota; Terrence Song; Julie A. Shah, "A Survey of Methods for Safe Human-Robot Interaction," A Survey of Methods for Safe Human-Robot Interaction , 2017.

[7] S. M. Chacko and V. Kapila, "An Augmented Reality Interface for Human-Robot Interaction in Unconstrained Environments," 2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Macau, China, 2019, pp. 3222-3228.

[8] C. P. Quintero, S. Li, M. K. Pan, W. P. Chan, H. F. Machiel Van der Loos and E. Croft, "Robot Programming Through Augmented Trajectories in Augmented Reality," 2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Madrid, 2018, pp.1838-1844.

[9] "Getting started with MRTK version 2 - Mixed Reality", Docs.microsoft.com, 2020. [Online]. Available: https://docs.microsoft.com/en-us/windows/mixed-reality/mrtk-getting-started

[10] "siemens/ros-sharp", GitHub, 2020. [Online]. Available: https://github.com/siemens/ros-sharp/wiki

[11] A. Aristidou and J. Lasenby, "FABRIK: A fast iterative solver for the Inverse Kinematics problem", Graphical Models, vol. 73, no. 5, 2011, pp. 243-260

[12] "Final IK | Animation Tools | Unity Asset Store", Assetstore.unity.com, 2020.[Online].Available: https://assetstore.unity.com/packages/tools/animation/final-ik-14290

[13] "ROS Basics 1 - Nodes, Topics, Services & Actions", Tarang Shah - Blog, 2019. [Online]. Available: https://tarangshah.com/blog/2017-04-01/ros-basics-1-nodes-topics-services-actions

[14] "MoveIt Tutorials — moveit_tutorials Kinetic documentation", Docs.ros.org, 2020. [Online]. Available: http://docs.ros.org/melodic/api/moveit_tutorials/html/index.html

[15] T. B. Rodrigues *et al.*, "An evaluation of a 3D multimodal marker-less motion analysis system | Proceedings of the 10th ACM Multimedia Systems Conference", Dl.acm.org, 2020. [Online]. Available: https://dl.acm.org/doi/10.1145/3304109.3306236

[16] C. Keighrey *et al.*, "Comparing User QoE via Physiological and Interaction Measurements of Immersive AR and VR Speech and Language Therapy Applications | Proceedings of the on Thematic Workshops of ACM Multimedia 2017", Dl.acm.org, 2019. [Online]. Available: https://dl.acm.org/doi/10.1145/3126686.3126747

[17] E. Hynes, R. Flynn, B. Lee and N. Murray, "A Quality of Experience Evaluation Comparing Augmented Reality and Paper Based Instruction for Complex Task Assistance," 2019 IEEE 21st International Workshop on Multimedia Signal Processing (MMSP), Kuala Lumpur, Malaysia, 2019, pp. 1-6, doi: 10.1109/MMSP.2019.8901705.