Available online at www.sciencedirect.com

ScienceDirect

IFAC PapersOnLine 58-19 (2024) 581–585

An Augmented Reality Interface for Safer Human-Robot Interaction in Manufacturing

Igor Rybalskii * Karl Kruusamäe * Arun Kumar Singh *
Sebastian Schlund **

* University of Tartu, Estonia
(e-mail: igor.rybalskii@ut.ee; karl.kruusamae@ut.ee; arun.singh@ut.ee).
** TU Wien, Austria
(e-mail: sebastian.schlund@tuwien.ac.at)

Abstract: With the adaptation of Industry 4.0 and emerging ideas of Industry 5.0, the manufacturing industry is shifting back towards human-centered production. In these scenarios, collaboration robots (Cobots) become an important part of this shift toward human-robot collaboration (HRC). However, with the introduction of robots meant to work in close proximity to and in collaboration with people, problems of safety and collaboration efficiency arise. On the other hand, there are developments in the field of Augmented Reality (AR), which allow overlaying digital information onto the physical environment and having people interact with it. This paper introduces an AR interface for HRC with cobots, which can tackle the issues of human safety and efficiency. The use of the interface is demonstrated on an HRC ski assembly use case. All the source code can be found on GitHub: https://github.com/ut-ims-robotics/rybalskii-incom-2024-replication-package

Keywords: In-process Manufacturing Monitoring, Human-Automation Integration, Robots manipulators, Security and safety of HMS, Augmented Reality

1. INTRODUCTION

1.1 Human-Robot Collaboration (HRC)

With Industry 4.0's advancements in cloud technologies, smarter robots, and analytics capabilities (European Parliament and Directorate-General for Internal Policies of the Union et al. (2016)) being implemented and extensively researched since the term's inception (Paraskevopoulos (2022)), and with Industry 5.0 aiming to prioritize human-centric manufacturing processes (European Commission and Directorate-General for Research and Innovation and Breque et al. (2021)), scenarios where humans and robots work closely together on the same tasks can be expected. Such work scenarios bring up the issues of safety and efficiency of human-robot communication.

Works such as Menon and Vidalis (2021) look at the safety and security with a more general engineering perspective. According to Nuseibeh (2001), those safety and security requirements, in combination with architectural elements, are crucial in designing the manufacturing system.

Based on this information, Hosseini et al. (2023) proposes a methodology on how to create and later configure HRC use cases according to customers' needs. This methodology was used to modify the ski assembly use case to allow the robot and human to work on the same task at the same time instead of working on the same task, but only one at a time. (Figs. 1 and 2).

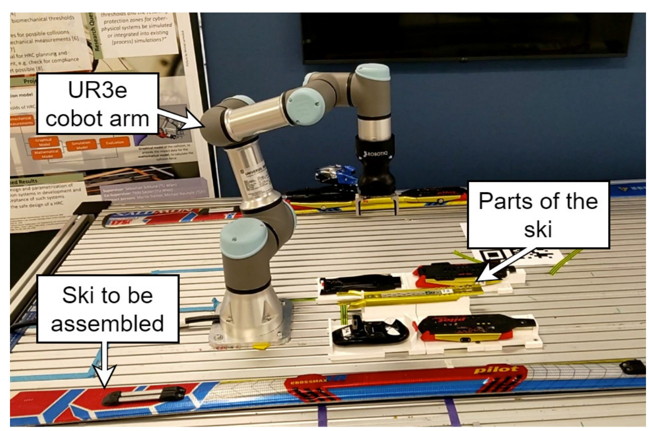

Fig. 1. Assembly station view from the operator perspective, consisting of cobot arm, ski to be assembled, and parts, laying on custom-made holder, to be picked by cobot arm

1.2 Augmented Reality (AR)

Augmented reality is a perception of digital information visualized onto physical objects (IEEE Digital Reality (2024)). Among several AR technologies, one of the approaches is Head Mounted Displays (HMD) (Suzuki et al. (2022)). They usually have a transparent screen to visualize information directly over the physical world (Fang et al.

⋆ This work is in part supported by the project "Increasing the knowledge intensity of Ida-Viru entrepreneurship" co-funded by the European Union and the grant PSG753 from Estonian Research Council.

10.1016/j.ifacol.2024.09.275

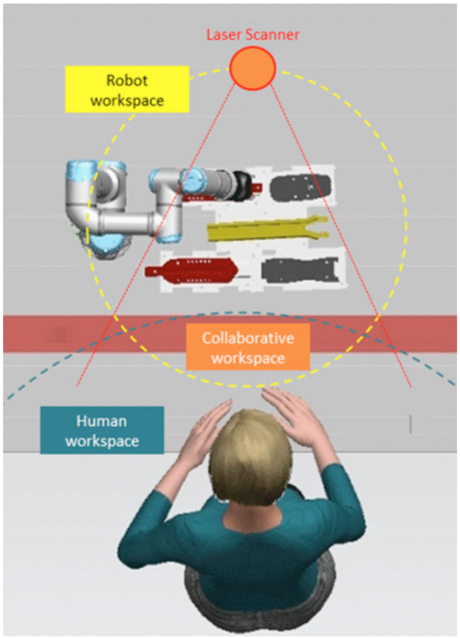

Fig. 2. Figure of configured HRC use case for ski assembly (Hosseini et al. (2023))

(2023)). It stands out due to the ability to overlay information directly into the user's field of view, thus seamlessly integrating digital cues and the physical environment.

1.3 Contribution

This paper proposes the AR interface built for HMD AR glasses, which could visualize the information about cobot intent to a human, thus improving safety.

The contribution of this work is the adaptation of the use case from Hosseini et al. (2023) and the creation of a novel AR interface, which allows the human operator to preview the cobot's movement before it starts, having the potential to significantly improve the safety in HRC scenarios. By making the source code publicly available [1], we also lay the groundwork for the interface to be further iterated and expanded, depending on manufacturing processes.

2. PREVIOUS WORK

2.1 Safety and Security requirements

In their previous work, Hosseini et al. (2023) go through several state-of-the-art approaches for identifying different safety and security dependencies and conflicts between them. These include:

(1) System Modelling Language (SysML) for Requirement Traceability, allowing to make and trace requirement changes during the complete life-cycle of the process and how those changes can affect other architecture elements and vice-versa (Wiegers and Beatty (2013)). SysML is a language for system modeling, allowing to link requirements with system elements (Vidal and Villota (2018)).

(2) For Categorizing Safety and Security Dependencies, it was pointed out that according to Kavallieratos et al. (2020), there are two main classes of approaches: **Unified Approach**, where Safety and Security are considered a single term, and **Integrated Approach**, where Safety and Security are dealt with separately. It is also pointed out that according to Kriaa et al. (2015), there can be four different dependencies between safety and security: Conditional dependency, Mutual Reinforcement, Conflict Dependency, and Independence.

(3) For Safety and Security Requirements Conflicts Detection and Resolution stated that according to Pradeep and Kant (2022), there are three approaches: rule-based, taxonomy-based, and ontology-based. According to Chaki et al. (2020), identified conflicts can be resolved dynamically on run-time or statically during the design process. According to Robinson et al. (2003), this can be done by either of the following: Elimination, Prioritization, Refinement, Postponement.

(4) For the Safety and Security of Human-Robot Interaction, Hosseini et al. (2023) discusses several safety standards, such as ISO/DIS 10218-2:2020 and ISO/TS 15066:2016. Based on those standards and literature (Malik and Bilberg (2019)), several levels of interactions are considered: Coexistence, where robot and human are next to each other with no fences but in separate workspaces; Synchronization, where robot and human have same workspace but use it separately at different time; Cooperation, where human and robot can enter the workspace at the same time, but work on different work-pieces; Collaboration, where human and robot work on the same workplace at the same time on the same work-piece.

2.2 Methodology

Based on the considerations for safety and security, Hosseini et al. (2023) proposes the following methodology for requirement interaction management (Fig. 3). It focuses on dividing the configuration and all existing requirements and design specifications into smaller partitions based on goals, functions or other requirements, which allows to focus on specific parts. After that the relationships between the requirements and design elements are identified per partition. After that, all the changes in the processes are made, and emerging conflicts are identified and resolved, leading to a new configuration.

2.3 Reconfiguration and results

To showcase the developed methodology, the ski assembly use case was reconfigured according proposed methodology. The original configuration is called **Synchronization**, where each assembly part is handled one at a time, with a human stopping the robot when needed. The second one is called **Collaboration**, which allows both the human and the cobot to work on the assembly simultaneously, with the cobot picking a new part while the human still attaches the previous one.

During the reconfiguration, two conflicts were found:

[1] https://github.com/ut-ims-robotics/rybalskii-incom-2024-replication-package

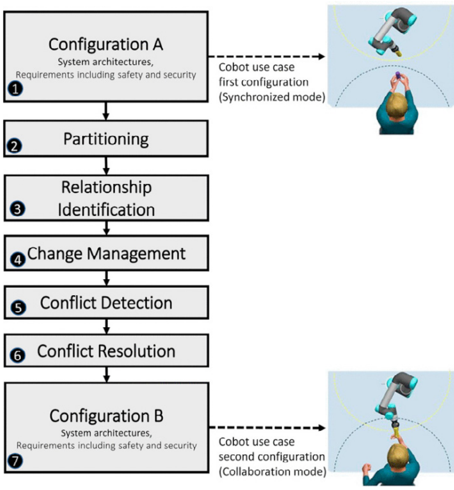

Fig. 3. Methodology for the Requirement Interaction Management (Hosseini et al. (2023))

(1) The first one was between the requirement for the robot's safety limits not to be changed and the requirement for the system administrator to be able to update the configuration remotely. In this conflict, the first requirement took priority, and the second requirement was limited to non-safety configurations.
(2) The second one was between the requirement for the system to be able to detect failures through the cloud and the requirement for cloud-related information to be protected from unauthenticated users. In this case, the first requirement was refined, so failure detection is to be done on the workstation and not the cloud.

This use case resulted in two takeaways. The first one is to avoid the remote access to system's settings to not compromise security and potential safety. The second one is to use cloud computing with caution. Developed methodology allows to see the clear picture and resolve any potential conflicts with safety and security requirements.

3. DESCRIPTION OF THE SETUP

In this paper, the Collaboration configuration of the setup is adapted, meaning both the operator and the robot work in the same workspace at the same time (Fig. 4).

During the assembly task, the operator is wearing MS HoloLens 2, which is wirelessly connected to the computer controlling the cobot. Information about the robot's current state and future actions are visualized in AR to the operator. By seeing the cobot's intent, the human operator can see which part of the working area needs to be avoided in advance, thus reducing the risk of the cobot colliding with the human. By warning the operator about cobot movement in advance, the cobot's speed can also be increased, thus having the potential to increase the overall task execution speed.

The setup consists of:

- UR3e cobot with 6 Degrees of Freedom (DOF) and a payload of 3kg [2]
- Hand-E Adaptive Gripper by Robotiq [3]

[2] https://www.universal-robots.com/products/ur3-robot/
[3] https://robotiq.com/products/hand-e-adaptive-robot-gripper

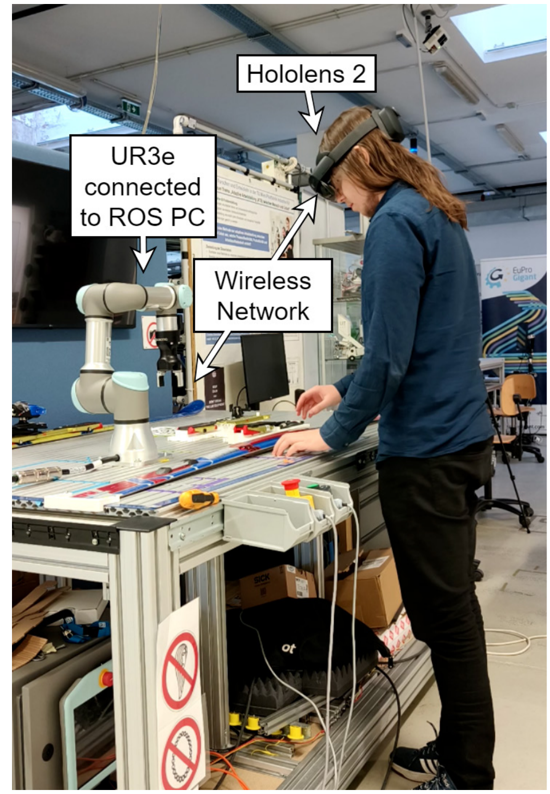

Fig. 4. Updated AR setup with AR interface

- 3D printed assembly parts holder (Naggay (2021))
- External laptop with Ubuntu 20.04 and ROS1 Noetic, controlling the robot
- Wireless network router
- Microsoft Hololens 2 AR headset capable of tracking the position and orientation of the user's hands and head while projecting digital information relative to the real-world environment [4].

4. DESCRIPTION OF THE AR INTERFACE

4.1 Overview of used tools

All the source code for the solution presented in this publication can be found in the GitHub repository [1].

The application was built with Unity 2020.3. This engine was chosen due to its popularity (SteamDB (2024)), support for AR development (Unity Technologies (2023)), and is free until yearly revenue is below 100 thousand dollars (Unity Technologies (2024)).

Microsoft Mixed Reality Toolkit (MRTK) was used within Unity to develop for Hololens 2, as it provides all the necessary tools to develop applications with user input and spatial interactions (polar kev (2022)).

UR3e is controlled with ROS1 Noetic [5] and MoveIt Motion Planning Framework [6] which allows the creation of motion plans for configured cobots.

[4] https://www.microsoft.com/en-us/hololens
[5] https://www.ros.org
[6] https://moveit.ros.org

To allow communication between cobot controlled by ROS and the interface, ROS# created by Siemens[7] and Rosbridge[8] server are used. However, because Hololens can run only with Universal Windows Platform (UWP) applications, the fork by EricVoll[9] is used instead of the Siemens version.

To visualize information relative to the real environment, QR codes were chosen as anchors. Hololens 2 can locate and read them by default. To utilize this functionality within Unity, a package created by Microsoft was used (qianw211 (2022)).

4.2 Interface Architecture

This section describes the communication framework and visual feedback presented to the operator, as illustrated in Figure 5. Cobot uses MoveIt to generate trajectories on run-time in a predetermined order. Before the trajectory is executed, cobot's end-effector positions are extracted and sent to Hololens through the Rosbridge server and TCP connection. These positions are used to create the robot movement hologram, which is anchored to the real robot setup through a QR code and a predefined offset. Upon receiving the position information, Hololens sends back a confirmation message. After receiving this confirmation message, MoveIt starts the trajectory execution. When the trajectory is executed, the message to clear the displayed hologram is sent to Hololens. When Hololens sends back the message that the "clear" message was received, the next goal is taken from the list, and a new plan is generated, repeating the previous cycle.

4.3 The resulting AR user experience

When wearing the headset and launching the application, the operator is able to start performing the collaborative assembly task. Before the task is started, the operator sees the cobot with no data added through AR (Fig. 6A). During the task, red spheres linked to the cobot's gripper position appear, showing the cobot's future movement trajectory (Fig. 6B), and after waiting for 0.5 seconds, the cobot will start moving (Fig. 6C). Once the cobot finishes the planned task, spheres are removed, to be replaced with new ones before cobot start the next task (Fig. 6D)

5. CONCLUSION

This paper introduces the Augmented Reality interface prototype, which allows human operators to preview the intention of the cobot 1-2 seconds ahead of time while executing the collaborative ski assembly. However, in the future, the switch to ROS2 should be considered, as it can provide better data security, which is crucial in the manufacturing environment.

The effectiveness of the AR interface in improving operational safety and efficiency remains to be empirically validated through comprehensive user studies. Additionally, further investigation is required to understand the impact of data transfer delays, particularly in relation to the volume of data being communicated simultaneously.

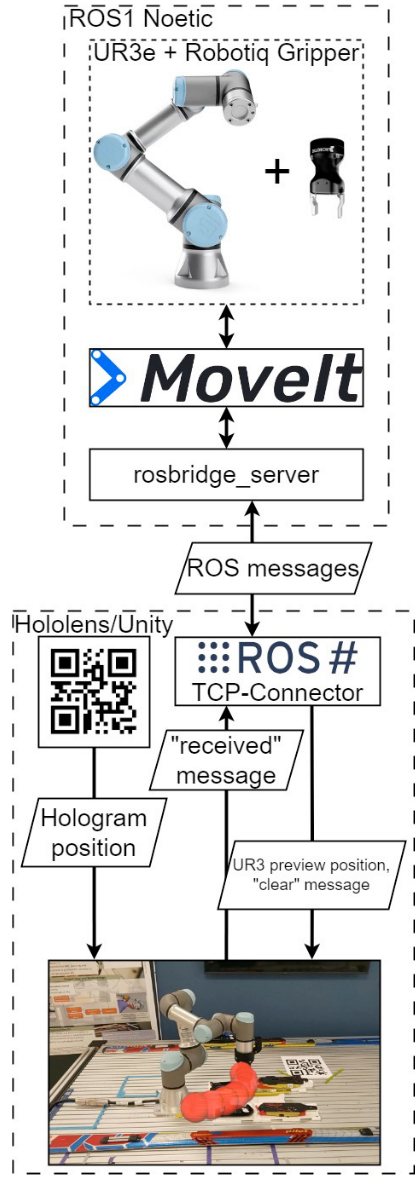

Fig. 5. The architecture of communication between UR3e with ROS1 Moveit and Hololens2 and what the operator sees during work

As the interface continues to be developed, it is important to explore the scalability across different collaborative scenarios and its adaptability to various cobot models and tasks. The ultimate goal is to develop a robust, secure, and efficient AR interface that can be widely adopted in diverse industrial settings

REFERENCES

Chaki, D., Bouguettaya, A., and Mistry, S. (2020). A conflict detection framework for iot services in multi-resident smart homes. In *2020 IEEE International Conference on Web Services (ICWS)*, 224–231. doi: 10.1109/ICWS49710.2020.00036.

European Commission and Directorate-General for Research and Innovation and Breque, M., De Nul, L., and Petridis, A. (2021). *Industry 5.0 – Towards a*

[7] https://github.com/siemens/ros-sharp
[8] https://wiki.ros.org/rosbridge_suite
[9] https://github.com/EricVoll/ros-sharp

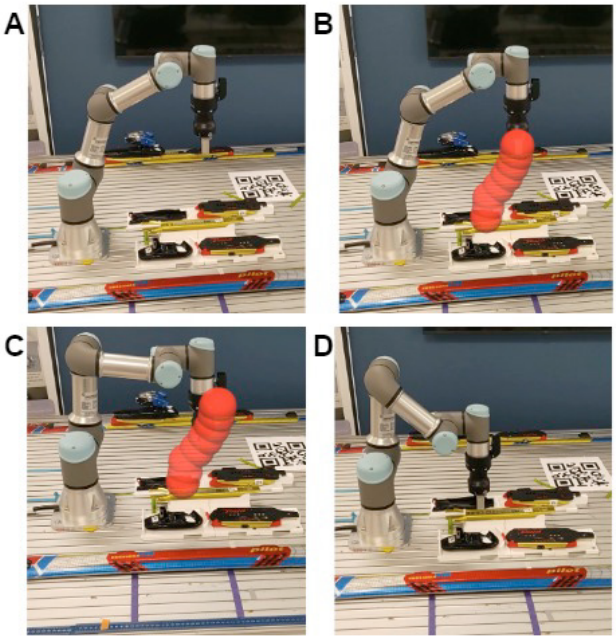

Fig. 6. View of the cobot performing its part of the task. (A) In the beginning of the task, the operator sees the cobot with no hologram. (B) Then a red hologram appears that visualizes the future path of cobot's gripper. (C) Subsequently, the cobot moves along the visualized trajectory. (D) When the trajectory is fully completed, the corresponding AR visualization is removed

sustainable, human-centric and resilient European industry. Publications Office of the European Union. doi:10.2777/308407.

European Parliament and Directorate-General for Internal Policies of the Union, Carlberg, M., Kreutzer, S., Smit, J., and Moeller, C. (2016). *Industry 4.0*. European Parliament. doi:10.2861/947880.

Fang, W., Chen, L., Zhang, T., Chen, C., Teng, Z., and Wang, L. (2023). Head-mounted display augmented reality in manufacturing: A systematic review. *Robotics and Computer-Integrated Manufacturing*, 83, 102567. doi:10.1016/j.rcim.2023.102567.

Hosseini, A.M., Fischer, C., Bhole, M., Kastner, W., Sauter, T., and Schlund, S. (2023). A safety and security requirements management methodology in reconfigurable collaborative human-robot application. In *2023 IEEE 19th International Conference on Factory Communication Systems (WFCS)*, 1–8. doi:10.1109/WFCS57264.2023.10144233.

IEEE Digital Reality (2024). Standards - ieee digital reality. URL `https://digitalreality.ieee.org/standards`.

Kavallieratos, G., Katsikas, S., and Gkioulos, V. (2020). Cybersecurity and safety co-engineering of cyberphysical systems—a comprehensive survey. *Future Internet*, 12(4). doi:10.3390/fi12040065.

Kriaa, S., Bouissou, M., Piètre-Cambacedes, L., and Halgand, Y. (2015). A survey of approaches combining safety and security for industrial control systems. *Reliability Engineering & System Safety*, 139, 156–178. doi:10.1016/j.ress.2015.02.008.

Malik, A.A. and Bilberg, A. (2019). Developing a reference model for human–robot interaction. *International Journal for Interactive Design and Manufacturing (IJIDeM)*. doi:10.1007/s12008-019-00591-6.

Menon, C. and Vidalis, S. (2021). Towards the resolution of safety and security conflicts. In *2021 International Carnahan Conference on Security Technology (ICCST)*, 1–6. doi:10.1109/ICCST49569.2021.9717390.

Naggay, A. (2021). Enabling a flexible use of a stationary cobot. Supervised by Univ.-Prof. Dr.-Ing. Sebastian Schlund and Dipl.-Ing. Clara Fischer.

Nuseibeh, B. (2001). Weaving together requirements and architectures. *Computer*, 34(3), 115–119. doi:10.1109/2.910904.

Paraskevopoulos, D. (2022). The rise of Industry 4.0 in 5 stats. URL `https://iot-analytics.com/industry-4-0-in-5-stats/`.

polar kev (2022). Mrtk2-unity developer documentation - mrtk 2. URL `https://learn.microsoft.com/en-us/windows/mixed-reality/mrtk-unity/mrtk2/?view=mrtkunity-2022-05`.

Pradeep, P. and Kant, K. (2022). Conflict detection and resolution in iot systems: A survey. *IoT*, 3(1), 191–218. doi:10.3390/iot3010012.

qianw211 (2022). QR code tracking overview - Mixed Reality. URL `https://learn.microsoft.com/en-us/windows/mixed-reality/develop/advanced-concepts/qr-code-tracking-overview`.

Robinson, W.N., Pawlowski, S.D., and Volkov, V. (2003). Requirements interaction management. *ACM Comput. Surv.*, 35(2), 132–190. doi:10.1145/857076.857079.

SteamDB (2024). Technologies. URL `https://steamdb.info/tech/`.

Suzuki, R., Karim, A., Xia, T., Hedayati, H., and Marquardt, N. (2022). Augmented reality and robotics: A survey and taxonomy for ar-enhanced human-robot interaction and robotic interfaces. In *Proceedings of the 2022 CHI Conference on Human Factors in Computing Systems*, CHI '22. Association for Computing Machinery, New York, NY, USA. doi:10.1145/3491102.3517719.

Unity Technologies (2023). Unity - manual: Ar development in unity. URL `https://docs.unity3d.com/2020.3/Documentation/Manual/AROverview.html`.

Unity Technologies (2024). Unity plans i& pricing. URL `https://unity.com/pricing`.

Vidal, E.J. and Villota, E.R. (2018). Sysml as a tool for requirements traceability in mechatronic design. In *Proceedings of the 2018 4th International Conference on Mechatronics and Robotics Engineering*, ICMRE 2018, 146–152. Association for Computing Machinery, New York, NY, USA. doi:10.1145/3191477.3191494. URL `https://doi.org/10.1145/3191477.3191494`.

Wiegers, K. and Beatty, J. (2013). *Software Requirements 3*. Microsoft Press.