

Article

Robo-HUD: Interaction Concept for Contactless Operation of Industrial Cobotic Systems

Dominykas Strazdas * , Jan Hintz  and Ayoub Al-Hamadi * 

Neuro-Information Technology, Otto-von-Guericke-University Magdeburg, 39106 Magdeburg, Germany; jan.hintz@ovgu.de

* Correspondence: dominykas.strazdas@ovgu.de (D.S.); ayoub.al-hamadi@ovgu.de (A.A.-H.)

Abstract: Intuitive and safe interfaces for robots are challenging issues in robotics. Robo-HUD is a gadget-less interaction concept for contactless operation of industrial systems. We use virtual collision detection based on time-of-flight sensor data, combined with augmented reality and audio feedback, allowing the operators to navigate a virtual menu by “hover and hold” gestures. When incorporated with virtual safety barriers, the collision detection also functions as a safety feature, slowing or stopping the robot if a barrier is breached. Additionally, a user focus recognition module monitors the awareness, enabling the interaction only when intended. Early case studies show that these features present good use-cases for inspection tasks and operation in difficult environments, where contactless operation is needed.

Keywords: augmented reality; activity recognition; cooperative systems; gesture recognition; human–robot interaction; intelligent robots; interactive systems; robot control; safety



Citation: Strazdas, D.; Hintz, J.; Al-Hamadi, A. Robo-HUD: Interaction Concept for Contactless Operation of Industrial Cobotic Systems. *Appl. Sci.* **2021**, *11*, 5366. <https://doi.org/10.3390/app11125366>

Academic Editors: Kouroush Jenab and Jorge A. Ortega-Moody

Received: 6 May 2021

Accepted: 7 June 2021

Published: 9 June 2021

Publisher’s Note: MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



Copyright: © 2021 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

The industry must adapt ever faster to constantly changing environmental conditions, such as technological leaps and product individualization. An important prerequisite for the design of the future production factories is therefore their adaptability. The current new generation of autonomous and collaborative robots further amplifies this factor [1–3]. Right now humans and robots coexist in the industrial environment mostly with physical barriers, to ensure safety. Removing these barriers creates new possibilities and challenges [4] in the field of human–robot interaction (HRI). The hazard posed by an industrial robot can be reduced to a certain degree by utilizing cobots, robots especially designed for human–robot collaboration, cooperation and coexistence, an important field of HRI [5].

Yet the challenge of an intuitive human–machine interaction system for industrial robots remains unresolved.

1.1. Related Works

In order to implement cobotic systems into industrial fields, the safety requirements must be met, most notably the safety standards ISO 10218-1/2 and the technical specification ISO/TS 15066. These identify four forms of collaboration:

1. *Safety-rated monitored stop*, stopping the robot in case the robot operations are halted when safety zones are violated;
2. *Hand guiding*, allowing the operator to teach new positions without the need of a teaching interface;
3. *Speed and separation monitoring*, changing the robots speed in relation to the position of the operator;
4. *Power and force limiting mode*, restricting the contact force in collaborative work.

As shown by Pasinetti et al. [6], time-of-flight (ToF) cameras can be utilized to reliably monitor the operator and, in combination with virtual barriers, slow down or stop the robot

if safety protocol is breached. Magrini et al. [7] have proposed a system that ensures the human safety in a robotic cell and enables gesture recognition for low-level robot control (e.g., start/stop).

There have been many different implementations of contactless HRI [8–16]. Tölgyessy et al. utilized a pointing gesture to point the robot to a certain location, where the direction of the pointing gesture intersects with the planar surfaces. The approach can be generalized by their proposed “Laws of Linear HRI” utilizing an intersection of a line, formed by any two joints of the detected human and a plane in the robots environment, creating a point of interest (POI) or a potential target for navigation. This pointing gesture method can also be used to distinguish between pre-programmed objects. The approach is specialized in finding POIs; however, it lacks user feedback, as it is unclear where the user was pointing until the robot executes the command [9].

Alvarez-Santos et al. [10] presented an augmented reality graphical user interface (AR-GUI) as the core element of their tour-guiding robot’s interaction. A user sees himself together with augmentation on the screen of the laptop, placed on top of the mobile robot and can push virtual buttons and perform gestures and patterns after performing an initialization for hand detection. Many other approaches utilize head mounted displays [11–14]; unfortunately these are not intuitive, require additional hardware for each user and are often difficult to integrate into the industrial environment. In addition to the time-of-flight sensors, there are approaches with capacitive [15], radar [17] or tomographic [16] sensors to detect gestures.

1.2. Previous Work

In a previous Wizard-of-Oz (WoZ) study, we explored the different forms of communication required for intuitive human–robot interaction (HRI). We implemented a WoZ framework “RoSA”, the “Robot System Assistant”, to overcome resource and current technology limitations, and carried out a study with 36 participants in which a “wizard” actively controlled the system. Participants were able to use speech, gaze, mimics and gestures without additional constraints to interact with a stationary cobot to solve some tasks related to cube stacking. Figure 1 shows the participant interacting with the system. According to strictly defined rules, the wizard, who observed the participant from another room, controlled the robot, following the participant’s instructions. Based on the results of the study, we intend to implement a real system that can use natural multimodal inputs to control robot assistants. Our findings suggest that speech and gesture recognition are indispensable for such a system, to allow intuitive HRI. [18]

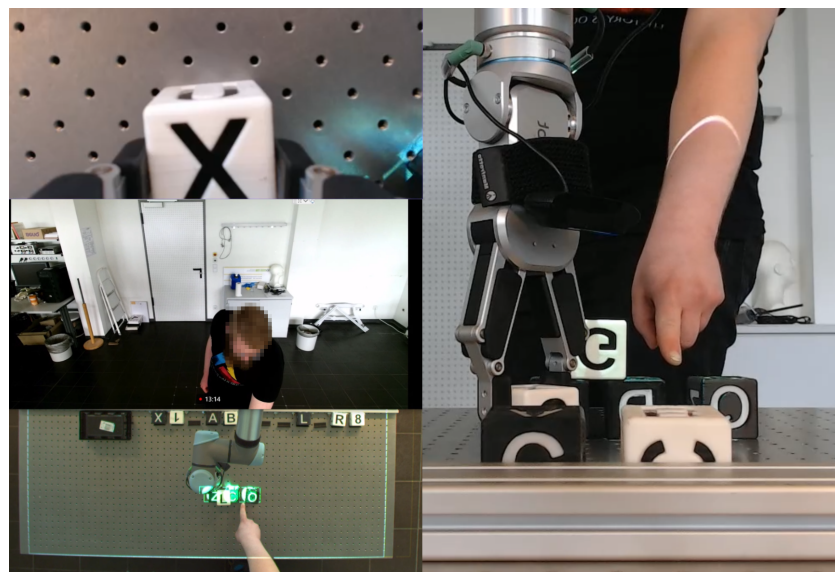


Figure 1. A participant using a pointing gesture for the robot to place the cube.

1.3. Our Approach

While the safety aspects of collaborative robot systems are already well defined in ISO/TS 15066, we focus our work on the intuitive operation and communication between humans and robots. Using augmented reality as visual user interface (UI) increases the simplicity and control in this field, as shown by Williams et al. [19]. The use of speech and gesture recognition also influences the efficiency of the communication. We propose a contactless UI approach, as avoiding the physical interaction between operator and robot can also increase the safety of the human operator. Our concept combines safety aspects and intuitive control using virtual barriers, contactless gestures and a Head-up-Display (HUD), a concept which is also referred to as *virtual mirror* [20]. The operator can move freely without having to wear virtual reality goggles or another gadget. Although not stated directly in their work, the method used by Alvarez-Santos et al. [10] would also classify as *virtual mirror*. The main difference would be the ability for multi-user input, no need for initialisation and use of the system for safety purposes. To further improve safety and user experience, we utilized gaze data [21] to ensure the operator is attentive [22,23].

2. System Setup

In the following, we will describe the hardware and software of the system setup used for our real-time HUD in robotic environments.

2.1. Hardware

The system is based on a UR5e cobot, by Universal Robots, equipped with a RG6 gripper by On-Robot, to fit safety standards for HRI [24]. A display, positioned behind the robot, acts as a *virtual mirror*, giving visual feedback while speakers give audio feedback and voice outputs. For contactless input, the RGB-D data from a Kinect V2 camera are used. The Kinect is positioned above the monitor.

The robot is placed on a sturdy metal table. The floor in front of the robot is divided into interaction zones, which are highlighted with tape. All devices are connected to the same PC (Specs: Intel Core i7-9800X @ 3.8GHz, NVIDIA 1080 TI, 32Gb RAM, SSD). The build can be seen in Figure 2.

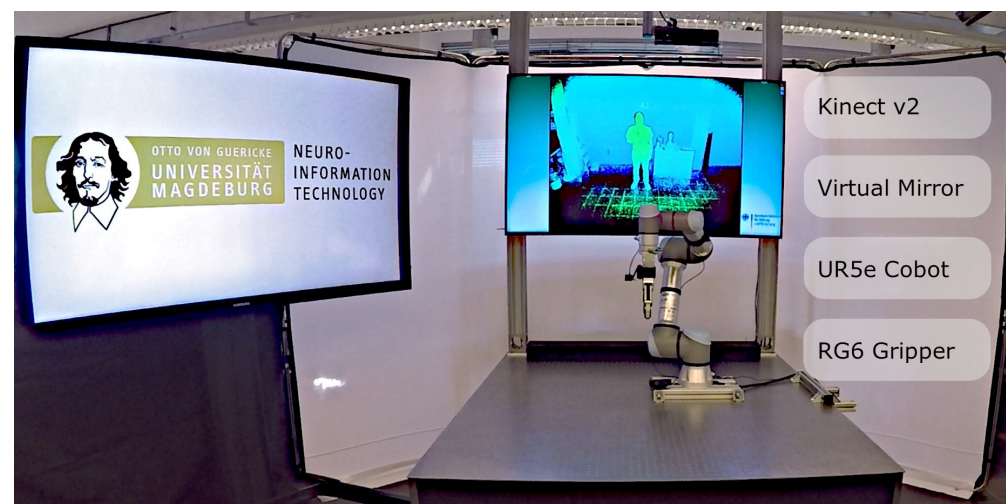


Figure 2. Experimental setup.

2.2. Software

The system is split into different modules for simplicity and re-usability purposes: sensor module, HUD module, safety module, attention module, speech module, robot module and collision module. Each module is prepared to work as a standalone, using a publish and subscribe service (e.g., *Robot Operating System (ROS)*: open source robotics middleware suite, or *Message Queuing Telemetry Transport (MQTT)*: a lightweight, publish-

subscribe network protocol) as a middleware. For the initial test, the modules are compiled together and share the data in a main program. The summary of the interaction and information flow can be seen in Figure 3.

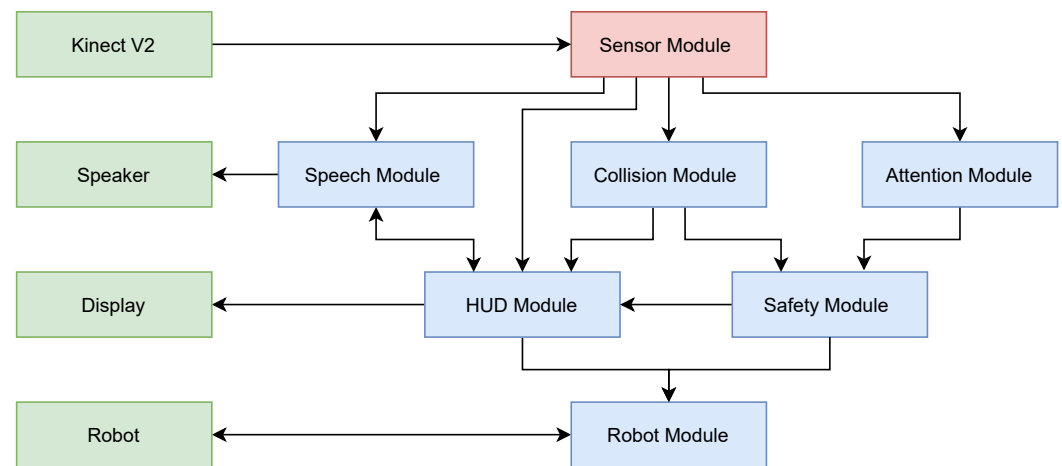


Figure 3. Final schematic representation for the interaction of the components (system concept).

3. Modules

In the following, we will explain each module and its functionalities as described in the previous chapter.

3.1. Sensor Module and Calibration

The sensor module initializes the camera and loads the extrinsic calibration between the camera and the robot. The calibration is done using Radon chessboard detection [25] and World/Robot-Tool-Flange calibration with 3D-EasyCalib™ [26]. For the body and gesture detection, the Kinect software development kit (SDK) is used. In this step, the depth data are converted to a body skeleton and an open/closed/pointing hand gesture estimation. Our preliminary experiment with 21 subjects for Kinect skeletal accuracy shows a mean deviation under 20 mm for hand detection under optimal conditions, when the hands of the subject are clearly visible and not occluded. These findings are supported by different studies and contribute to the reason why Kinect v2 was chosen for our setup [27,28].

To increase robustness against outliers, a mean over the 30 recently estimated hand positions is used as a filter, when time is uncritical but deliberate user input is needed. The sensor module subsequently outputs the skeleton data, audio input and RGB feed towards other modules for further processing.

3.2. Collision Module

The skeleton, provided by the sensor module, is processed by the collision module to allow the user to interact with virtual objects such as safety planes or augmented UIs. All object interaction is calculated in three-dimensional Euclidean space and is based on a $point(x, y, z)$, to which we refer as *SpacePoint*. For example, the skeleton as estimated by the Kinect SDK, consists of 25 *SpacePoints*.

ColBase, a base class for collision, contains the name of the virtual object and a boolean attribute of the momentary collision. *ColPoint* is an inheritance of *ColBase* and contains a *SpacePoint* as its center and a radius in which a collision can occur. The main collision detection uses the algorithm for a sphere–sphere collision detection:

$$|c_1 - c_2| < r_1 + r_2$$

which compares the distance between the two sphere centers (c_1, c_2), with the combined radii (r_1, r_2). An overlap or collision is possible only if the distance between the sphere

centers is smaller than the radius combined. Further objects with increased complexity can be created by grouping *ColPoints* together: *ColLine* consists of different *ColPoints*, *ColQuad* of different *ColLines*, etc. The full structure of collision objects can be seen in Figure 4.

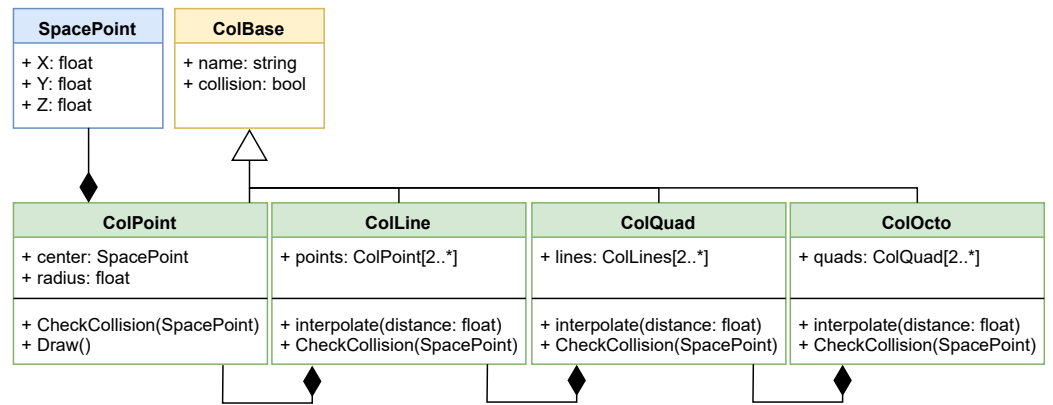


Figure 4. *ColPoint* Class diagram in UML. *ColPoint* contains a *SpacePoint* as a center. *ColBase* inherits its attributes (name and collision) to *ColPoint*, *ColLine*, *ColQuad* and *ColOcto*.

To fill in the spaces between two *ColPoints*, a linear interpolation can be used. This allows the creation of mesh-like objects that require only the outer points and a subdivision count for definition. The radius r , when considering the distance l between two *SpacePoints*, should be at least $\frac{1}{2}\sqrt{2}l$ in 2D and $\frac{1}{2}\sqrt{3}l$ in 3D to ensure a gap-free collision object.

A *ColQuad* can be defined by its outer corners and span a mesh of *ColPoints* to be impenetrable for collision detection. The advantage of this approach versus the use of a rectangle or a plane is that it can adapt to the real-world geometry, which is usually inexact. These imperfections are then leveled by the two-dimensional interpolation forming a curved mesh, which suits the real world better than a mathematically defined rectangle (Figure 5). A good use for a *ColQuad* would be as a virtual barrier for a safety fence or an interaction zone. In Figure 6, an interaction between a skeleton and a *ColQuad* can be seen.

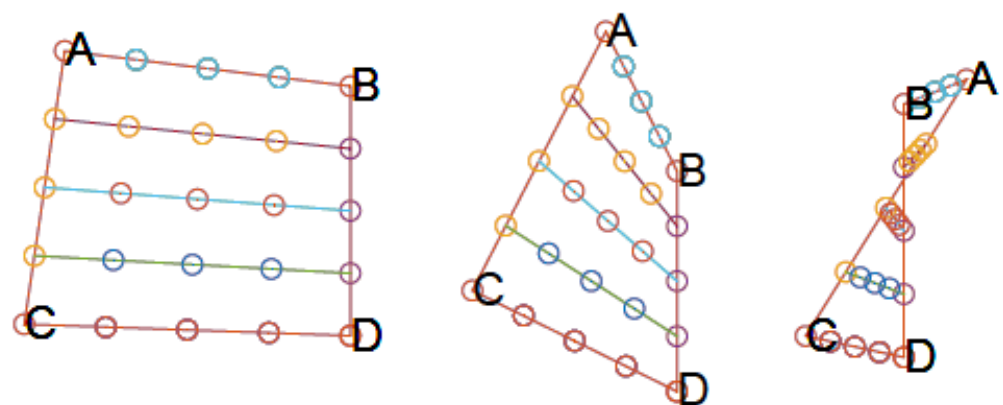


Figure 5. Representation of a *ColQuad* with a subdivision factor of five from three different perspectives. Note: in this example point A is not on the plane defined by BCD. The *ColQuad* is not a quadrilateral restricted to 2D but a 3D construct.

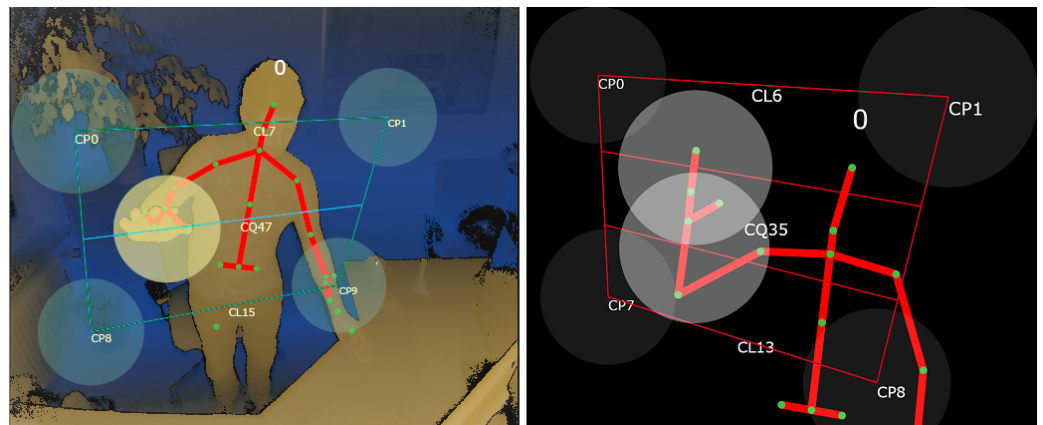


Figure 6. Examples of real-time visualization and interaction with *ColPoint* objects. Collisions between a *ColQuad* and body *SpacePoints* are shown.

3.3. HUD Module

The HUD module is connected to each module, as it controls the outputs and the robot’s actions and gives feedback to the user. Our real-time head up display for robotic environments utilizes the concept of a *virtual mirror*, as described by Billingham et al. [20], to augment a virtual menu as the main user interface. The user sees a mirrored live-feed of the Kinect depth data as well as the estimated skeletal and gaze data. When the preconditions are met (only one user in the interaction zone, gaze is focused on the robot/virtual mirror) a menu is augmented into the scene, fixed to the virtual skeleton, half a meter in of the subject. This way it follows the user moving in the interaction zone. Users can interact with the menu by extending their arms.

The menu consists of up to nine *ColPoints* arranged on a grid, represented as buttons. Only active menu items are displayed. With the intention to keep the system simple and consistent, the menu’s appearance was devised in a circular design. In order for the operator to activate a virtual button, the collisions of skeleton and corresponding *ColPoint* need to be calculated by the collision module. Any prior collision of skeleton and virtual menu results in a “click/tick sound” as feedback for a possible interaction. An input is only accepted when an expected gesture is held in place for a certain number of frames (15–30). This way accidental inputs are disregarded. The progress is also visualized by gradually changing the color of the chosen button and by playing a sound when the input is accepted. We refer to this action as “hover and hold”.

The process of selection can be seen in Figure 7: first an outline around the button is displayed, then the infill gradually becomes green.

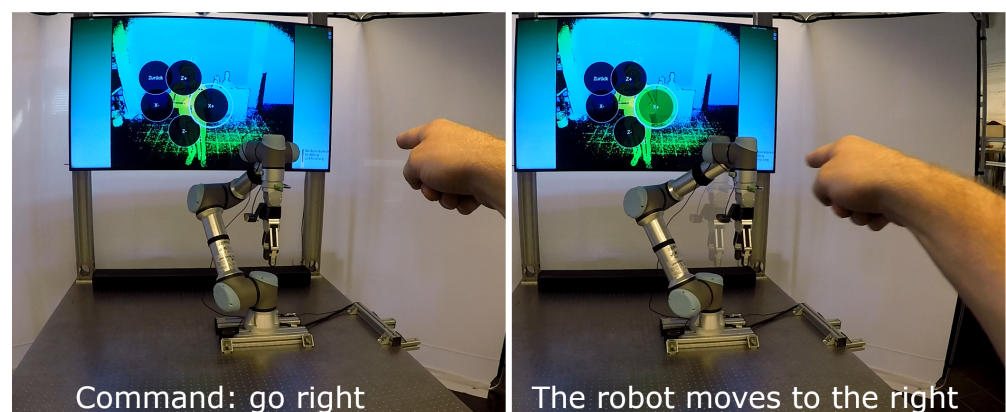


Figure 7. Augmented virtual menu for user interaction.

The menu allows to switch between the different robot modes: speed and separation monitoring, hand guiding/freedrive mode and direct robot control. Each choice needs to be confirmed (e.g., “Wish to continue?”, “yes/no”). If canceled, the user returns to the main menu and the robot returns to its home position.

3.4. Attention Module

We implemented a Kinect SDK independent head pose tracker that estimates the person’s gaze to monitor awareness and intention for interaction. The head pose estimation uses the HPFL method and is trained with the SyLaHP database both introduced by Werner et al. [21]. The head pose is predicted through Support Vector Regression [29]. If the resulting angle deviates from the given field of view for a few seconds, the input for the virtual menu is deactivated, as seen in Figure 8. This is done to ensure that the communication between the operator and the robot is only possible when a direct line of sight is present.

The implemented tracker also allows an identification and differentiation between users. Face attribute detection with deep metric learning [30,31] is used to generate user IDs. Users reentering the scene get recognized and assigned to the same ID that was used the last time. This would allow an access control, if paired with an user database. For our research interest, we only tracked whether the same person would be assigned the correct ID.

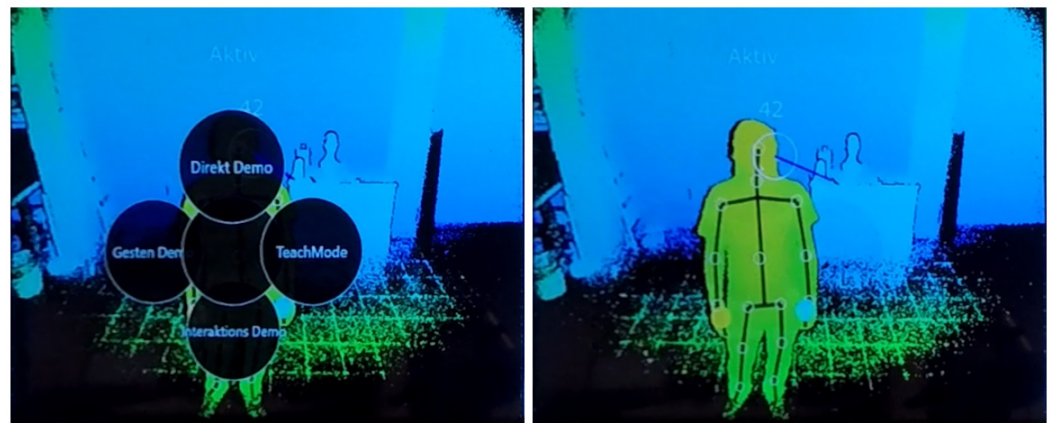


Figure 8. Attention detection: interaction is disabled when the user looks away.

3.5. Safety Module

To further ensure the human safety, the menu is only active when the operator is looking at the robot and stands within a predefined operating range. The operating range is defined by virtual barriers not shown on the virtual mirror, but highlighted on the ground. Violation of the virtual barriers are calculated in real time by the collision module Section 3.2. The robot is halted if the operator comes too close or leaves the work space abruptly. Additionally, speed reduction can be activated if the operator loses the robot out of sight (Section 3.4). The safety module operates parallel to the robot’s own safety measures described in Section 3.7.

3.6. Speech Module

For a more intuitive multi-modal user experience, speech synthesis was implemented. The system is able to readout prompts and questions and give audio feedback such as: “Ok”, “robot: ready”, etc. It utilizes the Microsoft Speech Platform, as it is easy to implement in the Windows environment and works offline. For now only speech synthesis is used.

3.7. Robot

The robot module controls the status, position and mode of the UR 5e robot. The robot's status is permanently monitored by the safety module allowing fast reaction and bi-directional control using Real-Time Data Exchange (RTDE). The robot is running basic movement programs, awaiting changes in its local registers, updated by the PC at 125 Hz that change depending on the user input:

- *Speed and separation monitoring*, where the robot is moving between predefined points and slows down if the user looks away;
- *Hand guiding/Freedrive mode*, allowing the operator to manipulate the robot manually;
- *Direct control*, allowing the user to move the robot arm directly via the menu.

By default the robot is running in *Speed and separation monitoring* mode. The robot module does not interfere with the built-in safety measures by UR. These include Safeguard Stop, stopping the robot if certain forces are exceeded, as well as the limits for joint position, pose and power. These features have been tested in accordance with EN ISO 13849-1:2015, Cat.3, PL [24]. Additionally integrated robot safety planes restrict the robot in its movements, to ensure there is no movement out of certain bounds that could damage the robot or the equipment.

4. Results and Discussion

In summary, our system combines safety aspects as proposed by Magrini et al. [7], a similar interaction as Alvarez-Santos et al. [10] with the addition of gaze detection by Werner et al. [21] and our implementation of a collision system based on *ColPoints*. We use Microsoft Speech Platform for the speech synthesis

Further we will discuss our results and overall findings.

4.1. Industrial Use

Additionally, in our laboratory environment we deployed our setup in industrial environment for user feedback. It was presented as an interactive demonstrator, to explore the features, without defined tasks. In Figure 9 two users can be seen detected by the system, while only one is marked as "active".

Summary of our observations:

- concern for fatigue for prolonged use;
- need for height adjustment, as taller persons had minor difficulties with the HUD;
- users trying to "click" the buttons instead of "hover and holding";
- users extending their arms too far, instead of holding, thus resulting in leaving the predefined menu *ColPoint* resulting in abortion of the wanted action;
- quick adaption to the system and the ability to control the robot within minutes;
- overall positive feedback for the user experience;
- good use-case for inspection tasks and operation in difficult environments where contactless operation is advantageous (high voltage, acid, sharp pieces).

4.2. Collision

Our goal for collision detection was real-time capability, which is 30 frames per second or above, as defined by the frame rate of the 3D input. The experiments showed that a worse-case scenario allowed about 100,000 verifications in single CPU thread mode. With multi-threading optimization, the load can be distributed evenly between the CPU threads allowing practically a million sphere–sphere collisions to be detected in real time. Considering six persons that can be detected by Kinect V2 with their respective 25 skeletal input *SpacePoints*, this sums up to 150 necessary collision verifications per *ColPoint*. This leads to total of 6000–7000 possible *ColPoints* that can be safely checked every frame. To further reduce the computational cost, a hull *SpacePoint* or a sphere-tree approach could be added. It would be also interesting to port the algorithm to a GPU.



Figure 9. Test in industrial environment.

4.3. Safety

Avoiding physical contact between the human and robot, through our gesture-based, contactless approach, increases the safety of the user. By adding *ColPoints* to the joint positions of the robot, simple human–robot collision prevention can be achieved. The robot, in this case, stops before the collision occurs because a virtual collision with a *ColPoint* happens first. Our introduced “Virtual Barriers” present an alternative for laser-based security scanners. This would be a novel approach since the first safety-rated (*performance level D*) 3D ToF camera (*Spotguard*® from *Tofmotions*) became available for the market.

By certifying the collision detection algorithm, our method could be used in the industrial environment. At this point, laser-based security scanners can be added as redundancy, to further improve safety and ensure a safety-rated monitored stop.

4.4. UI

Our “hover and hold” approach system differs from most conventional UIs (hover and click/touch), as there is no need for depth movement. Some users wanted to press a button/object expecting the system to react to pointing/clicking gestures varying in depth. The safety aspect of our multi-modal approach can therefore be perceived as less intuitive in comparison to Alvarez-Santos et al. [10], who included both options.

4.5. Future Works and Author Remarks

The current setup is limited to 2D gesture control. Dynamic gestures and micro gestures (moving just the fingertips) are not detected by the system. As derived from our previous *RoSA* research, the system still lacks a pointing gesture similar to the one implemented by Tölgyssey et al. [9], as well as voice control.

The described system is a part of an upcoming field study with a significant group of subjects. The goal is to allow a multi-modal natural interaction with robotic assistance systems utilizing speech, gaze, mimics and gestures using current technologies. Since the goal and the necessary system are of higher complexity, we decided to split the research into smaller parts. To replicate the *RoSA* System from the *WoZ* Study, but without the “wizard”, the speech input, a dialog system, augmented projection, pointing gestures and ROS middleware are yet to be implemented. The evaluation, which was currently not possible due to the COVID-19 pandemic, will then happen in a large field study for the overall system while also containing tasks for each module separately.

Author Contributions: Conceptualization, D.S.; methodology, D.S. and J.H.; software, D.S.; validation, D.S.; investigation, D.S. and J.H.; resources, A.A.-H.; writing—original draft preparation, D.S. and J.H.; writing—review and editing, D.S., J.H. and A.A.-H.; visualization, D.S. and J.H.; supervision, A.A.-H.; project administration, D.S., A.A.-H.; funding acquisition, A.A.-H. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by the Federal Ministry of Education and Research of Germany (BMBF) RoboAssist No. 03ZZ0448L, HuBa No. 03ZZ0470 and Robo-Lab No. 03ZZ04X02B within the Zwanzig20 Alliance 3Dsensation

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Informed consent was obtained from all subjects involved in the study.

Data Availability Statement: Not applicable.

Conflicts of Interest: The authors declare no conflict of interest. The funders had no role in the design of the study; in the collection, analyses, or interpretation of data; in the writing of the manuscript, or in the decision to publish the results.

References

1. Flacco, F.; De Luca, A. Safe physical human-robot collaboration. In Proceedings of the 2013 IEEE/RSJ International Conference on Intelligent Robots and Systems, Tokyo, Japan, 3–7 November 2013; IEEE: Piscataway, NJ, USA, 2013; p. 2072.
2. Fast-Berglund, Å.; Palmkvist, F.; Nyqvist, P.; Ekered, S.; Åkerman, M. Evaluating cobots for final assembly. *Procedia CIRP* **2016**, *44*, 175–180. [\[CrossRef\]](#)
3. Chen, Q.; Heydari, B.; Moghaddam, M. Levering Task Modularity in Reinforcement Learning for Adaptable Industry 4.0 Automation. *J. Mech. Des.* **2021**, *143*, 1–35. [\[CrossRef\]](#)
4. Saenz, J.; Elkmann, N.; Gibaru, O.; Neto, P. *Survey of Methods for Design of Collaborative Robotics Applications—Why Safety Is a Barrier to More Widespread Robotics Uptake*; ACM International Conference Proceeding Series; Association for Computing Machinery: New York, NY, USA, 2018; Volume Part F137690, pp. 95–101. [\[CrossRef\]](#)
5. Peshkin, M.; Colgate, J.E. Cobots. *Industrial Robot. Int. J.* **1999**, *26*, 335–341. [\[CrossRef\]](#)
6. Pasinetti, S.; Nuzzi, C.; Lancini, M.; Sansoni, G.; Docchio, F.; Fornaser, A. Development and Characterization of a Safety System for Robotic Cells Based on Multiple Time of Flight (TOF) Cameras and Point Cloud Analysis. In Proceedings of the 2018 Workshop on Metrology for Industry 4.0 and IoT, Brescia, Italy, 16–18 April 2018; pp. 1–6. [\[CrossRef\]](#)
7. Magrini, E.; Ferraguti, F.; Ronga, A.J.; Pini, F.; De Luca, A.; Leali, F. Human-robot coexistence and interaction in open industrial cells. *Robot. Comput.-Integr. Manuf.* **2020**, *61*, 1–55. [\[CrossRef\]](#)
8. Lentini, G.; Falco, P.; Grioli, G.; Catalano, M.G.; Bicchi, A. *Contactless Lead-Through Robot Interface*; Technical Report; I-RIM: Pisa, Italy, 2020. Available online: https://i-rim.it/wp-content/uploads/2020/12/I-RIM_2020_paper_114.pdf (accessed on 1 April 2021).
9. Tölgyessy, M.; Dekan, M.; Duchoň, F.; Rodina, J.; Hubinský, P.; Chovanec, L. Foundations of Visual Linear Human–Robot Interaction via Pointing Gesture Navigation. *Int. J. Soc. Robot.* **2017**, *9*, 509–523. [\[CrossRef\]](#)
10. Alvarez-Santos, V.; Iglesias, R.; Pardo, X.M.; Regueiro, C.V.; Canedo-Rodriguez, A. Gesture-based interaction with voice feedback for a tour-guide robot. *J. Vis. Commun. Image Represent.* **2014**, *25*, 499–509. [\[CrossRef\]](#)
11. Fang, H.C.; Ong, S.K.; Nee, A.Y. A novel augmented reality-based interface for robot path planning. *Int. J. Interact. Des. Manuf.* **2014**, *8*, 33–42. [\[CrossRef\]](#)
12. Ong, S.K.; Yew, A.W.; Thanigaivel, N.K.; Nee, A.Y. Augmented reality-assisted robot programming system for industrial applications. *Robot. Comput. Integr. Manuf.* **2020**, *61*, 101820. [\[CrossRef\]](#)
13. Gadre, S.Y.; Rosen, E.; Chien, G.; Phillips, E.; Tellex, S.; Konidaris, G. End-user robot programming using mixed reality. In Proceedings of the 2019 International Conference on Robotics and Automation (ICRA), Montreal, QC, Canada, 20–24 May 2019; pp. 2707–2713. [\[CrossRef\]](#)
14. Kousi, N.; Stoubos, C.; Gkournelos, C.; Michalos, G.; Makris, S. Enabling human robot interaction in flexible robotic assembly lines: An augmented reality based software suite. *Procedia CIRP* **2019**, *81*, 1429–1434. [\[CrossRef\]](#)
15. Stetco, C.; Muhlbacher-Karrer, S.; Lucchi, M.; Weyrer, M.; Faller, L.M.; Zangl, H. Gesture-based contactless control of mobile manipulators using capacitive sensing. In Proceedings of the I2MTC 2020—International Instrumentation and Measurement Technology Conference, Dubrovnik, Croatia, 25–28 May 2020; pp. 21–26. [\[CrossRef\]](#)
16. Muhlbacher-Karrer, S.; Brandstötter, M.; Schett, D.; Zangl, H. Contactless Control of a Kinematically Redundant Serial Manipulator Using Tomographic Sensors. *IEEE Robot. Autom. Lett.* **2017**, *2*, 562–569. [\[CrossRef\]](#)
17. Wang, Y.; Ren, A.; Zhou, M.; Wang, W.; Yang, X. A Novel Detection and Recognition Method for Continuous Hand Gesture Using FMCW Radar. *IEEE Access* **2020**, *8*, 167264–167275. [\[CrossRef\]](#)
18. Strazdas, D.; Hintz, J.; Felßberg, A.M.; Al-Hamadi, A. Robots and Wizards: An Investigation Into Natural Human–Robot Interaction. *IEEE Access* **2020**, *8*, 207635–207642. [\[CrossRef\]](#)

19. Williams, T.; Hirshfield, L.; Tran, N.; Grant, T.; Woodward, N. Using augmented reality to better study human-robot interaction. In *Lecture Notes in Computer Science (Including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*; Springer: Cham, Switzerland, 2020; Volume 12190, pp. 643–654. [[CrossRef](#)]
20. Billinghurst, M.; Clark, A.; Lee, G. A survey of augmented reality. *Found. Trends Hum. Comput. Interact.* **2014**, *8*, 73–272. [[CrossRef](#)]
21. Werner, P.; Saxen, F.; Al-Hamadi, A. Landmark based head pose estimation benchmark and method. In Proceedings of the 2017 IEEE International Conference on Image Processing (ICIP), Beijing, China, 17–20 September 2017; IEEE: Piscataway, NJ, USA, 2017; pp. 3909–3913.
22. Bee, N.; André, E.; Tober, S. Breaking the Ice in Human-Agent Communication: Eye-Gaze Based Initiation of Contact with an Embodied Conversational Agent. In *Intelligent Virtual Agents*; Ruttkay, Z., Kipp, M., Nijholt, A., Vilhjálmsson, H.H., Eds.; Springer: Berlin/Heidelberg, Germany, 2009; pp. 229–242.
23. Fischer, K.; Jensen, L.C.; Suvei, S.D.; Bodenhagen, L. Between legibility and contact: The role of gaze in robot approach. In Proceedings of the 25th IEEE International Symposium on Robot and Human Interactive Communication, RO-MAN 2016, New York, NY, USA, 26–31 August 2016; pp. 646–651. [[CrossRef](#)]
24. Nord, T. *Certificate*; Test Report No. 3520 1327; TÜV NORD CERT GmbH: Essen, Germany, 2018.
25. Duda, A.; Frese, U. Accurate Detection and Localization of Checkerboard Corners for Calibration. In Proceedings of the BMVC, Birmingham, UK, 3–6 September 2018; p. 126.
26. Vehar, D.; Nestler, R.; Franke, K.H. 3D-EasyCalib™-Toolkit zur geometrischen Kalibrierung von Kameras und Robotern Journal 22. In *Anwendungsbezogener Workshop zur Erfassung, Modellierung, Verarbeitung und Auswertung von 3D-Daten, 3D-NordOst*; GfAI Gesellschaft zur Förderung angewandter Informatik: Berlin, Germany, 2019; pp. 15–26.
27. Otte, K.; Kayser, B.; Mansow-Model, S.; Verrel, J.; Paul, F.; Brandt, A.U.; Schmitz-Hübsch, T. Accuracy and Reliability of the Kinect Version 2 for Clinical Measurement of Motor Function. *PLoS ONE* **2016**, *11*, e0166532. [[CrossRef](#)] [[PubMed](#)]
28. Abbondanza, P.; Giancola, S.; Sala, R.; Tarabini, M. *Accuracy of the Microsoft Kinect System in the Identification of the Body Posture*; Springer: Cham, Germany, 2017; Volume 192, pp. 289–296. [[CrossRef](#)]
29. Chang, C.C.; Lin, C.J. LIBSVM: A library for support vector machines. *ACM Trans. Intell. Syst. Technol. TIST* **2011**, *2*, 1–27. [[CrossRef](#)]
30. King, D.E. dlib C++ Library: High Quality Face Recognition with Deep Metric Learning Knuth: Computers and Typesetting. Available online: <http://blog.dlib.net/2017/02/high-quality-face-recognition-with-deep.html> (accessed on 1 April 2021).
31. King, D.E. Dlib-ml: A machine learning toolkit. *J. Mach. Learn. Res.* **2009**, *10*, 1755–1758.

Short Biography of Authors



Dominykas Strazdas was born in Vilnius, Lithuania, in 1989. He received his B.Sc. and M.Sc. degrees in mechatronics from Otto-von-Guericke University Magdeburg and is currently pursuing his Ph.D. degree in electrical engineering at the Otto-von-Guericke University Magdeburg. His research interests include human-machine interaction, especially natural, intuitive, and contact-less communication between humans and robots. Since 2017 he has been a Research Assistant with the Neuro-Information Technology research group at the Otto-von-Guericke University Magdeburg.



Jan Hintz was born in Brunswick, Lower-Saxony, Germany, in 1996. He received his B.Sc. degree in electrical engineering and information technology from the Otto-von-Guericke University Magdeburg. He is currently pursuing his M.Sc. degree in electrical engineering and information technology at the Otto-von-Guericke University Magdeburg. His research interests include computer vision, image processing, machine learning, and human-machine interaction. Since 2018 he has been a Research Assistant with the Neuro-Information Technology research group at the Otto-von-Guericke University Magdeburg.



Ayoub Al-Hamadi received his Masters Degree in Electrical Engineering & Information Technology in 1997 and his PhD. in Technical Computer Science at the Otto-von-Guericke-University of Magdeburg, Germany in 2001. Since 2003 he has been Junior-Research-Group-Leader at the Institute for Electronics, Signal Processing and Communications at the Otto-von-Guericke-University Magdeburg. In 2008 he became Professor of Neuro-Information Technology at the Otto-von-Guericke University Magdeburg. In May 2010 Prof. Al-Hamadi received the Habilitation in Artificial Intelligence and the Venia Legendi in the scientific field of "Pattern Recognition and Image Processing" from Otto-von-Guericke-University Magdeburg, Germany. Prof. Al-Hamadi is the author of more than 360 articles in peer-reviewed international journals, conferences and books. His research interests include human-robot interaction, computer vision, pattern recognition, and AI. See www.nit.ovgu.de for more details.