CIRP Annals - Manufacturing Technology 69 (2020) 425–428

Contents lists available at ScienceDirect

CIRP Annals - Manufacturing Technology

journal homepage: https://www.editorialmanager.com/CIRP/default.aspx

Closed-loop augmented reality towards accurate human-robot collaboration

Xi Vincent Wang*, Lihui Wang (1), Mingtian Lei, Yongqing Zhao

Department of Production Engineering, KTH Royal Institute of Technology, Sweden

A R T I C L E I N F O

Article history:
Available online 17 May 2020

Keywords:
Manufacturing system
Augmented reality
Human-robot collaboration

A B S T R A C T

Augmented Reality provides an interactive approach to combining the physical manufacturing environment with the computer-generated virtuality. However, it is still challenging to establish an efficient AR system due to the complexity of the actual manufacturing environment, and difficulties in human-robot collaboration. In this research, a feasible AR system is developed with a novel closed-loop structure. A feasible AR-based human-robot interaction method is also developed based on the HoloLens device, together with an advanced compensation mechanism towards accurate robot control. The proposed work is implemented and validated through case studies, and quantifiable performance assessment on system response time, compensation accuracy, etc.

1. Introduction

In recent years, the manufacturing industry calls for a new generation of paradigms, systems and technologies, and new initiatives have been launched worldwide. Despite different terminologies in different countries, the mutual spirit of the ongoing efforts is to strengthen the manufacturing with the help of new automation, information and communication technologies. Among the novel technologies, Augmented Reality (AR) is considered a promising approach as it provides fast and efficient interaction methods for the operator by combining the physical manufacturing environment with the virtuality in the cyber space [1]. It is also aligned with the objectives of Cyber-Physical Systems. In parallel, the Human-Robot Collaboration (HRC) approach also aims to offer a friendly and safe working environment that the operator and industrial robot can work in the same space at the same time [2,3]. HRC provides efficient support to both manufacturing and disassembly operations [4]. Hence, it is a logical thinking to integrate the AR approach with the HRC application to achieve a more flexible and interactive manufacturing solution. A thorough system design is needed to integrate multiple technology components. Meanwhile, the manufacturing tasks requires high precision during execution, e.g. welding, assembly, machining, etc. So it is essential to develop an AR-based HRC approach with high accuracy. Therefore, this research is conducted to answer the research questions of 1) how to establish a comprehensive AR-based HRC system integrating essential technology modules efficiently, and 2) how to deploy a compensation mechanism in the AR-based HRC system. The research contributions in this paper includes a novel AR system based on wearable devices, and a closed-loop mechanism which is capable of compensating potential execution deviations towards accurate HRC processes and operations. During the system implementation and evaluation, the performance of the proposed system is validated in terms of response speed, and the result of the compensation mechanism is evaluated via the accuracy measurement.

* Corresponding author.
E-mail address: wangxi@kth.se (X.V. Wang).

https://doi.org/10.1016/j.cirp.2020.03.014

2. Related work

In 2008, Reinhart et al. [1] introduced a programming system based on Mixed Reality (MR). Interactive drag-and-drop functions were realised for robot programming [4]. Similarly, a Mix Reality (MR) approach was proposed for collaborative design [5,6]. In this approach, the virtual prototype can be manipulated and its process sequence can be decided. Then the research was extended to an AR-based interaction system from design to virtual assembly training [7]. Ong et al. [8] combined the AR technology with a finite element analysis for easy interpretation of the data and the location of the critical regions on the product.

During the planning phase, Akan et al. [9] introduced an AR approach based on a multimodal language. The planning focus is shifted from traditional coordinate-based to intuitively object-based. Jiang et al. [10] presented a facility layout planning system based on augmented reality. An Analytical Hierarchy Process–Genetic Algorithm (AHP-GA) based optimisation algorithm was integrated with the AR approach, towards automatic planning suggestions.

At the process execution stage, operator-support methods were proposed to provide virtual instructions during assembly [11,12]. The AR approach is utilised for the direct communication for the control both mobile and industrial robots [13,14]. The production and process information is provided to the user to enhance the operator's integration with the assembly process.

After the task execution, Erkoyuncu et al. [15] introduced an AR system supporting context-aware maintenance. The proposed system is able to provide overlaid data animation, AR content, and context awareness in the virtuality.

Kardos et al. [16] proposed a touch-based solution developed to control industrial robots. Similarly, a touchscreen-based approach was introduced to manipulate the mobile robot remotely [17]. The approach provides intuitive operation via the third-person view through the touchscreen.

Lambrecht and Kruger [18] developed a hand-held device and a motion tracking system. Both static and dynamic gestures can be captured to define poses, trajectories, and tasks. Hofmann et al. [19] also proposed a Go&See system on a smartphone-enabled AR environment to identify the bottlenecks in production lines. Eventually, Maly et al. [20] compared the AR environment on the touchscreen-based smartphone, and in-air gesture-based glasses. These two approaches are evaluated from four main

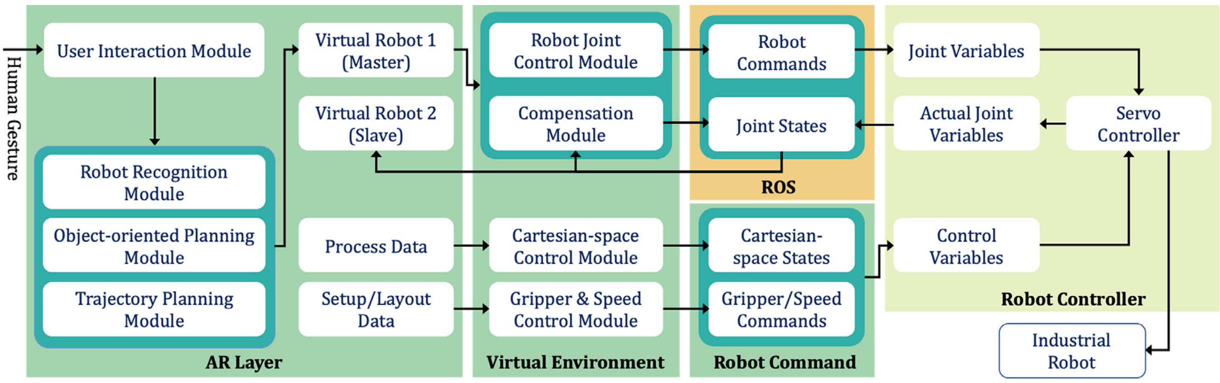

Fig. 1. System structure.

perspectives, i.e. marking limitation, view limitation, visualisation techniques, and interaction techniques.

To recap, despite the significant development of the AR-based robotic approach in recent years, there is still a lack of a comprehensive structure design for the AR-based industrial robot system. Most of the previous work is based on open-loop systems and there is no compensation mechanism proposed in the literature. Thus in this research, a novel AR-based robot system is proposed with detailed and feasible system structure design. A closed-loop compensation mechanism is also developed towards accurate HRC operations.

3. Proposed system

As shown in Fig. 1, the system consists of four major layers, i.e. AR Layer, Virtual Environment, Robot Operating System (ROS) with robot commands, and the Robot Controller layer.

3.1. AR Layer

The AR Layer is the main interaction cockpit for the user. All the user commands and inputs are communicated from here. The interaction logic of the AR Layer is summarised in Fig. 2. After the AR system is launched, the Robot Recognition Module firstly identifies the geometry and location of the physical robot, and deploys the Virtual Robot 1 (master model) right over the physical robot in the AR vision. Then User Interaction Module recognises and analyses the operator's gesture. According to the user's hand gesture inputs, e.g. tap, gaze, grip, etc., the user interactions are interpreted to the moving commands to drive the Virtual Robot 1 (master model), whose posture will be moved, or rotated accordingly. The master model represents the final location and posture of the industrial robot. Conventionally, it is difficult to identify the detailed and accurate robot coordinates and posture in the AR space. Hence in this research, the Object-oriented Planning Module is also developed. The user can manipulate a generalised virtual object to precisely define the desired final position and angle of the end-effector. The user can also input exact parameters of the coordinates when the physical objects' position and orientation is available before the collaborative process.

The Trajectory Planning Module is also developed at the AR device layer. The operator is able to set multiple via points in the virtual space in the AR, to address the feasible best trajectory of the robot, e.g. avoid potential collisions, pass through narrow areas, etc. Then the robot processes through the via points step by step toward the final destination, according to the user's definition. The planned path and via point can be edited as long as the user needs to. During the interactive review phase, the determined trajectory is dynamically visualised by the Virtual Robot 1 to provide an intuitive result to the user. After the trajectory plan is confirmed by the user, the result is sent to the virtual environment layer and the interaction module is terminated.

In the proposed system, two virtual robot models are actually established in the AR. During the trajectory execution, the Virtual Robot 1 (master model) presents the final posture of the robot, and Virtual Robot 2 (slave model) reflects the dynamic posture of the robot in real-time, according to the current joint states feedbacked from the robot controller. The joint position, speed, torque data can also be visualised in the AR space, thus forming the first closed-loop in the system. If there are any potential issues identified by the user, he or she can stop the robot and modify the plan and improve the trajectory plan accordingly.

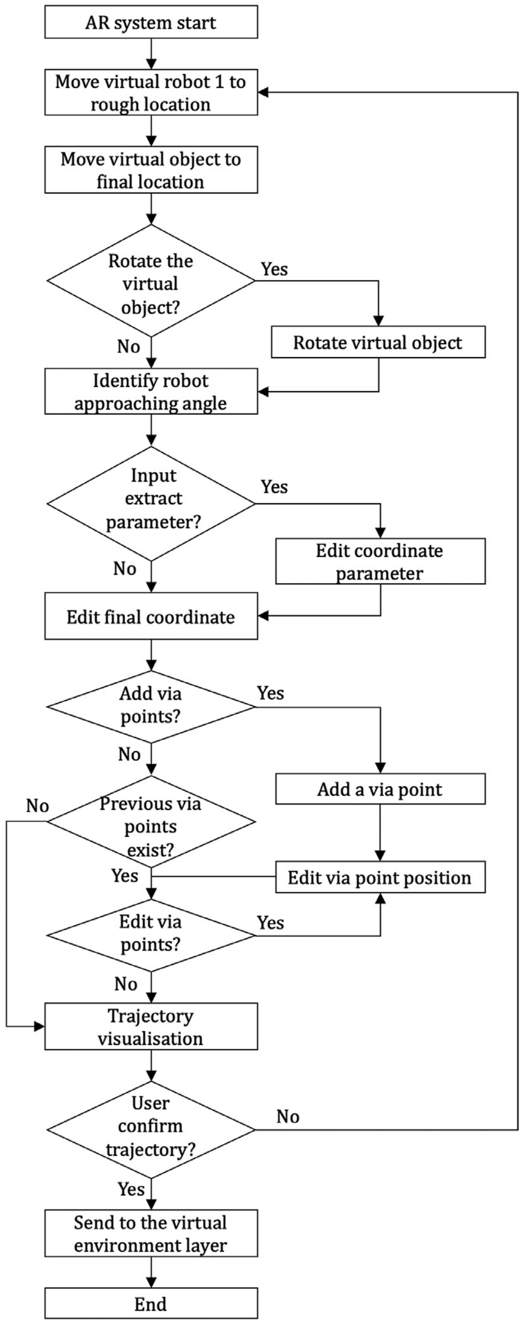

Fig. 2. Interaction logic of the AR Layer.

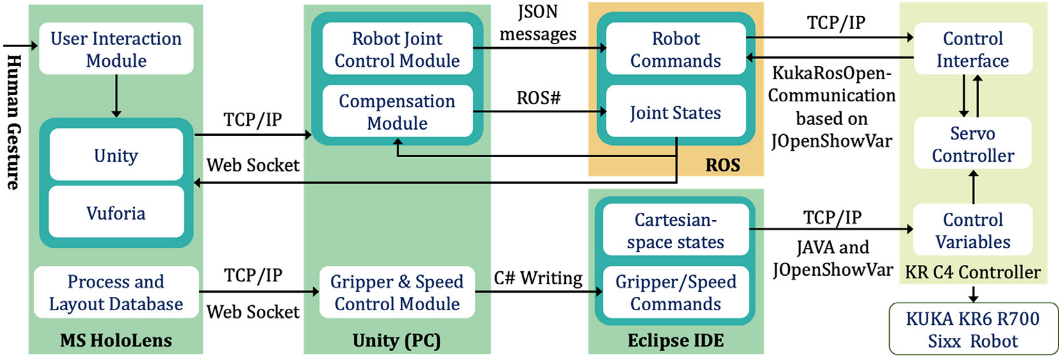

Fig. 3. Deployment structure.

3.2. Virtual Environment

The Virtual Environment layer includes the Robot Joint Control Module, which reads the trajectory plan, as soon as it is confirmed by the end user at the AR Layer (Fig. 1). The robot plan is converted from trajectory plan into robot commands for industrial robot. At this stage, there are two types of controls of the robot and two control modules are developed in the proposed system, respectively.

The Robot Joint Control Module manipulates the robot based on the combination of every joints, and the Cartesian-space Control Module utilises the final position of the end-effector in the Cartesian space. In practice, the manufacturing data, e.g. process plan, robotic cell layout, fixture geometry, etc., is static knowledge that is maintained before the operation planning. This type of static data is normally stored in the Cartesian space. Thus, the system developed in this research provides two mechanisms to integrate the dynamic planning and static control knowledge via both joint combination and Cartesian space. It also offers more potentials for interfacing with different AR devices and robot systems in the future. The Denavit–Hartenberg method is utilised to establish the coordinate transformation,

$$ {}^{i-1}_{\ i}T = Trans(Z, d_i) * Rot(Z, \theta_i) * Trans(X, a_i) * Rot(X, \alpha_i) \quad (1) $$

where T is the transformation locating the end-link, d is the link offset, θ is the joint angle, a is the link length, and α is the link twist. Eventually, the robot command is transferred to the next layer, together with the results of the Gripper & Speed Control Module.

3.3. ROS, Robot Command and Controller layers

In this research, the Robot Operating System (ROS) is utilised to process the robot commands, and proprietary robot data file is responsible of maintaining the Cartesian space states respectively. The robot commands are eventually sent to the control variable interfaces within the industrial robot controller.

As mentioned above, the manufacturing scenario has high requirement for robot operations. From the control's perspective, there are classic approaches like Proportional-Derivative (PD) and Proportional–Integral–Derivative (PID) controllers. However, the dynamic approach might affect the robot's stability and linearity. Hence in this research, a post-compensation mechanism is developed instead without altering the original control cycle. It needs to be noted that the actual position of the robot is sent back to ROS only at the last phase of the robot operation execution. The Compensation Module automatically compares the expected position variables against the actual robot positions. When the difference is larger than the pre-defined threshold, the compensation algorithm will be initiated accordingly and generate a compensatory movement command for the robot controller towards the planned position. Then the robot advances again until the full completion of the task. Thus, the control accuracy of the system is guaranteed.

In practice, it is feasible to deploy the compensation mechanism at either the ROS level or virtual level. In the proposed system, it is deployed at Unity PC layer to decrease the computation task on the robot side and to provide the potential of more complex compensation mechanism in the future, while ROS is utilised as the computer-robot interface only. As a result, the actual robot position is also dynamically looped back to the AR device, which drives the virtual model representing the actual robot posture constantly. Thus it achieves a *closed-loop* control mechanism for the AR-based human-robot collaboration with high accuracy throughout the system.

4. Implementation and system performance evaluation

The proposed system is implemented in the real industrial robot cell to evaluate the feasibility and performance. The deployment structure of the proposed system is shown in Fig. 3, which is compliant with the system structure design introduced in the previous section.

4.1. Deployment structure and implementation

The Microsoft HoloLens is utilised as the AR device to host the interaction module. The AR environment and user interface are developed based on Unity real-time development platform. The Vuforia engine is chosen for the model positioning and calibration. The user interaction module and trajectory planning module are developed based on the above-mentioned interaction logic. As shown in Fig. 4, the user is able to view both the virtual and real robots through the HoloLens view field, and interact with the functionality panel in the corner. After the connection from AR device to the robot is established, the user can manipulate the virtual robot via gesture commands to determine the posture, trajectory and task. Complex robot trajectories can also be planned via multiple via points defined by the user.

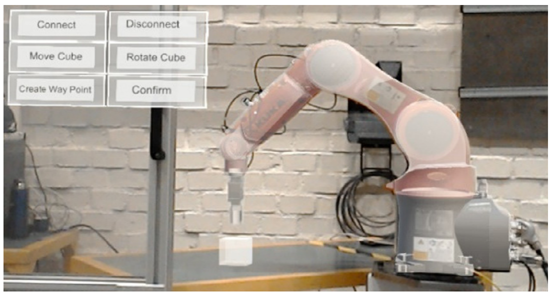

Fig. 4. AR-based industrial robot system (the virtual robot in white colour, and the real KUKA robot in orange).

After the trajectory plan is confirmed by the user, it is then transferred to the virtual environment hosted in MS Windows PC via web server-socket protocol. The Unity terminal on the PC interprets the results and generates the robot commands that are comprehensible for ROS. The Unity interface then establishes the communication with ROS via ROS# interface and writes in JSON messages. In parallel, the robot commands in Cartesian space are documented in the self-contained command data file via C# writing directly.

Then ROS interprets the commands to be streamed to the KUKA KR C4 controller directly through KukaRosCommunication interface based on JOpenShowVar method. The robot commands are sent to the controller via TCP/IP protocol and manipulate the KUKA KR6 R700 Sixx industrial robot. Eventually the robot starts to execute the operation accordingly, and the Virtual Robot 2 duplicates the physical robot with more details streamed back to the user interaction module on HoloLens, thus forming a closed-loop AR system.

4.2. Performance evaluation

After the proposed system is developed, the overall performance is evaluated via quantifiable assessments on the response speed. More than

60 tests have been taken and the response time results are presented in Table 1. The launch process connecting the AR environment takes 3,118 milliseconds on average. The initiation process takes more time as there are multiple modules needed to be triggered and connected. After the communication is successfully established, the actual communication interval between the virtual environment and ROS is 85.8 milliseconds on average. The average control latency from ROS to the KUKA robot is 69.6 milliseconds through the TCP/IP communication. Eventually, throughout the whole interactive control loop the total time interval is 155.4 millisecond on average. In practice, AR-based planning and control scenarios do not require real-time control and fast system response. Thus, it is a sufficient response speed especially considering the intermedium runtime requirement.

Table 1
System response time.

Response Time (ms)	Connection Initiation	Unity to ROS	ROS to Robot	Overall System Latency
Max	3824	156	87	243
Min	2278	62	50	112
Average	**3118.7**	**85.8**	**69.6**	**155.4**

Regarding the compensation mechanism, multiple random deviations are deployed on each joint from A2 (joint closer to the robot's base, according to KUKA's definition) to A5 (joint closer to the end-effector) and the result is presented in Table 2. The compensation mechanism automatically detects the difference between the desired joint angle and the actual one. If the difference is larger than the threshold, the compensation module generates the compensational move command and drives robot accordingly. From joints A2 to A4, the average deviation input is approximately 2 to 3 degrees, and the compensated position error decreases to 0.012 degrees, which is close to the maximum accuracy of the KUKA KR6 R700 Sixx robot. Overall more than 99.4% of the deviations are compensated. As the joint A5 is a smaller joint close to the end-effector, more deviations (average 4 degrees) are applied to the joint to evaluate a more challenging scenario. Eventually the system is capable of compensating the movement and decreases the deviation to 0.054 degrees on average, in which 98.65% of the position error is eliminated. Hence, the feasibility of the proposed closed-loop compensation mechanism is validated with high automation level and accuracy.

Table 2
Compensation results.

Average Deviation (degree)	Before Compensation	After Compensation	Compensate Rate
A2	2.341	0.012	99.49%
A3	2.679	0.009	99.66%
A4	2.009	0.012	99.40%
A5	4.006	0.054	98.65%

5. Discussions and conclusions

The modern manufacturing industry calls for new systems and technologies to achieve higher efficiency, user-friendliness and safety. The AR technology provides an opportunity for a more interactive and friendly manufacturing environment, while HRC solution combines the flexibility of the human operator and the accuracy of the industrial robot. Hence in this research, the AR and HRC approaches are integrated. The novel scientific contributions of this work include the comprehensive system structure for AR-based HRC, and the closed-loop compensation mechanism towards higher accuracy, answering the research question in system integration and compensation mechanism respectively. The proposed approach is validated and evaluated via quantifiable performance assessments focusing on both the system response time and accuracy performances.

In the future, the proposed AR-based HRC system can be further evaluated with more complex manufacturing environments and operations. The safety and ergonomic measures can be specifically taken to assess the operators' experience and acceptance of the proposed approach. The results can be much valuable for the future AR and HRC research in the manufacturing environment, especially the ones after the operator wears the AR devices for a longer time period for daily manufacturing operations. Moreover the compensation module can be further improved by more advanced algorithms, e.g. learning-by-failure mechanism.

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

References

[1] Reinhart G, Munzert U, Vogl W (2008) A programming system for robot-based remote-laser-welding with conventional optics. *CIRP Ann - Manuf Technol* 57(1):37–40.
[2] Wang XV, Kemény Z, Váncza J, Wang L (2017) Human-robot collaborative assembly in cyber-physical production: Classification framework and implementation. *CIRP Ann - Manuf Technol* 66(4):5–8.
[3] Wang L, Gao R, Váncza J, Krüger J, Wang XV, Makris S, Chryssolouris G (2019) Symbiotic human-robot collaborative assembly. *CIRP Ann - Manuf Technol* 68(2):701–726.
[4] Wegener K, Chen WH, Dietrich F, Dröder K, Kara S (2015) Robot assisted disassembly for the recycling of electric vehicle batteries. *Procedia CIRP* 29:716–721.
[5] Ong Sk, Pang Y, Nee AYC (2007) Augmented reality aided assembly design and planning. *CIRP Ann - Manuf Technol* 56(1):49–52.
[6] Ong SK, Shen Y (2009) A mixed reality environment for collaborative product design and development. *CIRP Ann - Manuf Technol* 58(1):139–142.
[7] Ong SK, Wang ZB (2011) Augmented assembly technologies based on 3D bare-hand interaction. *CIRP Ann - Manuf Technol* 60(1):1–4.
[8] Nee AYC, Ong S-K (2013) Virtual and augmented reality applications in manufacturing. *IFAC Proc Vol* 46(9):15–26.
[9] Akan B, Ameri A, Cürüklü B, Asplund L (2011) Intuitive industrial robot programming through incremental multimodal language and augmented reality. In: *2011 IEEE Int. Conf. Robot. Autom. IEEE*, 3934–3939.
[10] Jiang S, Nee AYC (2013) A novel facility layout planning and optimization methodology. *CIRP Ann - Manuf Technol* 62(1):483–486.
[11] Makris S, Pintzos G, Rentzos L, Chryssolouris G (2013) Assembly support using AR technology based on automatic sequence generation. *CIRP Ann - Manuf Technol* 62(1):9–12.
[12] Liu H, Wang L (2017) An AR-based Worker Support System for Human-Robot Collaboration. *Procedia Manuf.* : 1–9.
[13] Makris S, Karagiannis P, Koukas S, Matthaiakis A-S (2016) Augmented reality system for operator support in human–robot collaborative assembly. *CIRP Ann - Manuf Technol* 65(1):61–64.
[14] Kousi N, Stoubos C, Gkournelos C, Michalos G, Makris S (2019) Enabling Human Robot Interaction in flexible robotic assembly lines: an Augmented Reality based software suite. *Procedia CIRP* 81:1429–1434.
[15] Erkoyuncu JA, del Amo IF, Dalle Mura M, Roy R, Dini G (2017) Improving efficiency of industrial maintenance with context aware adaptive authoring in augmented reality. *CIRP Ann - Manuf Technol* 66(1):465–468.
[16] Kardos C, Kemény Z, Kovács A, Pataki B, Váncza J (2018) Context-dependent multimodal communication in human-robot collaboration. *PROCEDIA CIRP* 72:15–20.
[17] Hashimoto S, Ishida A, Inami M, Igarashi T (2011) Touchme: An augmented reality based remote robot manipulation. In: *21st Int. Conf. Artif. Real. Telexistence, Proc. ICAT2011. Citeseer*, pp 61–66.
[18] Lambrecht J, Krüger J (2012) Spatial programming for industrial robots based on gestures and augmented reality. In: *Intell. Robot. Syst. (IROS), 2012 IEEE/RSJ Int. Conf. IEEE*, pp 466–472.
[19] Hofmann C, Staehr T, Cohen S, Stricker N, Haefner B, Lanza G (2019) Augmented Go & See: An approach for improved bottleneck identification in production lines. *Procedia Manuf* 31:148–154.
[20] Malý I, Sedláček D, Leitão P (2016) Augmented reality experiments with industrial robot in industry 4.0 environment. In: *2016 IEEE 14th Int. Conf. Ind. Informatics. IEEE*, pp 176–181.