

Article

Programming of Industrial Robots Using Virtual Reality and Digital Twins

Andrzej Burghardt ¹, Dariusz Szybicki ¹, Piotr Gierlak ^{1,*} , Krzysztof Kurc ¹, Paulina Pietruś ¹ and Rafał Cygan ²

¹ Department of Applied Mechanics and Robotics, Faculty of Mechanical Engineering and Aeronautics, Rzeszow University of Technology, 35-959 Rzeszów, Poland; andrzejb@prz.edu.pl (A.B.); dszybicki@prz.edu.pl (D.S.); kkurc@prz.edu.pl (K.K.); p.pietrus@prz.edu.pl (P.P.)

² Consolidated Precision Products Poland sp. z o.o., 35-078 Rzeszów, Poland; rafal.cygan@cppcorp.com

* Correspondence: pgierlak@prz.edu.pl; Tel.: +48-17-865-1854

Received: 25 November 2019; Accepted: 8 January 2020; Published: 9 January 2020



Featured Application: The results of the work may find application in robot programming.

Abstract: The article presents a method of programming robots using virtual reality and digital twins. The virtual environment is a digital twin of a robotic station, built based on CAD models of existing station elements. The virtual reality system is used to record human movements in a virtual environment, which are then reproduced by a real robot. The method developed is dedicated mainly to such situations in which it is necessary for the robot to reproduce the movements of a human performing a process that is complicated from the point of view of robotization. An example of using the method for programming a robot implementing the process of cleaning ceramic casting moulds is presented.

Keywords: digital twin; virtual reality; robot programming

1. Introduction

1.1. Digital Twins

The digital twin [1] is an idea related to the concept of a virtual, digital equivalent of a physical product. The digital twin is used to reproduce a real, physically existing environment, process or single object. Its role is to map the main features of the physical object or process enabling simulation, prediction and optimization in the areas of system servicing [2,3], product design and manufacturing systems [4–6] and processes [7]. The main features of the digital twin, such as its form, upgradeability, the degree of precision with which it maps reality, and many others, depend on the purpose it serves [8]. Figure 1 shows an object and its digital twin, whose role is to map geometric features. In addition to areas such as: Big data analytics, cloud computing, the Internet of Things, Artificial intelligence, automation and integration of systems, additive manufacturing and augmented reality, digital twins are one of the elements of the Industry 4.0 concept [9,10].

Robotics, and mainly robotic manufacturing systems [11], is a specific area of the digital applications of twins. Numerous works relate to the use of digital twins in robotic assembly tasks [12–15], in issues of cooperation with people [16], or in issues related to predictive maintenance [17]. Robotics also falls under the Industry 4.0 area due to the flexibility of solutions, the ability to adapt and reconfigure systems and their interaction with the environment through extensive sensory systems.



Figure 1. Casting mould (lower part) combined with the casting bowl (upper part): (a) The actual mould; (b) the digital twin mould in the form of a CAD (Computer Aided Design) model. The role of the digital twin is to reproduce the geometrical features and mass of the object. The real object and the digital twin show some differences, e.g., in the proportions of the lower and upper parts. They result from the fact that the lower part of the casting mould is not very repeatable, because it consists of seven layers manually applied with the use of binders. The digital twin takes into account the largest possible dimensions of the object.

1.2. Virtual Reality

Virtual reality is one of the components of the Industry 4.0 idea. Virtual reality is a representation of reality created using information technology. It involves multimedia creation of a computer vision of space, objects, and events. The virtual image can reproduce elements of both the real and fictional worlds. In virtual reality it is possible to simulate the user's presence and his impact on the virtual environment. Feedback from the environment is sent to one or more senses in such a way that the user has a sense of immersion in the simulation [18].

The virtual reality system presents objects to the user with the help of image, sound and sensory stimuli, and allows for interaction, giving the impression of being in a simulation. Sensor-based feedback provides the user with direct sensory information depending on their location in the environment. Sensory information is sent via computer generated stimuli such as visual, sound, or tactile information. Most often, most feedback is done through visual information. In addition to the possibility of simulating the real environment, the virtual reality system requires the use of an interface that allows the user to “enter” the virtual reality.

Examples of the use of virtual reality are flight simulators [19] and surgical simulators [20], which have been known for years. They were among the first applications of virtual reality due to the particularly responsible tasks of pilots and surgeons.

1.3. Applications of Digital Twins and Virtual Reality in Robotics

In recent years, the use of virtual reality and digital twins covers increasingly wider areas of engineering. In robotics, for example, they are robotic station simulators used to train operators [21–23] or for design of robotic stations [24]. Another area of application of modern techniques is the issue of human-robot collaboration. The article [25] describes the use of virtual reality and digital twins in the analysis of the quality of human-robot cooperation, which allowed for safe implementation of the cooperation strategy. In [26] virtual reality was used to train the desired behavior of people in the presence of robots. It contributes to increasing the safety and mental comfort of employees in real conditions of cooperation with robots. The next area of application of virtual reality is systems optimization. An example is work [27], which presents a solution to the problem of optimizing robot movement in order to increase production efficiency. In recent years, virtual reality techniques have been increasingly used in robot programming [23,28]. This contributes to the simplification

and shortening of this activity, which affects the speed of implementing new solutions and reduces production costs, especially in areas with high production variability.

1.4. Robot Programming Methods

Robot programming methods are a field that has been rapidly developing in recent years. The purpose of the newly developed methods is mainly to speed up the programming process and facilitate its implementation by unqualified personnel. The reduction of programming time, which is a preparatory step for robotic production and implementation of programming by low skills personnel, is a source of savings, and therefore of great importance for businesses.

A detailed description of some methods of programming robots is described in [29,30]. These are the following methods: The lead-through programming method, walk-through programming method and its extension called programming by demonstrating. The work [31] describes the offline programming method that consists in generating robot paths using a virtual environment. CAD models of robots and auxiliary devices are imported into the environment. Robot paths are generated to track the indicated contours of CAD models or connecting the indicated points of the environment. Another method is programming using augmented reality. Augmented reality is a technology derived from virtual reality and consists of computer generated 3D objects that are immersed in a real world scene. This enables programming of robots without having to model all of the elements in a virtual environment [32]. For example, virtual elements can be robot arms whose CAD models are easily available. Robot models can be superimposed on the view of the real environment in which the objects with which the robots interact are located [30]. In this way, robots can be programmed to track the appropriate contours of the object [33]. Examples of applications implemented with the use of augmented reality programming are: Robotic welding process [34], painting [32], packing [35], and polymer manufacturing process [36].

This article presents a method developed for programming robots using virtual reality and digital twins. The method developed is dedicated mainly to such situations in which it is necessary for the robot to reproduce the movements of a human performing a process that is complicated from the point of view of robotization. Section 2 describes the method of programming robots developed. Section 3 presents an example of using the method for programming a robot implementing the process of cleaning ceramic casting moulds. The work ends with a summary

2. Programming Robots Using Virtual Reality and Digital Twins

This section proposes a developed robot programming method based on the operator's interaction with elements of the virtual environment. It is a method developed to replace a human being with a robot when performing an operation, during which an experienced employee performs a sequence of movements that are difficult to copy without direct measurements of the movements. Such situations occur when the sequence of movements leading to the achievement of a specific goal was selected by the employee based on many years of experience. Examples include painting, manually grinding turbine blades, cleaning casting moulds, moving bulky components, and complicated assembly of components. This approach is particularly advantageous when it is necessary to move objects that are in fact large in mass and the operator can easily manoeuvre them in a virtual environment.

The virtual environment is a digital twin of a robotic station, built based on CAD models of existing station elements. With the help of a virtual visualization system, the operator is placed in a virtual robotic station. His role is to perform such manoeuvres with a selected object or tool as is the case during the implementation of real processes.

The digital twin of the robot was created on the basis of the CAD model offered by the robot manufacturer. This model accurately determines the mass of links and mass distribution, which is important for modeling dynamic properties. In this model, the geometrical features of the robot are sufficiently represented, because the robot does not come into contact with a rigid environment. Other

robot features such as deformability under load or temperature changes in the described application do not have a significant impact on the robot's movement.

The virtual reality system was built based on the Oculus Rift (Facebook, Inc., Menlo Park, CA, USA) solution and an efficient PC. The system (Figure 2) consists of goggles, two controllers for hand control (Oculus Touch), two vision position sensors, and a computer with appropriate software. The goggles are equipped with two displays—one for each of the user's eyes. In the case of VR goggles, specialized lenses placed in each of the screens, whose task is to maximize image curvature, are the key. This is necessary to create the impression that the virtual reality is as realistic as possible. For a good reception of virtual reality, a smooth image at a level of at least 90 frames per second is also needed. After the user puts the goggles on their head, they display in front of the user's eyes a computer-generated image of a world or a film recorded in 360 degrees. In the environment the user can look around in a natural way, i.e., by moving their head or whole body. This is possible thanks to the acceleration sensor and gyroscope placed in the goggles and diodes generating infrared radiation, whose position is tracked by external sensors that are part of the system.



Figure 2. The idea of functioning of the virtual reality system.

Each of the diodes located on the goggles and hand controllers sends a unique pulse signature. Thanks to this, vision sensors record information on the position of the diodes and on this basis the position and orientation of the goggles and hand controllers are determined. The Oculus Touch is a small size device that surrounds the user's hands, which are constantly tracked by an external vision sensor system. The devices detect all movements of the wrists and simultaneously track the fingers: Thumbs, index and middle, which significantly improves immersion and allows the user, for example, to grasp objects in a three-dimensional environment. The option of grasping objects was used to program the robot's movements during the cleaning of casting moulds.

The work [37] presents data on the accuracy of the Oculus Rift system. These data are summarized in Table 1.

Table 1. Oculus Rift system parameters.

Translational Accuracy Based on Goggles	Translational Accuracy Based on Hand Controllers	Rotational Accuracy Based on Goggles	Rotational Accuracy Based on Hand Controllers
1.66 ± 0.74 mm	4.36 ± 2.91 mm	0.34 ± 0.38 deg	1.13 ± 1.23 deg

Such system parameters are sufficient for the analyzed task.

3. Example of Using the Programming Method

The application of the robot programming method is presented based on the example of robotization of the cleaning process for casting moulds for the production of steering apparatus in the aviation industry.

3.1. Description of the Problem

The axial turbine steering apparatuses of an aircraft engine are precision castings of nickel superalloys, made as one invisible part with complex shapes, consisting of a set of steering blades properly arranged on the internal and/or external bearing flanges (Figure 3). Castings of steering apparatuses are among the most difficult parts that are made using casting methods due to the extremely high requirements in terms of the complexity of the structure, accuracy of execution, and operational requirements. They are operated under the conditions of complex stress and vibration fields at a temperature exceeding 1000 °C, in a highly aggressive environment of corrosive flue gases. The use of integral steering apparatuses is a competitive solution in terms of quality compared to the collars composed of blade segments and simultaneously perfecting the design of the aircraft engine to increase its reliability and durability. At the same time, it results in a significant complication of the casting process.



Figure 3. Integral steering apparatus.

The quality of axial turbine steering apparatuses is determined, among others, by the quality of performance of multilayer ceramic casting moulds and the acceptable content of undesirable impurities in these moulds. Therefore, the mould cleaning process to remove all contaminants is an extremely important part of the casting production process.

Until now, cleaning of moulds was performed by an employee and consisted in appropriate shaking and rotation of a casting mould with washing liquid. Rinsing time is extremely important because the moulds become soaked and damaged when the cleaning process is too long.

3.2. Solution to the Problem

The programming of a robot to perform the mould cleaning process is part of a larger task also including the construction of the station, the schematic diagram of which is shown in Figure 4.

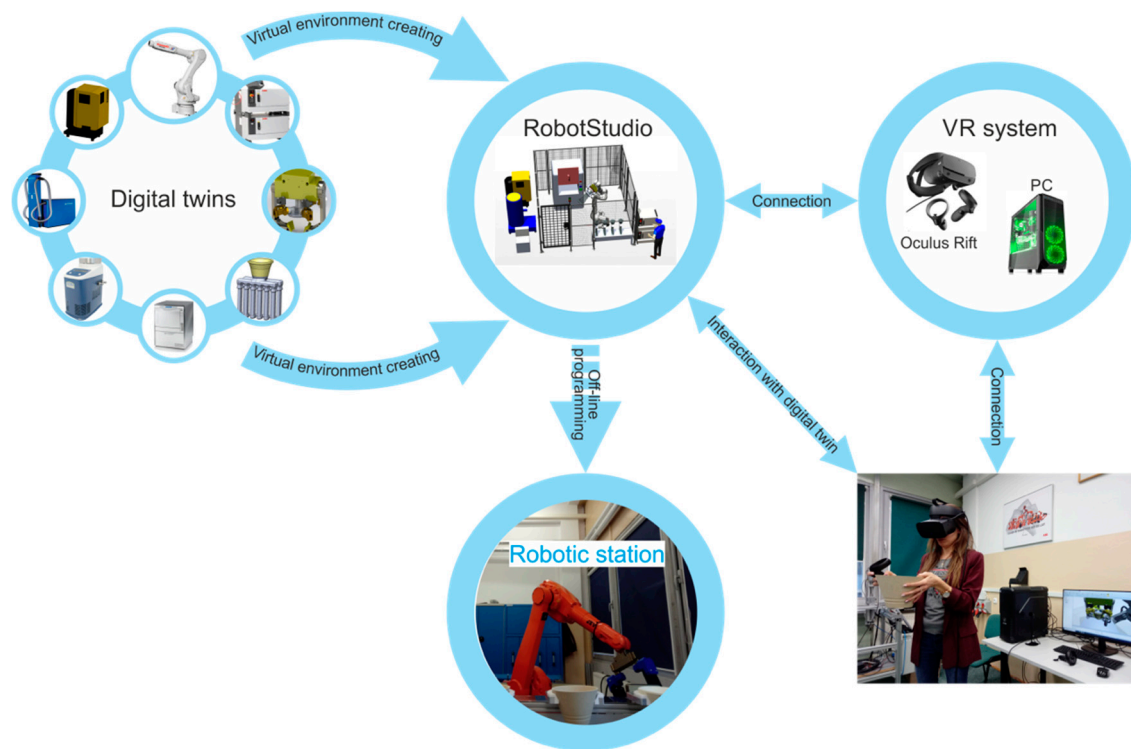


Figure 4. Schematic diagram of the building and programming of a robotic station using a VR (Virtual Reality) system.

The individual stages of this task are as follows:

1. Design and implementation of a robotic station with its digital twin. The station consists of an industrial, six-axis robot with a load capacity of 80 kg, compressed warm air supply system, washing liquid supply system, water and dirt suction system, drying chamber, loading and receiving station and auxiliary devices such as fencing and a safety system (Figure 5).
2. Making a digital twin casting bowl and a ceramic casting mould combined with it (Figure 1b) based on existing real objects (Figure 1a).
3. Design and implementation of a device for grasping the casting bowl together with a digital twin. The precision instrument for the robot to grasp the casting bowl is equipped with connections for supplying washing liquid and drying air and connection for extracting washing liquid, air, and dirt (Figure 6).
4. Configuration of the virtual environment consisting in placing digital twins of all of the elements in RobotStudio (Version 6.08, ABB Asea Brown Boveri Ltd., Zurich, Switzerland, 2019).
5. Connection of the VR system with RobotStudio.
6. Placing a human in the virtual environment via the VR system interface.
7. Data acquisition during virtual cleaning of the mould, which consists in the human performing typical movements of actual mould coupled with virtual mould with simultaneous tracking of movements by a sensory system integrated with VR.
8. Programming a robotic station in offline mode to perform the process of cleaning casting moulds based on the collected data.
9. Tests of the mould cleaning process.

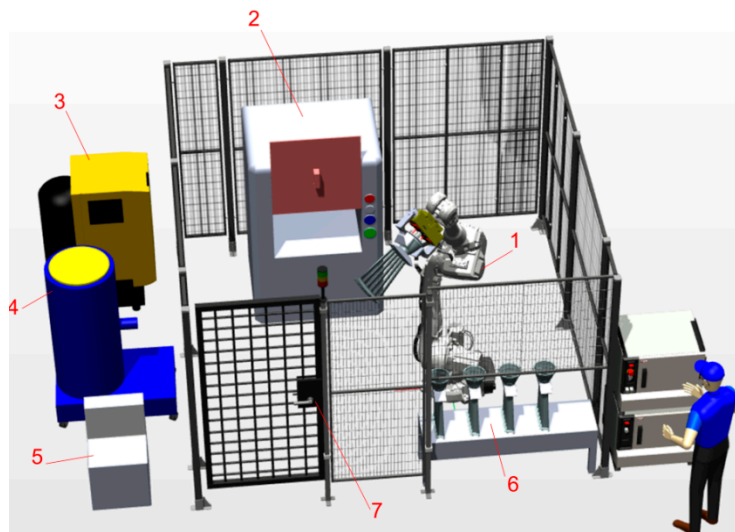


Figure 5. Digital twin of a robotic station: 1—Robot, 2—Drying chamber, 3—Compressor, 4—Suction device, 5—Heating ultrathermostat, 6—Loading and receiving station, 7—Safety elements.

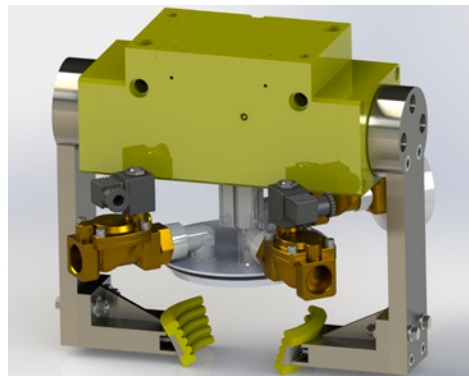


Figure 6. Device for grasping the casting bowl.

Figure 5 shows a view of a digital twin of a robotic station.

The flow of information regarding the position and orientation of the bowl with the mould is shown in Figure 7.

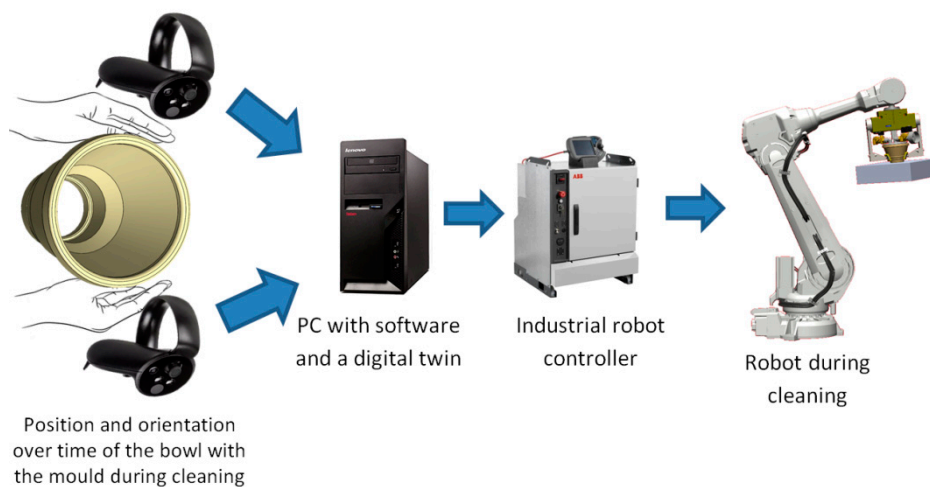


Figure 7. Flow of information regarding the position and orientation of the bowl with the mould.

The subsequent robot targets, consisting of position and orientation and collected during virtual cleaning are included in a program generated by RobotStudio software. The program code for the programmed task is attached in Supplementary Materials (Text file S3: Program Code). The robot targets are presented in Figure 8 in the form of square markers. They are not evenly distributed in time, their average sampling time is 0.1333 s. The robot's position and orientation coordinates (Figure 8) have been generated in virtual environment based on robot targets. These coordinates are marked with solid lines in Figure 8. Additionally, the joint coordinates corresponding to the coordinates of the position and orientation of the robot end-effector are shown in Figure 9. Coordinates presented in Figures 8 and 9 are calculated with taking into account dynamical properties of robot and they correspond to real performance in the assumed operating conditions.

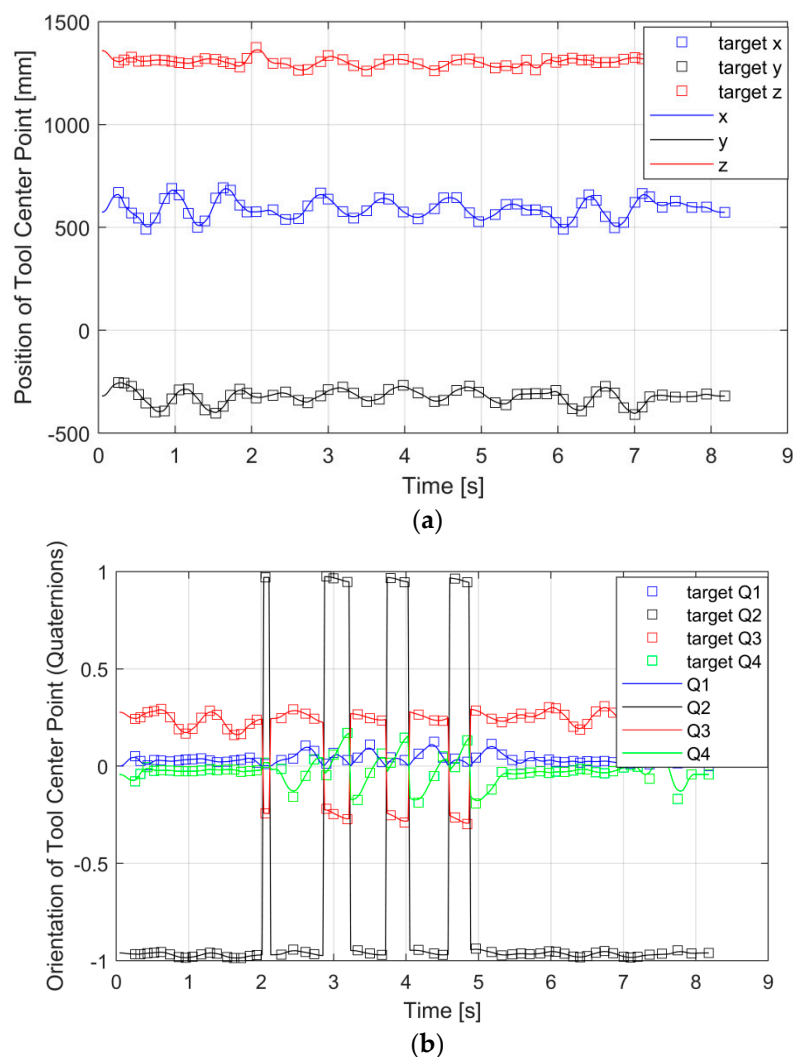


Figure 8. Coordinates of tool center point (TCP) of robots: (a) Position; (b) orientation expressed in quaternions.

During programming of the robotic station, three restrictions were noted: Singular configurations, movement restrictions in robot joints, and mould collisions. To avoid the first two situations, RobotStudio software was used, which at the programming stage returns information to the programmer in the event of reaching singular configurations or joint restrictions, just like it is in other programming methods. In order to avoid collision of the mould with other elements of the cell, in a virtual environment seen by the programmer, a rectangular block with a contour larger than the dimensions

of the mould was used. Thanks to this, it is guaranteed that if the rectangular contour does not collide with other elements of the cell, then the mould collision will not occur.

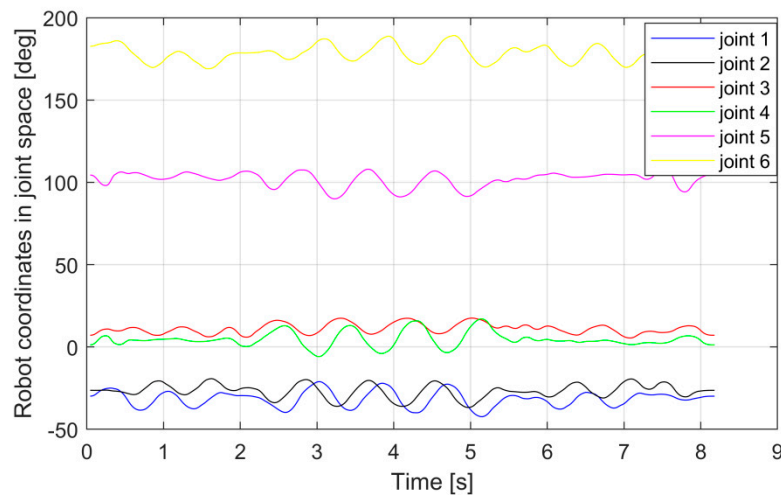


Figure 9. Robot coordinates in joint space.

During programming the station, there were no messages about singular configurations or about restrictions on the movement of robot links. There were also no collisions. The subsequent positions and orientations of the robot were saved, so the programming task was successful. Movies showing the programming process and the programming effect are also included in Supplementary Materials (Video S1: Robot Programming, Video S2: Robot Path).

4. Conclusions

The article presents a method of programming industrial robots. Its novelty lies in the fact that data generated during the virtual experiment is used to generate a given robot path. The experiment consists in the implementation of the task by a human interacting with a digital twin of one of the physical elements of the station. During this time, information about the position and orientation of the manipulated object is obtained. This is a different situation to the previously known solutions, in which a human guides a virtual robot along the desired path of movement. Instead, the human acts on the selected object, his movements are recorded, and the task of the robot is to recreate the human movements.

Table 2 compares programming methods due to their time-consuming, as well as advantages and disadvantages. The most difficult to determine is time needed for programming, because it depends largely on the experience of the programmer in a given method. The “programming time consuming” column has the most subjective nature. The authors used their knowledge resulting from numerous applications in the aviation industry. However, time consumption is to some extent related to the described advantages and disadvantages.

Further work will include the implementation of a mould cleaning system at the foundry plant. The next element will be a series of trainings for foundry plant personnel who will be able to program the robotic station themselves when new elements are introduced to production.

Table 2. Comparison of programming methods.

Programming Method	The Main Steps Necessary to Robot Programming	Advantages of Programming Methods ¹	Disadvantages of Programming Methods ¹	Programming Time Consuming
Lead-through programming	- Configuring the real robot station		- Inability to quickly program the reference motion path - Time-consuming and nonintuitive manual selection of all points on the robot path	Very high
Walk-through programming	- Configuring the real robot station - Equipping the robot with a programming system with force measuring system	- Possibility of quick programming of reference path.	- High cost of the system for measuring the robot-programmer interaction force	Very low
Offline programming	- Configuring the real robot station - Creating a station model in a virtual environment	- Ability to quickly check the effects of programming in a virtual environment	- Inability to quickly program the reference motion path - Time-consuming and nonintuitive manual selection of all points on the robot path	Very high
Programming using virtual reality and digital twins	- Configuring the real robot station - Configuring the VR system - Creating a station model in a virtual environment	- Possibility of quick programming of reference path - Ability to quickly check the effects of programming in a virtual environment - Relatively low cost of the VR system		Very low

¹ In relation to the described process of mould cleaning.

Supplementary Materials: The following are available online at <http://www.mdpi.com/2076-3417/10/2/486/s1>. Video S1: Robot Programming; Video S2: Robot Path; Text file S3: Program Code.

Author Contributions: A.B. contributed to the methodology and supervision; D.S. contributed to the investigation and software; P.G. contributed to the conceptualization and writing of this paper; K.K. contributed to the software and validation; P.P. contributed to the software and visualization; R.C. contributed to the founding acquisition and project administration. All authors have read and agreed to the published version of the manuscript.

Funding: This project is financed by the Minister of Science and Higher Education of the Republic of Poland within the “Regional Initiative of Excellence” program for years 2019–2022. Project number 027/RID/2018/19, amount granted 11,999,900 PLN.

Acknowledgments: The authors would like to thank Consolidated Precision Products Poland for providing products used for research and other facilities.

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. Glaessgen, E.; Stargel, D. The Digital Twin Paradigm for Future NASA and U.S. Air Force Vehicles. In Proceedings of the 53rd AIAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics and Materials Conference, 20th AIAA/ASME/AHS Adaptive Structures Conference, 14th AIAA, American Institute of Aeronautics and Astronautics, Honolulu, Hawaii, HI, USA, 23–26 April 2012. [\[CrossRef\]](#)
2. Zhang, H.; Ma, L.; Sun, J.; Lin, H.; Thüerer, M. Digital Twin in Services and Industrial Product Service Systems: Review and Analysis. *Procedia CIRP* **2019**, *83*, 57–60. [\[CrossRef\]](#)
3. Muszyńska, M.; Szybicki, D.; Gierlak, P.; Kurc, K.; Burghardt, A.; Uliasz, M. Application of Virtual Reality in the Training of Operators and Servicing of Robotic Stations. In *Collaborative Networks and Digital Transformation*; Camarinha-Matos, L.M., Afsarmanesh, H., Antonelli, D., Eds.; Springer International Publishing: Cham, Germany, 2019; Volume 568, pp. 594–603. [\[CrossRef\]](#)
4. Tao, F.; Cheng, J.; Qi, Q.; Zhang, M.; Zhang, H.; Sui, F. Digital twin-driven product design, manufacturing and service with big data. *Int. J. Adv. Manuf. Technol.* **2018**, *94*, 3563–3576. [\[CrossRef\]](#)
5. Lu, Y.; Liu, C.; Wang, K.I.K.; Huang, H.; Xu, X. Digital Twin-driven smart manufacturing: Connotation, reference model, applications and research issues. *Robot. Comput. Integr. Manuf.* **2020**, *61*, 101837. [\[CrossRef\]](#)
6. Liu, J.; Du, X.; Zhou, H.; Liu, X.; Ei Li, L.; Feng, F. A digital twin-based approach for dynamic clamping and positioning of the flexible tooling system. *Procedia CIRP* **2019**, *80*, 746–749. [\[CrossRef\]](#)
7. Oleksy, M.; Budzik, G.; Sanocka-Zajdel, A.; Paszkiewicz, A.; Bolanowski, M.; Oliwa, R.; Mazur, L. Industry 4.0. Part I. Selected applications in processing of polymer materials. *Polimery* **2018**, *63*, 531–535. [\[CrossRef\]](#)
8. Stark, R.; Freseman, C.; Lindow, K. Development and operation of Digital Twins for technical systems and services. *CIRP Ann.* **2019**, *68*, 129–132. [\[CrossRef\]](#)
9. Tao, F.; Qi, Q.; Wang, L.; Nee, A.Y.C. Digital Twins and Cyber-Physical Systems toward Smart Manufacturing and Industry 4.0: Correlation and Comparison. *Engineering* **2019**, *5*, 653–661. [\[CrossRef\]](#)
10. Vachalek, J.; Bartalsky, L.; Rovny, O.; Sismisova, D.; Morhac, M.; Loksik, M. The digital twin of an industrial production line within the industry 4.0 concept. In Proceedings of the 2017 21st International Conference on Process Control (PC), Strbske Pleso, Slovakia, 6–9 June 2017; pp. 258–262. [\[CrossRef\]](#)
11. Zhang, C.; Zhou, G.; He, J.; Li, Z.; Cheng, W. A data- and knowledge-driven framework for digital twin manufacturing cell. *Procedia CIRP* **2019**, *83*, 345–350. [\[CrossRef\]](#)
12. Baskaran, S.; Niaki, F.A.; Tomaszewski, M.; Gill, J.S.; Chen, Y.; Jia, Y.; Mears, L.; Krovi, V. Digital Human and Robot Simulation in Automotive Assembly using Siemens Process Simulate: A Feasibility Study. *Procedia Manuf.* **2019**, *34*, 986–994. [\[CrossRef\]](#)
13. Bilberg, A.; Malik, A.A. Digital twin driven human-robot collaborative assembly. *CIRP Ann.* **2019**, *68*, 499–502. [\[CrossRef\]](#)
14. Malik, A.A.; Bilberg, A. Digital twins of human robot collaboration in a production setting. *Procedia Manuf.* **2018**, *17*, 278–285. [\[CrossRef\]](#)
15. Kousi, N.; Gkournelos, C.; Aivaliotis, S.; Giannoulis, C.; Michalos, G.; Makris, S. Digital twin for adaptation of robots' behavior in flexible robotic assembly lines. *Procedia Manuf.* **2019**, *28*, 121–126. [\[CrossRef\]](#)
16. Dröder, K.; Bobka, P.; Germann, T.; Gabriel, F.; Dietrich, F. A Machine Learning-Enhanced Digital Twin Approach for Human-Robot-Collaboration. *Procedia CIRP* **2018**, *76*, 187–192. [\[CrossRef\]](#)
17. Aivaliotis, P.; Georgoulis, K.; Arkouli, Z.; Makris, S. Methodology for enabling Digital Twin using advanced physics-based modelling in predictive maintenance. *Procedia CIRP* **2019**, *81*, 417–422. [\[CrossRef\]](#)
18. Burdea, G.; Coiffet, P. *Virtual Reality Technology*, 2nd ed.; J. Wiley-Interscience: Hoboken, NJ, USA, 2003.
19. Mihelj, M.; Novak, D.; Beguš, S. Intelligent systems, control and automation: Science and engineering. In *Virtual Reality Technology and Applications*; Springer: Dordrecht, The Netherlands, 2014.
20. Gallagher, A.G.; Ritter, E.M.; Champion, H.; Higgins, G.; Fried, M.P.; Moses, G.; Smith, C.D.; Satava, R.M. Virtual Reality Simulation for the Operating Room: Proficiency-Based Training as a Paradigm Shift in Surgical Skills Training. *Ann. Surg.* **2005**, *241*, 364–372. [\[CrossRef\]](#)
21. Ortiz, J.S.; Sánchez, J.S.; Velasco, P.M.; Quevedo, W.X.; Carvajal, C.P.; Morales, V.; Ayala, P.; Andaluz, V.H. Virtual Training for Industrial Automation Processes Through Pneumatic Controls. In *Augmented Reality, Virtual Reality, and Computer Graphics*; De Paolis, L.T., Bourdot, P., Eds.; Springer International Publishing: Cham, Germany, 2018; Volume 10851, pp. 516–532. [\[CrossRef\]](#)

22. Pérez, L.; Diez, E.; Usamentiaga, R.; García, D.F. Industrial robot control and operator training using virtual reality interfaces. *Comput. Ind.* **2019**, *109*, 114–120. [\[CrossRef\]](#)
23. Crespo, R.; García, R.; Quiroz, S. Virtual Reality Application for Simulation and Off-line Programming of the Mitsubishi Movemaster RV-M1 Robot Integrated with the Oculus Rift to Improve Students Training. *Procedia Comput. Sci.* **2015**, *75*, 107–112. [\[CrossRef\]](#)
24. Szybicki, D.; Kurc, K.; Gierlak, P.; Burghardt, A.; Muszyńska, M.; Uliasz, M. Application of Virtual Reality in Designing and Programming of Robotic Stations. In *Collaborative Networks and Digital Transformation*; Camarinha-Matos, L.M., Afsarmanesh, H., Antonelli, D., Eds.; Springer International Publishing: Cham, Germany, 2019; Volume 568, pp. 585–593. [\[CrossRef\]](#)
25. Oyekan, J.O.; Hutabarat, W.; Tiwari, A.; Grech, R.; Aung, M.H.; Mariani, M.P.; López-Dávalos, L.; Ricaud, T.; Singh, S.; Dupuis, C. The effectiveness of virtual environments in developing collaborative strategies between industrial robots and humans. *Robot. Comput. Integr. Manuf.* **2019**, *55*, 41–54. [\[CrossRef\]](#)
26. Matsas, E.; Vosniakos, G.C. Design of a virtual reality training system for human–robot collaboration in manufacturing tasks. *Int. J. Interact. Des. Manuf.* **2017**, *11*, 139–153. [\[CrossRef\]](#)
27. Tahriri, F.; Mousavi, M.; Yap, H.J. Optimizing the Robot Arm Movement Time Using Virtual Reality Robotic Teaching System. *Int. J. Simul. Model.* **2015**, *14*, 28–38. [\[CrossRef\]](#)
28. Yap, H.J.; Taha, Z.; Md Dawal, S.Z.; Chang, S.W. Virtual Reality Based Support System for Layout Planning and Programming of an Industrial Robotic Work Cell. *PLoS ONE* **2014**, *9*, e109692. [\[CrossRef\]](#) [\[PubMed\]](#)
29. Villani, V.; Pini, F.; Leali, F.; Secchi, C.; Fantuzzi, C. Survey on Human-Robot Interaction for Robot Programming in Industrial Applications. *IFAC-PapersOnLine* **2018**, *51*, 66–71. [\[CrossRef\]](#)
30. Pan, Z.; Polden, J.; Larkin, N.; Van Duin, S.; Norrish, J. Recent progress on programming methods for industrial robots. *Robot. Comput. Integr. Manuf.* **2012**, *28*, 87–94. [\[CrossRef\]](#)
31. Neto, P.; Mendes, N. Direct off-line robot programming via a common CAD package. *Robot. Auton. Syst.* **2013**, *61*, 896–910. [\[CrossRef\]](#)
32. Pettersen, T.; Pretlove, J.; Skourup, C.; Engedal, T.; Lokstad, T. Augmented reality for programming industrial robots. In Proceedings of the Second IEEE and ACM International Symposium on Mixed and Augmented Reality, Tokyo, Japan, 10 October 2003; pp. 319–320. [\[CrossRef\]](#)
33. Ong, S.K.; Yew, A.W.W.; Thanigaivel, N.K.; Nee, A.Y.C. Augmented reality-assisted robot programming system for industrial applications. *Robot. Comput. Integr. Manuf.* **2020**, *61*, 101820. [\[CrossRef\]](#)
34. Mueller, F.; Deuerlein, C.; Koch, M. Intuitive Welding Robot Programming via Motion Capture and Augmented Reality. *IFAC-PapersOnLine* **2019**, *52*, 294–299. [\[CrossRef\]](#)
35. Araiza-Illan, D.; De San Bernabe, A.; Hongchao, F.; Shin, L.Y. Augmented Reality for Quick and Intuitive Robotic Packing Re-Programming. In Proceedings of the 2019 14th ACM/IEEE International Conference on Human-Robot Interaction (HRI), Daegu, Korea, 11–14 March 2019; p. 664. [\[CrossRef\]](#)
36. Quintero, C.P.; Li, S.; Pan, M.K.; Chan, W.P.; Van der Loos, H.F.M.; Croft, E. Robot Programming Through Augmented Trajectories in Augmented Reality. In Proceedings of the 2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Madrid, Spain, 1–5 October 2018; pp. 1838–1844. [\[CrossRef\]](#)
37. Jost, T.A.; Nelson, B.; Rylander, J. Quantitative analysis of the Oculus Rift S in controlled movement. *Disabil. Rehabil. Assist. Technol.* **2019**, 1–5. [\[CrossRef\]](#)

