electronics

MDPI

Article

Creating Digital Twins of Robotic Stations Using a Laser Tracker

Dariusz Szybicki *, Magdalena Muszyńska, Paulina Pietruś, Andrzej Burghardt and Krzysztof Kurc

Department of Applied Mechanics and Robotics, Faculty of Mechanical Engineering and Aeronautics,
Rzeszów University of Technology, al. Powstancow Warszawy 8, 35-959 Rzeszów, Poland;
magdaw@prz.edu.pl (M.M.); p.pietrus@prz.edu.pl (P.P.); andrzejb@prz.edu.pl (A.B.); kkurc@prz.edu.pl (K.K.)
* Correspondence: dszybicki@prz.edu.pl; Tel.: +48-(17)-8651843

Abstract: This article deals with the design and creation of digital twins of robotic stations. A literature review of digital twins, robot programming methods and laser tracker applications is presented. This paper shows that the construction of digital twins is closely related to one of the most popular methods of robot programming, i.e., off-line programming. In the case of digital twins of robotic stations, modeling accuracy and two-way communication with the real station proved to be crucial. The article proposes a methodology for solving the basic problem of off-line robot programming, i.e., the limited accuracy of the representation of the station and the details. The algorithm of proceeding in the case when the station already exists and its digital model is built and the case when the digital model is first created and the real solution is built on its basis is shown. According to the developed methodology, a digital twin of a real robotic station was created and the possibilities arising from the use of virtual tools were shown. The developed digital twin has the ability to communicate with advanced Matlab 2021-type tools, uses cloud solutions and virtual and augmented reality for training, simulates physical phenomena and provides the ability to accurately program robots off-line.

Keywords: digital twin; industrial robots; robot programming methods; virtual reality; augmented reality; laser tracker

check for updates

Citation: Szybicki, D.; Muszyńska, M.; Pietruś, P.; Burghardt, A.; Kurc, K. Creating Digital Twins of Robotic Stations Using a Laser Tracker. *Electronics* **2024**, *13*, 4271. https://doi.org/10.3390/electronics13214271

Academic Editor: Giuseppe Prencipe

Received: 20 September 2024
Revised: 10 October 2024
Accepted: 29 October 2024
Published: 31 October 2024

1. Introduction

The development of 3D modeling technologies related to the idea of Industry 4.0, measurements, smart sensors or communication systems has significantly influenced the development of design methods for complex robotic systems. Currently, the creation of digital twins is associated with tools for designing and programming robotic stations. Commercially available tools allow us in a virtual environment to both design new and program existing robotic stations. It is possible to plan the positioning of robots and other station components, taking into account manipulator workspaces or available hall space. The use of appropriate software makes it possible to verify various organizational variants of the workstation and work scenarios while maintaining the possibility of easy and quick adjustments. All work is carried out with the safety of program testing (e.g., collision detection) due to simulation in a virtual environment. Fast and automatic generation of programs based on the geometry of workpieces is performed. The design in the virtual environment is mainly related to off-line programming methods for robots and the creation of digital twins.

The off-line robot programming method is based on generating robot paths using a virtual environment. It takes advantage of the accuracy of industrial robots. In detail, programming methods with a special focus on off-line robots have been discussed in works [1–3]. The authors of these works focused on the off-line programming method using CAD models [1], an absolute laser tracker [2] or virtual reality [3]. In works [4–7], robot programming methods are shown in a broader perspective. Development perspectives [4], the possibilities of using artificial intelligence [5] and the possibility of using Industry 4.0

elements in programming [6,7] are shown. In the off-line programming method, CAD models of individual robots and auxiliary equipment are imported into the 3D system. It is possible to use 3D models generated in CAD systems.

However, there are some application problems associated with designing workstations in a virtual environment and off-line programming, such as the following:

- Limited positioning accuracy of handling robots;
- Limited mapping of the geometry of the robotic workstation by its virtual model;
- Errors in modeling the workpieces on which the robot operates.

Ongoing research seeks solutions to defined problems. The subject of robot accuracy and available methods to improve it has been covered in articles [8–10].

In this article, an attempt is made to solve problems related to the limited degree of mapping of the workstation and workpieces. The aforementioned problems are related to the topics present in the idea of Industry 4.0, i.e., the methodology of creating digital twins and ways to model them.

The idea of creating digital twins has been known since the Apollo 13 mission in 1970 [11]. Back then, despite the lack of a definition of the buzzword, it was possible to return astronauts safely to Earth by conducting tests and simulations on a physical and virtual copy of the spacecraft. Various more- or less-detailed definitions of the phrase appeared much later. According to researchers at the University of Cincinnati, it is "a coupled digital model of a real machine that runs on a cloud platform and simulates the state of the real machine using integrated knowledge from both data-driven analytical algorithms and other available physical knowledge" [12]. The emergence of such technology was described by David Gelernter in 1991 in his book *Mirror Worlds* [13]. In industry, the idea of digital twins emerged through product life cycle management specialist Michael Grieves in 2002.

A detailed review of existing definitions, modeling methods and requirements for digital twins is included in the paper [14]. The authors of this article identified the most important applications of digital twins and their importance in the context of Industry 4.0. In order to describe the methods of modeling and creation of digital twins, they performed a systematic literature review based on keyword searches, filtering of datasets and classification of the available literature. The authors very accurately distinguished between digital models, digital shadows and digital twins. They pointed out the difference, which is the nature and direction of data flow between physical and virtual systems. A digital model is a digital version of a planned or existing physical object, and there is no automatic exchange of data between the physical model and the digital model. In the case of a digital shadow, data are transferred from the physical object to the digital model. Therefore, the model is updated with new information from the real world. In the concept of a digital twin, data flows between the existing physical object and the digital object and is fully integrated in both directions.

A very extensive review of articles on digital twins was conducted by the authors of the paper [15]. In the review, the applications of digital twins were divided into the areas of manufacturing, healthcare and smart cities. The paper includes an assessment of the technologies supporting the creation of digital twins, the challenges and an overview of the ongoing research. The authors note that applications of digital twins are currently focused on manufacturing, as evidenced by the large number of articles in this area.

The article [16] provides an overview of digital twin technology and its application domains, along with a detailed discussion of network requirements. Various ways can be adopted to divide digital twins. One criterion may be their degree of sophistication. The simplest variant of a digital twin can be considered a digital representation of a single device, such as a robot in a factory. These simplest twins can be used for digital prototyping or designing new products. Production digital twins that carry out some kind of process, e.g., machining or welding, or represent an entire production line can be considered as more advanced ones. Even more advanced are digital twins of entire advanced objects like a factory or an airplane. They capture data from a system of objects, such as an airplane or an entire factory. However, it is the Earth twin developed by the European Space Agency (ESA), which aims to build a dynamic digital replica of our entire planet [17], that is considered to be the most complex digital model.

Creating digital twins is a complex process. Building a functional system requires multiple stages of design, modeling and implementation. Usually, a digital model is created first, which is supplemented with two-way communication becoming a digital twin. At the core of this technology are virtual models. For this reason, the most important step to create a twin is to develop high-fidelity virtual models to replicate geometry, physical properties and behavior. The world's best-known providers of digital twin platforms are Siemens, Autodesk, Bosch, Dassault Systems and Ansys. Google's Supply Chain Twin and Azure Digital Twins, among others, provide cloud solutions for this technique. There are also Amazon's services such as Amazon SageMaker, Kinesis Data Streams, AWS Lambda and others. Siemens has developed a platform called MindSphere [18], which uses the concept of Industry 4.0 along with the data cloud. The platform is based on a system that connects machines and physical infrastructure with a digital twin.

Interesting solutions in the field of digital twins are offered by NVIDIA, which unveiled Omniverse Replicator, a platform for designing digital twins. It is an engine that generates synthetic data simulating physical ones for training deep neural networks. Omniverse offers users the ability to connect to multiple software ecosystems, including Epic Games' Unreal Engine 5, Reallusion 2022, OnShape 2021, Blender and Adobe 2022. The company has unveiled two implementations of this engine for synthetic data-generating applications: the NVIDIA DRIVE Sim, a virtual world in which it is possible to place the digital twin of autonomous vehicles, and the NVIDIA Isaac Sim [19], a virtual world for the digital twin of robotic manipulators.

The authors of the article [20] developed a digital twin-based modular platform for creating system-independent and human-centered industrial process simulations. The developed solution uses visual programming concepts. It has been tested by users, who reported high functionality and pointed out the advantages of the proposed approach.

The subject of building a digital twin in production control applications was addressed in the article [21]. The authors used digital twin (DT) and Industrial Internet of Things (IIoT) technology to build a multi-agent and cloud–edge orchestration framework for production control. The developed solution creates a production line model in the cloud to support the optimal configuration of distributed idle production resources.

In the article [22], the authors performed an extensive literature review on digital twins with the integration criterion in mind. They note that at the time the article was written (2018), the number of publications on the highest stage of development, i.e., the twin, was much smaller than for digital models or shadows.

Real-world examples of applications of digital twins can be found in the areas of system servicing [23,24], product design and manufacturing systems [25]. A specific application area of digital twins is robotics, mainly robotic manufacturing systems [26]. The article [27] presents the application of the digital twin in assembly issues. The authors applied the latest technologies such as the Internet of Things, cloud technologies and Web 3D to create a digital twin. Assembly processes were also characterized in the paper [28]. The authors performed a review of existing digital assembly technologies and addressed the optimization of the assembly and commissioning process. In addition, a case study of the assembly process of a high-precision electrohydraulic servo valve was presented to verify the feasibility of the method. An interesting approach to digital twins was proposed by the authors of [29] who developed a hybrid modeling approach that combines a mechanism model and a time series forecasting model. This comprehensive model of a digital twin of a worker with a full life cycle is intended for production operations. The problem of building digital twins of industrial robots is discussed in [30], where a standardized methodology and a hierarchical, modular and generic architecture are proposed to represent a comprehensive and variable industrial robot digital twin (IRDT).

After a detailed explanation of the concept of digital twins and familiarization with their division and examples of applications, the authors decided to develop a methodology for building accurate digital twins of robotic stations. The required accuracy of twins is related to their use for off-line programming of robots and the necessary precise positioning of, for example, positioners, jigs or workpieces. Satisfactory accuracy here should be understood as the position error of workpieces close to the accuracy of the robot. The accuracy of a robot, for example, using the ABB Absolute Accuracy add-on, can be around 0.4 mm. The authors decided to use an absolute laser tracker as a precision tool to solve the problem of limited position mapping. The laser tracker is a tool that enables various types of precise geometric measurements with a long range, reaching up to 40 m in three-dimensional space. It can be used for automated position control, as well as for 3D scanning of objects, for example. In detail, the subject of laser trackers has been described in the works [10,31–33]. In this work, an attempt is made to solve the problem of limited mapping of the position and details on which, for example, the robot works. In addition to presenting the methodology, the authors decided to develop a digital model and create a digital twin. Referring to the cited divisions regarding digital twins in the developed solution, the data flows take place between the existing physical object and the digital object and are fully integrated in both directions. The developed digital twin is related to the manufacturing industry. It is a digital twin of "medium size", which can be described as a manufacturing one. Due to the tools adopted to create the digital twin, the developed accurate digital model can be imported into other platforms to work on digital twins.

2. Creating Digital Models of Robotic Stations Using Laser Trackers

The use of virtual tools for the design and programming of robotic stations allows for graphical (CAD) representation of station components and allows for the creation of advanced work programs and complex station logic. It contributes to shorter production downtime and faster integration of the robotic station. The aforementioned problems of modeling and positioning accuracy of station elements prompted the authors of this paper to propose a methodology to show the solution to the problem, speeding up the process of station programming and implementation. The proposed methodology is based on the creation of an accurate digital twin using a laser tracker, 3D CAD systems and off-line robot programming tools. The first stage of the implementation of the digital twin is the digital model, while in the second stage the connection to the real station is made.

The developed methodology includes two main variants. The first one is when the robotic station is physically built and its digital twin is needed. The second one is when the robotic station is only at the design stage and a digital model will be created first and the real station will be built based on it. In the case of work related to robotic stations, both variants and even their combinations can be encountered.

An existing real station often requires a digital twin for off-line programming applications and, for example, process implementation for new workpieces. The construction of a digital twin of a real station may also be prompted by the need to conduct operator or programmer training using the model. In this case, training conducted on the model is safe and does not require production downtime. Such solutions with examples of implementation are shown in the works [24,34,35].

Variants, when on the basis of a digital model the real station is created, allow for selecting the appropriate robot, determining the position and orientation of the station elements, determining the working spaces, determining the position of the workpieces (Figure 1) or planning the necessary amount of space in the hall.

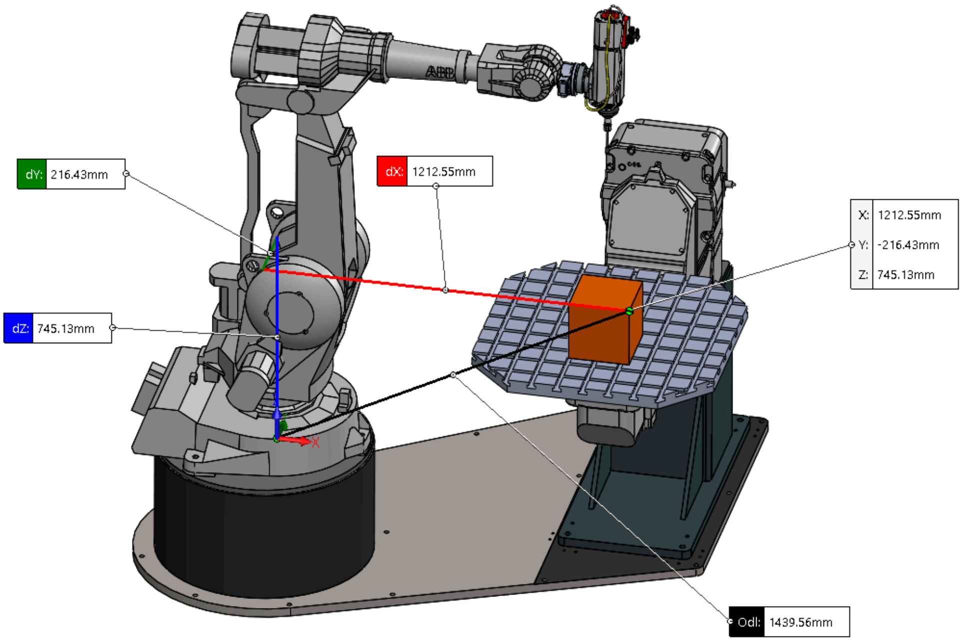


Figure 1. A digital model of a robotic station with the position of a component defined relative to the robot base frame.

By having a digital twin of the robotic station, it is possible to make the station's software even before it is physically built. Off-line robot programming techniques are used for this purpose. Developing the station's software before it is built significantly reduces the implementation time and allows us to accelerate the implementation into production, which reduces costs.

Situations where real stations are subject to modifications resulting, for example, from a change in the production profile can be considered as a combination of variants. The construction of a digital twin in such a case makes it possible both to develop off-line programs for new details and to simulate new organizational variants before their introduction.

Regardless of the chosen option, the tools already mentioned are required to implement the developed methodology:

- Three-dimensional CAD systems with libraries of finished components;
- Off-line robot design and programming software;
- An absolute laser tracker.

The capabilities and multitude of applications of CAD systems are well known. CAD is software that, with the help of a computer, allows the creation of 2D, 3D designs, technical documentation, calculations and various types of analysis. Sophisticated CAD systems help design and solve advanced engineering projects [36]. These systems continue to develop rapidly and multi-directionally. The authors of the article [37] suppose that there will be integration of augmented reality and artificial intelligence in them. In the process of designing with their participation, attention should be paid to libraries of ready-made components that can be used to build, for example, a robotic station.

Off-line robot design and programming tools, as already mentioned, allow advanced station design and off-line robot programming, and some of them communicate with real stations. For these tools, a distinction should be made between solutions offered by robot manufacturers and solutions that are somewhat universal. Examples of robot manufacturers' solutions include RoboGuide 8.1 from Fanuc (Oshino-mura in Japan), KUKA .Sim 3.1 from Kuka (Augsburg) or RobotStudio 2021 from ABB (Switzerland Zurich). These software products have very advanced functions related to the design and programming of robots from a given manufacturer. However, robot manufacturers' tools lack advanced solutions related to 3D modeling or extensive 3D model databases. In addition to dedicated and specialized solutions offered by robot manufacturers, there are many other programs on the market designed to support programming and simulation of their robots. These tools usually have databases of robot models from leading manufacturers and allow the creation of off-line programs. These include, for example, MASTERCAM 2024, Delmia Igrip 5, RobCAD 3.0.0 and Robot 3D 2020. Universal tools do not have the ability to support advanced features dedicated to a particular manufacturer, such as advanced robot motion types, creation of operator panels or direct communication with the real station. However, all of these tools for designing and programming off-line robots do not have the full capabilities of CAD systems. The solution to these problems is to use both 3D CAD systems and dedicated solutions from manufacturers and exchange data between them. The data exchange takes place through a two-way file exchange between the systems.

The last element necessary for the implementation of the developed methodology for creating accurate digital twins of robotic stations is a laser tracker. Its most important components are an absolute rangefinder and a laser interferometer. The trackers are used in testing the accuracy and repeatability of robotic manipulators and for their calibration. Laser trackers use mirrors as targets to reflect the laser beam generated by the device. Most often, these are retroreflectors of spherical design. The mirrors in the retroreflector are mounted in a precise way so that their point of contact (the apex) coincides exactly with the center of the SMR, or Spherically Mounted Retroreflector. This makes it possible to precisely determine the coordinates of the retroreflector′s position in three-dimensional space. The retroreflectors are offered with so-called bases that allow measurement of planes, edges or hole positions (Figure 2).

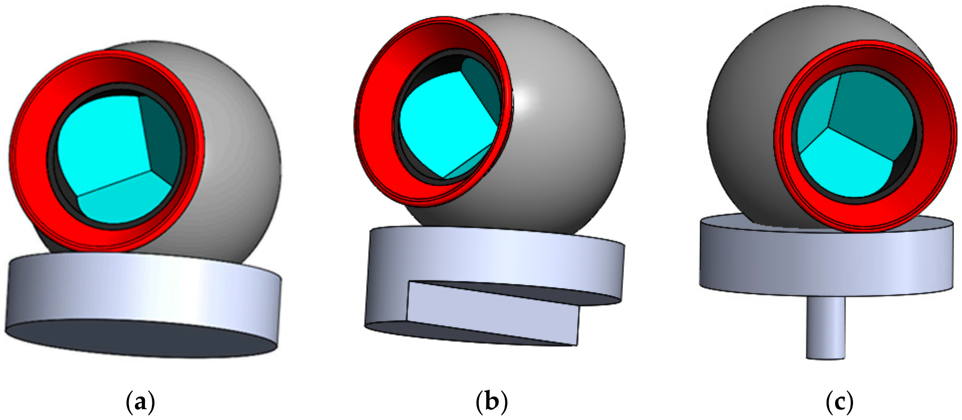

(**a**) (**b**) (**c**)

Figure 2. Models of retroreflectors with bases: (**a**) allowing measurement of planes; (**b**) edges; (**c**) hole positions.

The choice of the laser tracker (the model used in the developed solution was the Leica AT 960 (Wetzlar in Germany) was influenced by its very high accuracy of 58 µm, its range of 20 m and the software developed in previous work for sharing data using network communication standards. The concept of the methodology associated with variant one, when the robotic station is physically built, is shown in Figure 3.

To explain the data flow shown in the diagram, it is necessary to provide information related to the coordinate systems found in robotic stations. This issue is described in detail in the work [10]. In this article, the most relevant scheme related to the determination of coordinate systems will be shown (Figure 4).

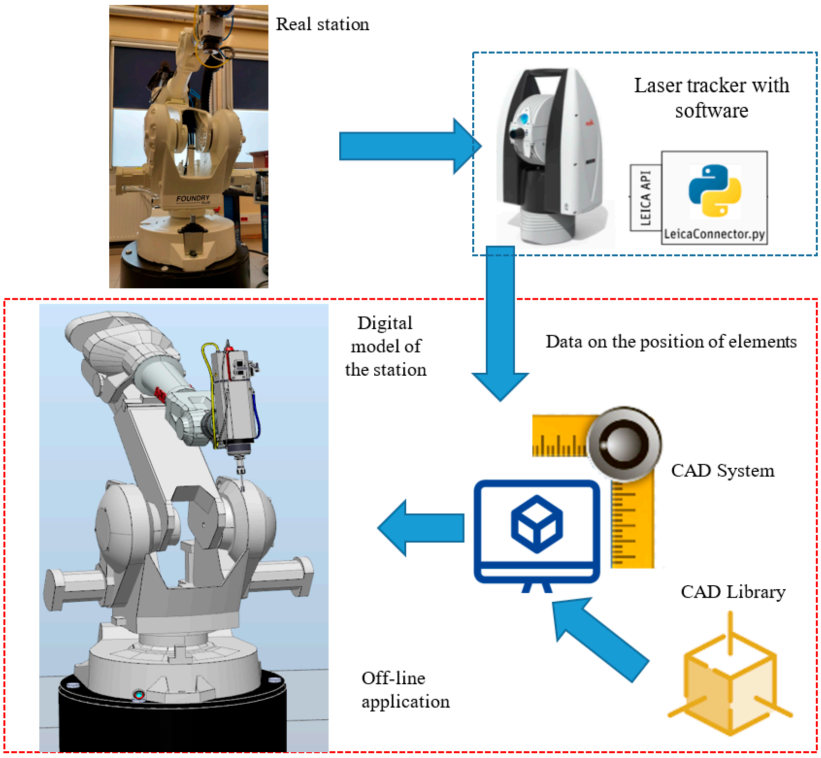


Figure 3. A diagram of the implementation of the methodology associated with the variant when the robotic station is physically built and its digital twin is created.

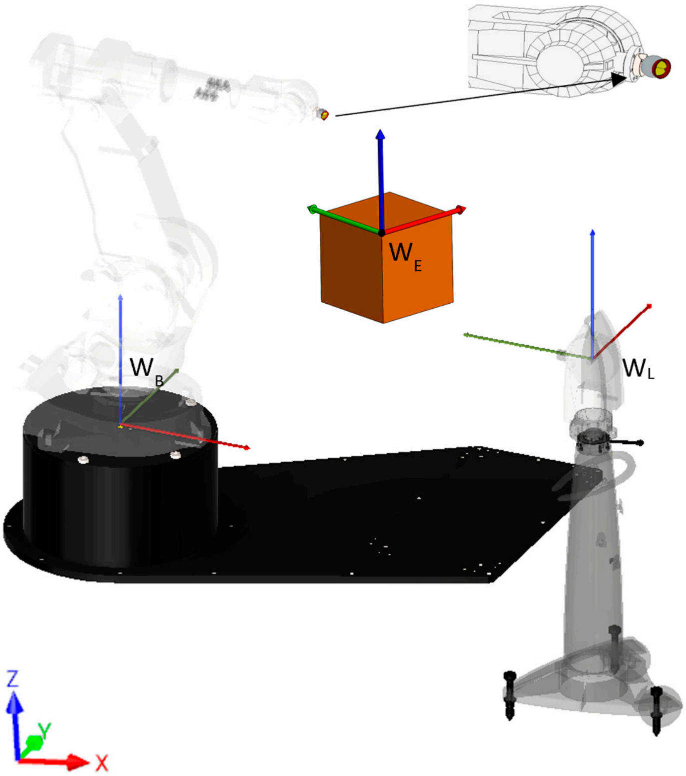


Figure 4. Position of base frame WB, Leica tracker WL and element WE.

The most important coordinate system found in robotic stations is the so-called Base-Frame. Relative to it, the position and orientation of the tool are determined, the positions and orientations of the path points are determined, and other coordinate systems are defined. The method of determining this system, located in the base of the robot, is shown in the work [10]. This coordinate system is the basis for creating a digital twin in both CAD systems and off-line programming software. Relative to this system, labeled WB in Figure 4, the position of all station elements must be determined. It is important to note that determining the position and orientation of the coordinate system associated with the element, labeled WE in Figure 4, requires measuring the positions of nine points (three on each plane) located on the walls of the element. Having a CAD model and the so-called zero-positioning reduces the number of necessary measurements. The position of the points measured in the real station, determined precisely through the laser tracker in the WL tracker system, is transferred to the position in the WB robot base frame. The flow of the element's position and orientation data into the CAD software is shown in Figure 5.

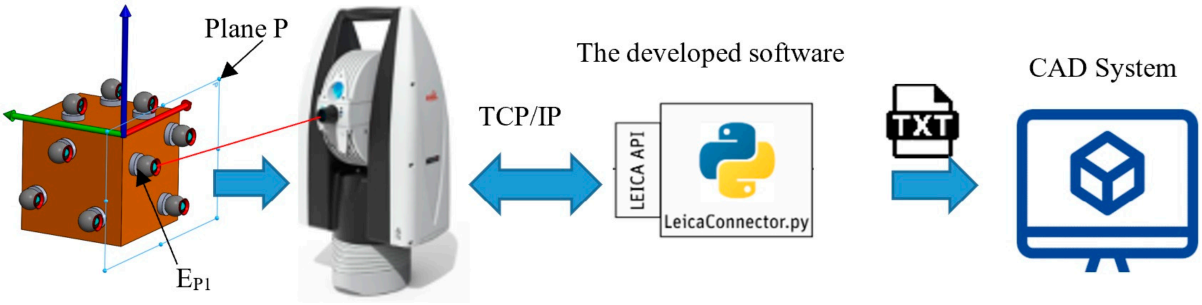


Figure 5. The flow of the element's position and orientation data into the CAD software.

The developed LeicaConnector 1.0 software is responsible for converting the positions of points from the WL tracker system to coordinates in the WB base frame. The process of developing and using this software is characterized in the paper [38]. The software saves the coordinates of the points as a text file, which can be automatically loaded by parametric CAD programs such as SolidWorks or Inventor. Based on the positions of the points, the equations of the planes can be determined as written in Equation (3). The positions of the points on a given plane P are stored as vectors containing the coordinates of the points in the assumed reference system WB.

$$E_{P1} = \begin{bmatrix} x_{EP1} \\ y_{EP1} \\ z_{EP1} \end{bmatrix}, E_{P2} = \begin{bmatrix} x_{EP2} \\ y_{EP2} \\ z_{EP2} \end{bmatrix}, E_{P3} = \begin{bmatrix} x_{EP3} \\ y_{EP3} \\ z_{EP3} \end{bmatrix} \tag{1}$$

The equation of the plane P passing through the point E_{P1} with normal vector $\vec{n}$ = $[A, B, C]$ is expressed by the formula:

$$A(x - x_{EP1}) + B(y - y_{EP1}) + C(z - z_{EP1}) = 0 \tag{2}$$

If the points E_{P1}, E_{P2} and E_{P3} are three fixed non-collinear points and the point E_P = (x, y, z) is any point of the plane P, then the equation for determining the plane is expressed by the formula:

$$\mathrm{P}: \begin{vmatrix} x - \mathrm{x_{EP1}} & y - \mathrm{y_{EP1}} & z - \mathrm{z_{EP1}} \\ \mathrm{x_{EP2}} - \mathrm{x_{EP1}} & \mathrm{y_{EP2}} - \mathrm{y_{EP1}} & \mathrm{z_{EP2}} - \mathrm{z_{EP1}} \\ \mathrm{x_{EP3}} - \mathrm{x_{EP1}} & \mathrm{y_{EP3}} - \mathrm{y_{EP1}} & \mathrm{z_{TP3}} - \mathrm{z_{EP1}} \end{vmatrix} = 0 \tag{3}$$

CAD software, based on the coordinates of points, is able to determine planes and, based on them, generate mates. When designing advanced assemblies in CAD software, the so-called assembly coordinate system is always adopted. Relative to it, the coordinates of parts, i.e., successive elements of the assembly, are determined parametrically by means of so-called mates. If the WB robot base frame is taken as the assembly system, then it is possible to build an accurate digital model of the real station. Based on the points measured by the laser tracker in CAD software, planes can be generated. With the planes, CAD models of the station elements in possession are associated (Figure 6). With the acquired solution, it is possible to obtain an accurate parametric digital model of the robotic station containing the positions and orientations of the walls, floor or controller positions.

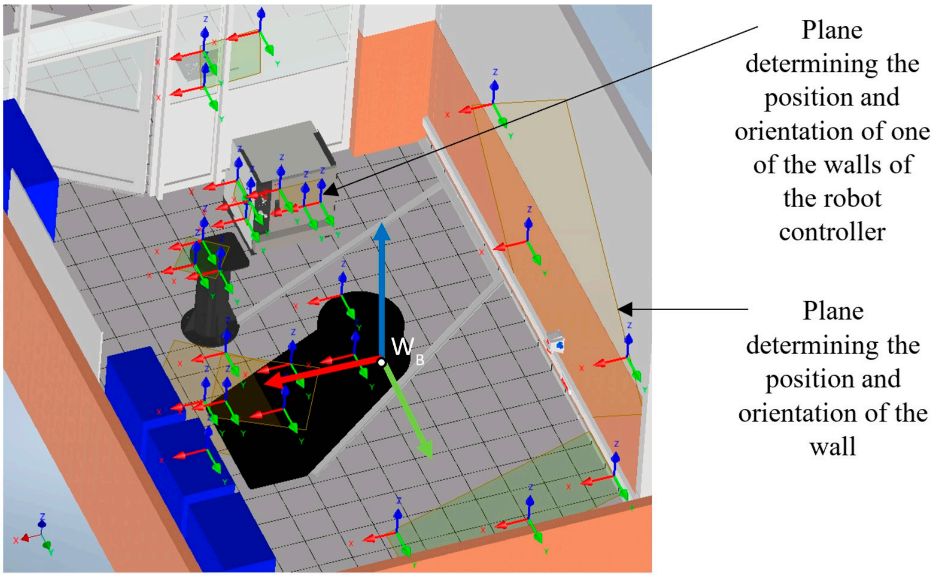


Figure 6. A CAD model of the robotic station with planes determining the positions of the components.

The exact digital model is not yet a digital twin; the way to combine the virtual model with the real station will be explained in Section 4. In the case where the robotic station is only at the design stage and a digital model of it was created first, a suitable scheme of action should be developed (Figure 7). For this purpose, the solution shown earlier (Figure 3) can be reversed and slightly modified.

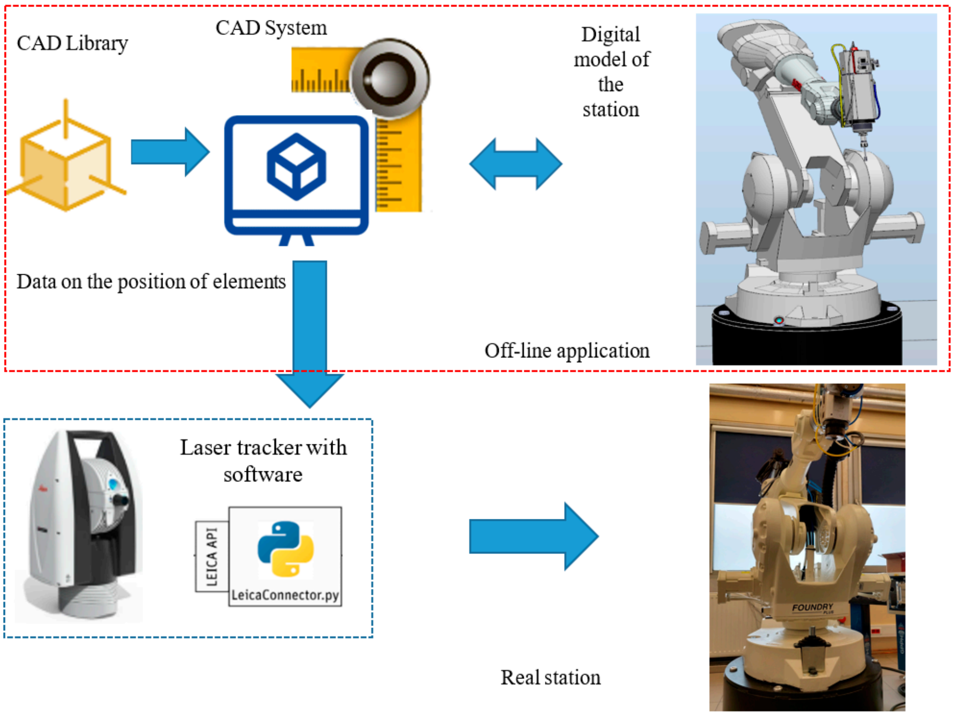
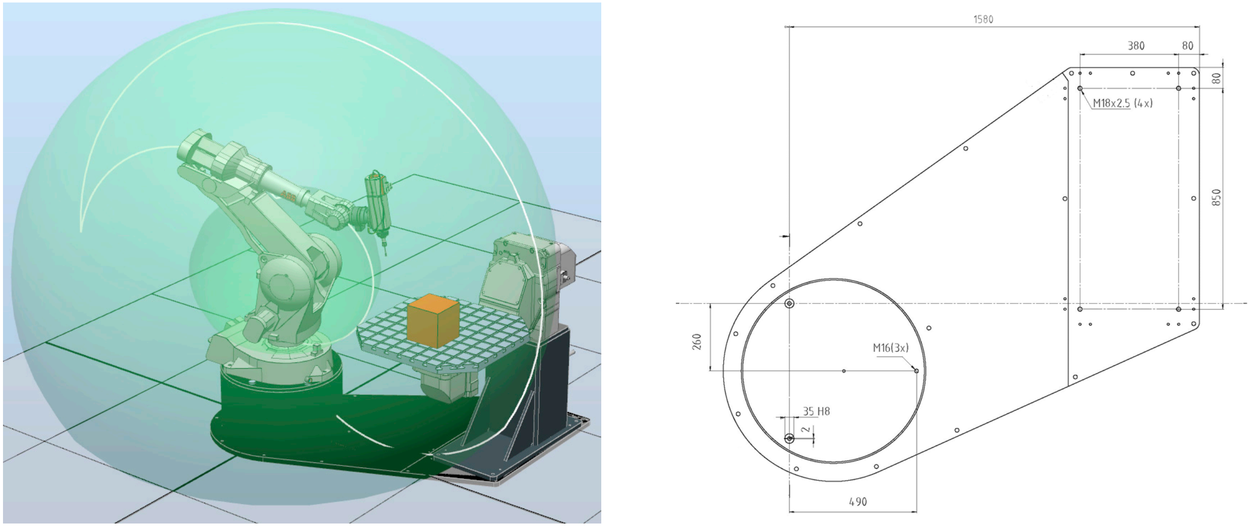


Figure 7. A diagram of the implementation of the methodology associated with the variant when the robotic station is only at the design stage and first a digital model is created, on the basis of which the real station will be built.

The diagram shown in Figure 7 indicates with a red dotted line the elements in which the design process of the robotic station is carried out. The cooperation of CAD software and off-line programming tools supplemented with models from component libraries allows for the rapid development of an advanced station model. Due to the exchange of data between systems, it is possible to easily check whether the workpiece lies in the manipulator's workspace, taking into account the tool used, as shown in Figure 8a. The digital model available in CAD software allows the design of, for example, a model of the robot pedestal and positioner and the generation of manufacturing drawings (Figure 8b).



(**a**) (**b**)

Figure 8. Effects of using the digital model: (**a**) the workspace of the manipulator considering the tool used; (**b**) a fragment of the detailed drawing of the pedestal.

Having a ready digital model of the station and the components made on its basis, such as the pedestal, the tool mounting on the robot or the workpiece positioning device, it is possible to move to the next stage, that is, the assembly of the real station. This is the last stage shown in the diagram in Figure 7. To build the real station on the basis of the model, a laser tracker and dedicated software are again used (Figure 9). The assembly begins with setting up the pedestal and the robot. Then, the real position of the WB robot base frame is determined with the tracker. The next step is the precise positioning and mounting of components such as positioners, fences, workpieces or tool changers.

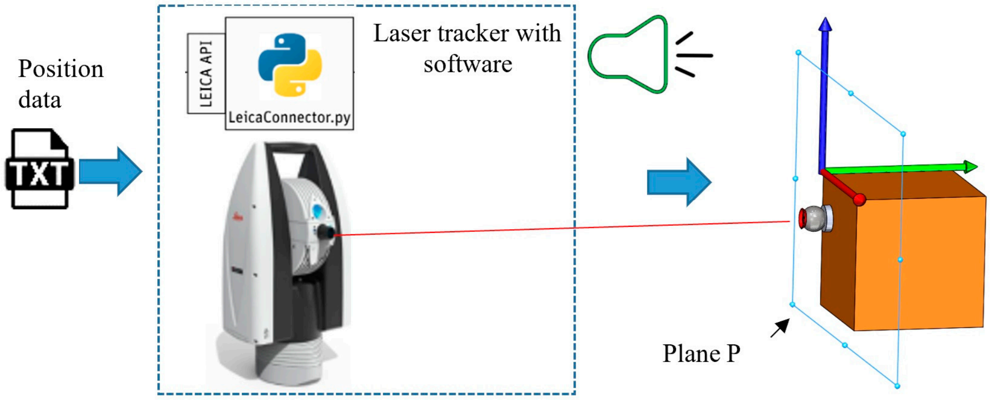
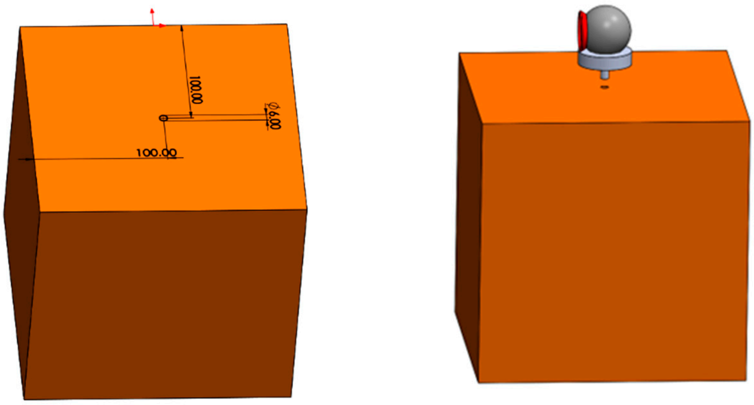


Figure 9. The assembly of components using a laser tracker and dedicated software.

Having the data of the planes defined with respect to the base frame, the retroreflector is moved on a dedicated base together with the mounted component until it is on the selected plane. The distance of the point $E_P = (x, y, z)$ from the plane P: $Ax + By + Cz + D = 0$ is expressed by the formula:

$$d(E_P, P) = \frac{|Ax + By + Cz + D|}{\sqrt{A^2 + B^2 + C^2}} \tag{4}$$

The position of the retroreflector on the plane with the selected accuracy is informed by sound, which is a function of dedicated software. The sound information is helpful during assembly under production conditions. A function of the software is also to inform in real-time coordinates about the position of the retroreflector and the distance to the selected plane. When designing elements of a robotic station for mounting with a tracker, it can be useful to plan the exact holes intended for mounting the retroreflector with a base (Figure 10).



(**a**) (**b**)

Figure 10. Station element model: (**a**) mounting hole position; (**b**) element with hole and retroreflector with base.

The use of dedicated precision holes facilitates the assembly of station components reducing the necessary number of measurement points.

So far, the idea of building digital models of robotic stations using a laser tracer has been explained. In the presented diagrams, models available in libraries of ready-made components were used to build the digital model. The following section will analyze the case when there are no models or their geometric parameters are not satisfactory.

3. Creating Digital Models by Modeling Methods

During the design of robotic stations, it is often found that accurate CAD models of important components are not available in the databases shown in the diagram (Figure 7). When a digital model is first made and then the real station is built, access to component models is usually easier. It is possible to match components with available models, and manufacturers of, for example, grippers, spindles or fences will provide precise models of their products. Often, it turns out that the level of detail in the available models is insufficient. This situation can arise when the model in question is to be used for training or for the design of cooperating devices. This article presents two cases of creating digital models by classical CAD modeling methods and their possibilities. It is possible to make models using, for example, a 3D structured light scanner or photogrammetry, but in this case they were not used. The developed models are elements of the digital twin characterized in Section 4.

The first object modeled was the most essential component of a robotic station, i.e., the industrial robot. This is because it turns out that major robot manufacturers, such as Kuka, ABB and Kawasaki, provide models of their robots with very little detail. Accurate 3D models are not available either on manufacturers' websites or in programming tools off-line. The available models do not have internal structure elements, and electrical wiring connections or pneumatics components are not shown there. In the technical documentation, the appearance and dimensions are shown in the drawings, but details are missing on the 3D models. The popular ABB IRB2400 general-purpose robot was chosen as an example. The 3D model (Figure 11) available in the manufacturer's RobotStudio 2021 software (Switzerland Zurich) and on the website lacks electrical, pneumatic and drive release buttons and connections to the robot controller.

Figure 11. CAD model of IRB2400 robot along with photos of real components.

The model of the robot shown in Figure 11 also lacks internal components such as drives or gears. The model offered on the manufacturer's website does not allow the available components to be linked in such a way that the lengths of the members are compliant with the drawing documentation, and it is possible to determine the TCP position by assigning six axis angles. As a result of the aforementioned shortcomings of the model, it was decided to make an in-house CAD model of the IRB2400 robot (Figure 12) for use in training and designing the robot's hardware and for use in simulations in Matlab software. In the developed solution, it was also decided to model the most relevant internal components (Figure 12).

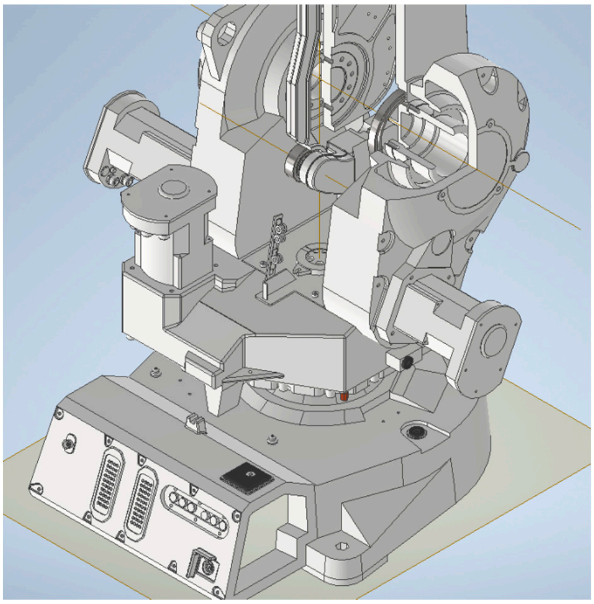
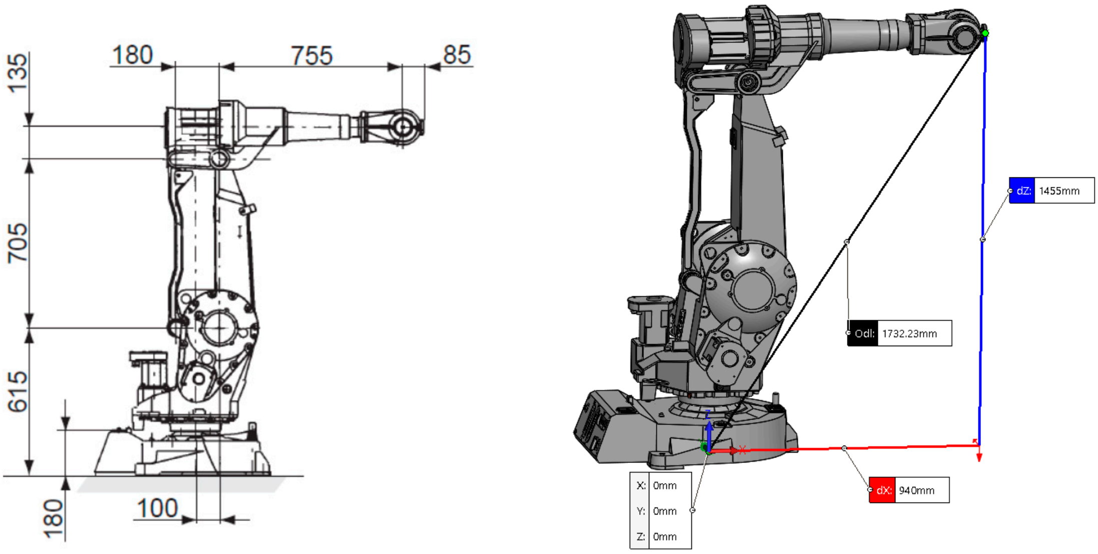


Figure 12. IRB 2400 robot model during design process.

During the design process, technical documentation from ABB, photos of the real robot and information on the dimensions and weights of components available in on-line spare parts databases were used. For robot programming and modeling applications, it is important that the axes of rotation of the model's members coincide with the real ones and that the position of the so-called tool, when the angles of all axes are zero, complies (Figure 13). In this case, the model performs the task of simple and inverse kinematics.



(**a**) (**b**)

Figure 13. IRB2400 robot: (**a**) excerpt from manufacturer's technical documentation; (**b**) developed 3D model.

The developed model, due to the addition of the most important internal components and the adoption of member masses, can be used for FEM (finite element method)-type analyses (Figure 14).

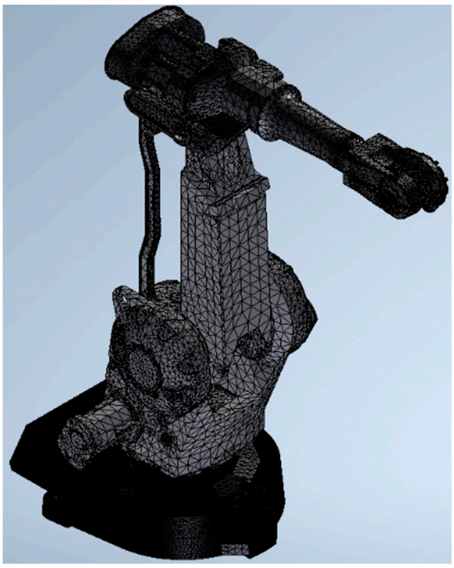


Figure 14. IRB2400 robot model during FEM analysis.

It is possible to perform strength simulations, frequency, modal or static analyses for the developed model. It is possible to perform simple analyses in SolidWorks or Inventor, as well as export to sophisticated tools such as Abaqus or Adams.

In the case of the second 3D model, the modeling process of which was presented, its purpose is primarily for training. A detailed 3D model was made for a cabinet with automation components (Figure 15a).

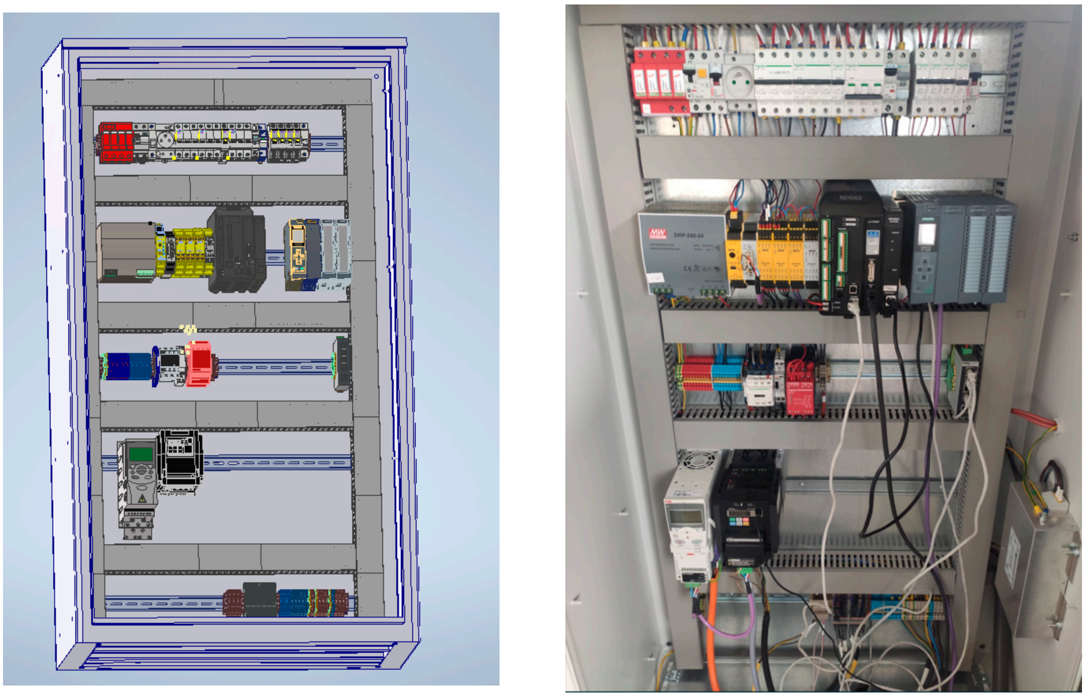

(**a**) (**b**)

Figure 15. Cabinet with automation components: (**a**) 3D model of cabinet; (**b**) photo of real cabinet.

This is a very important element from the point of view of designing robotic stations and other industrial installations. Such a cabinet is first designed using dedicated tools like ePLAN. Based on the documentation and 2D drawings, a model of the station is made, including components such as relays, PLCs and frequency converters. Having a 3D model of the cabinet, it is possible to predict the amount of space required for the individual components and offer training related to the operation of automation systems. An interesting application of a model of such a cabinet is performing FEM simulations of the heat flow within the cabinet. The 3D CAD model of the cabinet is part of the digital twin created in Section 4.

4. Available Data Exchange Methods for Digital Twins of Robotic Stations

A digital twin meets the criteria distinguishing it from a digital model and a digital shadow if the flow of data between an existing physical object and a digital object is in both directions. Only a two-way exchange of information makes it possible to use the benefits of implementing the idea of digital twins. In the general case, the data exchange of the real object–digital model is carried out using network communication standards or data file sharing methods. Showing the real case for robotic stations requires defining the tool in which this will be implemented. Due to its experience and hardware base, RobotStudio software was used to create the digital twin. It represents one of the most advanced tools for programming industrial robots. From the point of view of creating digital twins, the most important thing is the ability to communicate with various types of software using the TCP/IP protocol family and the OPC standard (Figure 16).

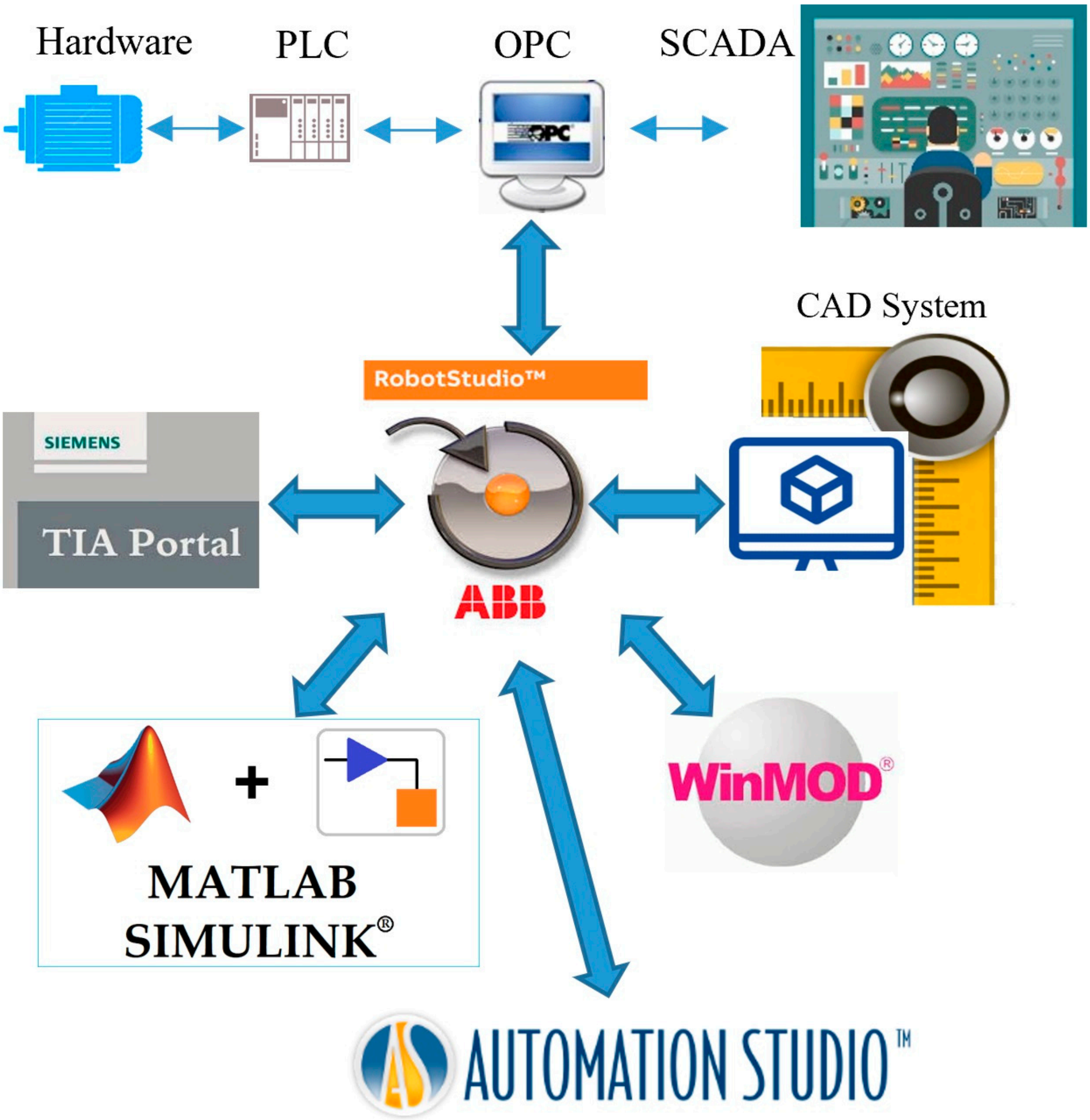


Figure 16. Graphic showing of RobotStudio software's ability to communicate with other software.

For RobotStudio and most off-line robot programming tools, the most important method of exchanging data with the real station is the so-called synchronization (Figure 17). It involves transferring programs from the real robot controller to the virtual system and vice versa.

Synchronization allows a robot program to be generated in a virtual environment and sent to a real controller. The program can be created based on a CAD model of the workpiece and other station components. It is linked to the robot's base frame. Synchronization can also be performed the other way around. A program developed on the real robot, e.g., by on-line programming methods, can be transferred to the virtual environment for editing, improvement or expansion. A feature of synchronization is that it is performed at the command of use and mainly involves robot programs. This is different from communication using TCP/IP, the OPC standard or data exchange by writing and reading files. In the case of these standards, data exchange can take place continuously and any information can be transmitted. TCP/IP and OPC standards are often used to transfer robot operating parameters and production data and manage the real station through tool selection or coordinate systems. An example of the use of the TCP/IP protocol for monitoring the performance of a robotic station is shown in the works [39–42].

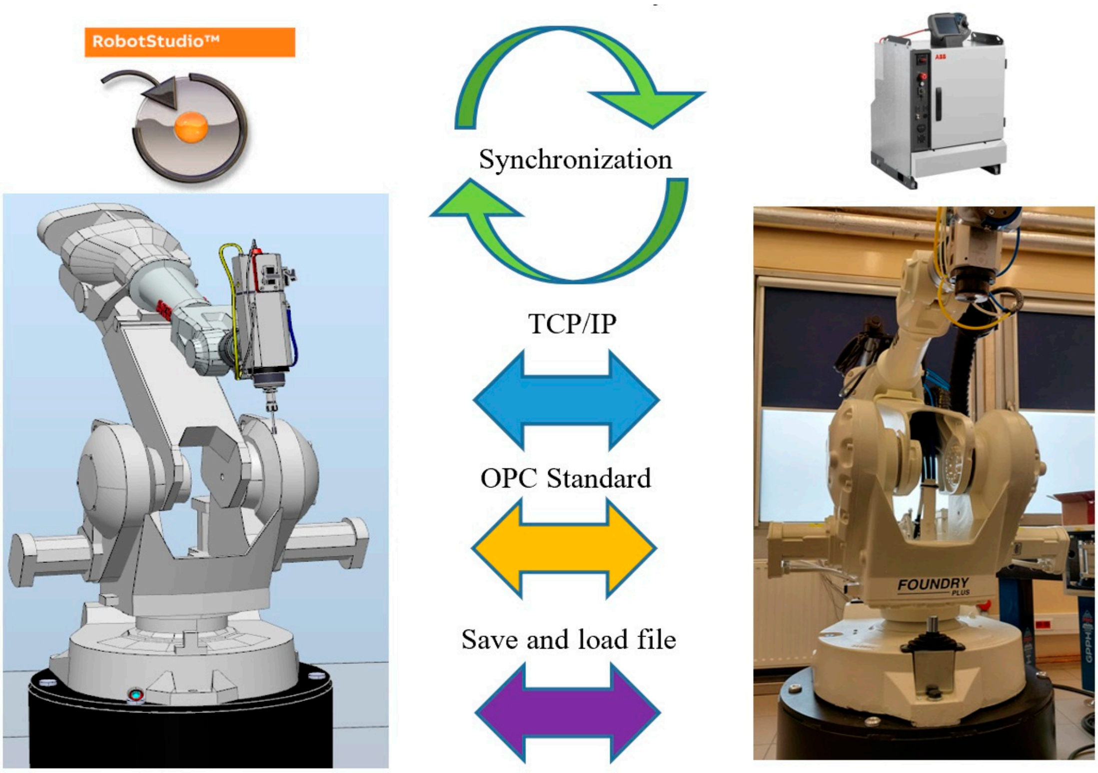


Figure 17. Most commonly used methods of exchanging data between digital model and real station.

The possibilities of data exchange between the digital model in RobotStudio software and the real station shown in Figure 18 prove a full implementation of the concept of digital twins.

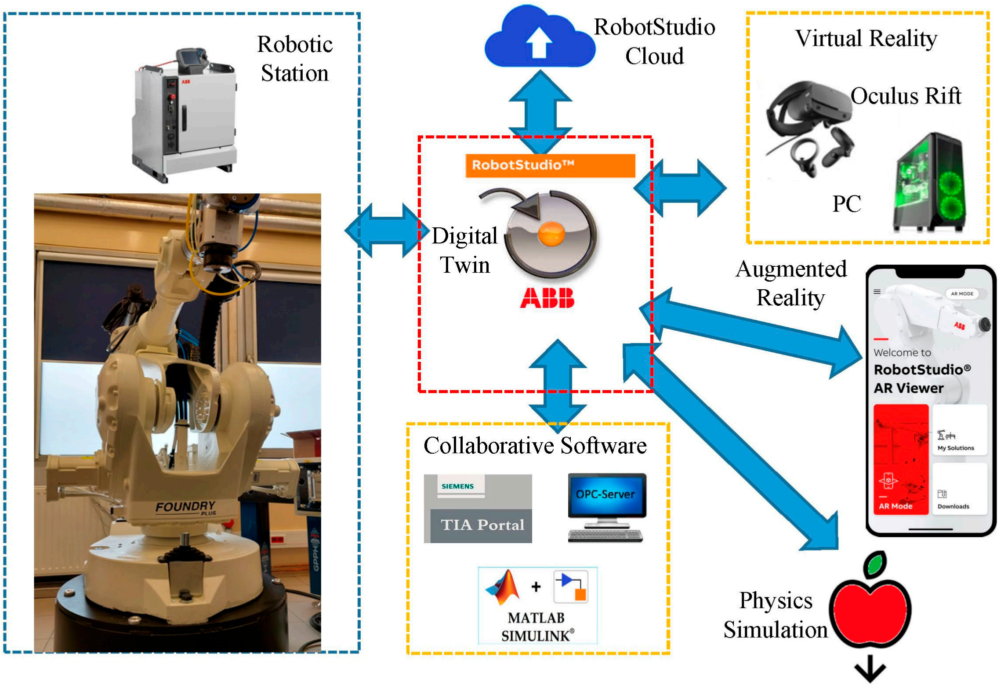


Figure 18. A diagram of the implementation of the concept of digital twins and Industry 4.0 solutions.

The solutions available for the presented software (Figure 18) allow not only the construction of digital twins but also the realization of other ideas of Industry 4.0. The used tool, in addition to communication with the real station and cooperating software, has the possibility of operating virtual reality with 3D glasses and augmented reality on smartphones. An example of the possibility of programming robots using virtual reality is shown in the work [2] and operator training in the work [24]. One of the features of the software is the so-called Virtual Meetings, i.e., a virtual meeting in a designed station of users from anywhere on earth. Station design and programming of robotic stations using virtual reality was shown in the work [34]. All the solutions mentioned were implemented using RobotStudio software and the concept shown in Figure 18. As for the use of cloud computing solutions, the available tool allows users from anywhere in the world to quickly access robot programs, make necessary changes and share data in real time. An example of the solution from Figure 18 using Matlab software and artificial neural networks is shown in the work [43,44]. In the case of classical 3D simulations and those using VR goggles, it is possible to simulate physical phenomena, e.g., items dropped by the robot fall and even bounce off obstacles.

5. An Example of Creating a Digital Twin of a Robotic Station Using a Laser Tracker

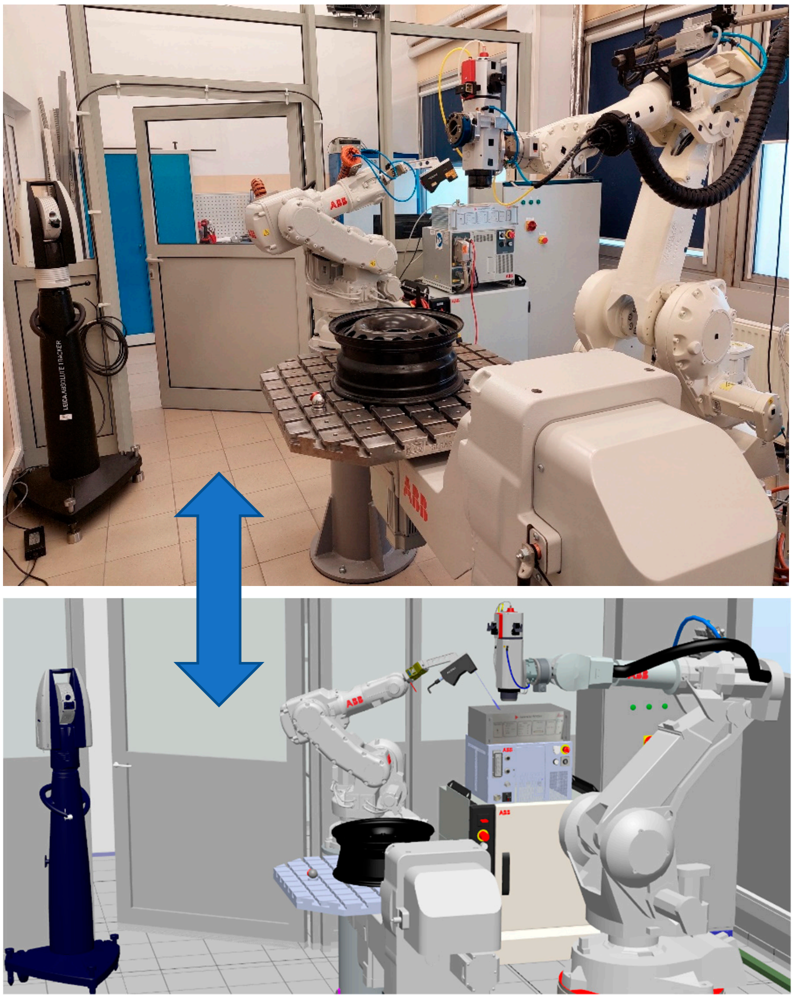
As an example for implementation, the variant shown in the diagram in Figure 3 was chosen, when the robotic station was physically built and its digital twin was needed. The real station is located in the Industrial Robotics Laboratory in the Department of Applied Mechanics and Robotics at the Faculty of Mechanical and Aeronautical Engineering at Rzeszow University of Technology.

The station was designed for the education of students in the field of Mechatronics and for research related to robotization for industries such as aerospace. A photo of the real station is shown in Figure 19. It is an advanced station consisting of two controllers, two robots (ABB IRB2400 and ABB IRB140) (Switzerland Zurich), a positioner, fences, an automation cabinet and various types of tools. According to the methodology shown in Figure 3, the position of the various components of the station was precisely determined using a Leica AT 960 absolute laser tracker (Wetzlar in Germany). The base frame of the IRB2400 robot was adopted as the assembly coordinate system. Relative to this system, the position of the other station elements was determined (Figures 20 and 21).

Figure 19. The robotic station that was used to create the digital twin.

(**a**) (**b**) (**c**)

Figure 20. Photos of station measurements using retroreflector: (**a**) pedestal position measurement; (**b**) floor position measurement; (**c**) controller position measurement.



(**a**)

(**b**)

Figure 21. Robotic station: (**a**) real station; (**b**) digital model of station in RobotStudio software.

According to the established methodology, the laser tracker data recalculated by the LeicaConnector software was transferred to CAD software. In this software, based on the received points and planes, the position of elements from libraries and available CAD models was defined. Precision models of the robot and the automation cabinet were added to the assembly for the reasons described in Section 3. The elements that would still need to be added to the model are the wires connecting the various components. The wires that run along the robot are particularly important. An accurate model of them could be used to avoid possible breakage during the robot's movement. They were partially modeled as shown in Figure 21b, but their models need to be extended.

The assembly was then exported to the off-line programming software, RobotStudio (Figure 21b). In order to show the similarity of the real station and its model, they are both presented in Figure 21.

In this software, as explained in Section 4, the digital model becomes a digital twin. Through the implementation of two-way communication based on synchronization, the exchange of programs between the digital model and the real station is realized. Programs are built both on the basis of CAD models and created by on-line programming methods. The accuracy of the digital model, obtained due to the laser tracker, allows precise generation of programs and determination of coordinate systems or tool positions. Communication using TCP/IP protocol is provided between the real station and the digital model. The role of the server is played by the real controller; the client is the virtual controller. Due to the continuous data exchange between the controllers, data such as those related to the operating parameters of the real robot are transmitted. The operating temperature of the gear, the cycle time or the currently used tool with its parameters are transmitted. The real controller is connected via the DeviceNet industrial network standard to the PLC and the frequency converters shown in the automation components cabinet (Figure 15). Due to this communication, data from these devices (start signal, errors occurring, set frequency of the inverter) can be transmitted to the digital twin. These data are presented on the digital model in the form of lit LEDs and virtual display indications. A solution based on the OPC standard is currently being implemented, where data from the real and virtual station will be sent to SIMATIC WinCC SCADA 7.5 software. The developed twin is used for education and training of students using augmented and virtual reality methods (Figure 22).

Na górze kontrolera znajduje się FlexPendant. Służy on m.in. do programowania robotów w trybie on-line. Po jego prawej stronie znajduje się przycisk bezpieczeństwa...

(**a**) (**b**)

Figure 22. The digital twin of the robotic station in training applications: (**a**) RobotStudio AR app on a smartphone with training on the station; (**b**) an excerpt of the training delivered using 3D glasses.

Students learn about the lab's structure and the role of its components using 3D glasses before they start working in the lab. While working in the lab, students have the opportunity to use augmented reality applications for smartphones. The RobotStudio AR Viewer app is available for free and allows for the station in question to show detailed information on, for example, the parameters of the robot or the cabinet with automation components.

6. Discussion

The developed methodology for the design of robotic stations has unquestionable advantages due to the solution with satisfactory accuracy of the problem of the limited degree of representation of the station geometry and details. It is helpful both in the design of robotic stations and their implementation and in the construction of a digital model based on the real station. It saves time associated with station programming due to the possibility of extensive use of the off-line programming method. Accurate digital models of robotic stations, as for industrial conditions, obtained through the implementation of the proposed methodology, can be used in training and various types of simulations in software such as Matlab or NVIDIA Isaac Sim.

The disadvantages of the developed solution include some inaccuracy in position determination due to the robot's accuracy error. The accuracy of the ABB robot using the AbsolutAccurency additive is about 0.38 mm. This is the value of the maximum error for the IRB2400 robot. A detailed explanation of the robot accuracy problem is presented in [10]. Using a robot that does not have this additive or a third-party solution can increase the position error up to 2–3 mm. The positioning error of the components is slightly affected by the accuracy of the tracker (these are of the order of 50 μm), which is very high compared to the accuracy of the robot. When positioning real station elements, the difference between the geometry of the real object and its 3D model can be a problem. In the real element, the walls may not be perfectly parallel or perpendicular to each other. In such a case, it is necessary to take the plane that is most important from the modeling point of view and correct in CAD software the alignment of the other walls.

Due to the problems mentioned above, the error of the method can be determined to be about 0.5 mm. A certain problem with the proposed method is the need for an absolute laser tracker, which is a very expensive device. Universities or large companies involved in the integration and design of robotic systems can afford to purchase it. Furthermore, the necessity of determining the robot's base frame before starting the work related to both the implementation of the designed station and the construction of the digital model can be also considered as a certain disadvantage of the method. As for the implementation of the concept of a digital twin, RobotStudio software has the capabilities described in Section 4. The author is not exactly familiar with the solutions of other robot manufacturers. However, software from Kuka and Fanuc Similar has capabilities related to advanced communication with other tools. Virtual and augmented reality applications are also being developed for software from these manufacturers.

7. Conclusions

This article deals with creating digital twins of robotic stations. They are created using modern tools and are used, among other things, for off-line robot programming. Analyzing the available publications on digital twins, it was found that in the case of robotic station twins, two-way data exchange and high model accuracy are crucial. In this paper, an attempt was made to solve the problem related to the limited accuracy of the mapping of the station and the details. According to the mentioned definitions, a digital twin should be characterized by the fidelity of the representation of the real object and the two-way data exchange shown in Figure 17 and discussed in Section 4. Creating a twin in virtual space using the attributes of a physical entity requires modeling geometric, physical and behavioral features. The degree of accuracy in modeling individual features depends on the available information about the object and the planned applications of the twin. During the design and programming of robotic stations, it was assumed that the physical features of the robot (its geometric and kinematic properties), as well as the geometric features of the workpiece and other important elements of the station (devices, fences, etc.), should be modeled. An accurate model of the robot's kinematics, dynamics, cycle time, etc., is obtained by using the virtual controller provided by the robot manufacturer (with 99% compliance guaranteed) and off-line simulation software. Therefore, the work carried out mainly concerns geometric mapping. The use of an absolute laser tracker was proposed as a solution. It is a device that provides high precision and long range. It should be mentioned that in the case of measuring robotic station components, the area sometimes reaches tens of square meters. However, having an accurate measurement system does not solve all the problems associated with station construction. Due to experience in station design and programming, a methodology related to the most common cases has been proposed. A scheme has been developed for the case where a station already exists and a digital model of it is built. The second scheme deals with the case when a digital model is first created and the real station is built on its basis. The advantages of the proposed approach are indicated, and based on other publications on the subject, the whole system based on accurate tracker measurements and advanced software is shown. Previously developed solutions together with the methodology proposed in the article complement each other and create a synergistic effect. The presented solutions allow us to design and implement robotic stations with a new quality. They make it possible to program industrial robots using a laser tracker and virtual reality. Analyzing the available literature, it was found that despite the fact that the topic of digital twins is taken up often, there are few realized examples of the implementation of this concept. It was decided to build a digital twin of an existing robotic station based on the developed methodology. A production twin of medium size was built. The tools in which it was realized and the available possibilities related to the realization of communication with various types of software were shown. The construction of the digital twin described in Section 5 took about 5 h. The most time-consuming tasks were unfolding the laser tracker and measuring the characteristic points of the station, such as the positions of the robots and the planes of the device walls (which took about 2 h). The method of determining the robot's position is detailed in the author's publication [10]. Due to the software used, the elements could be measured together within a common coordinate system. Another 2 h were spent modeling in the CAD system and positioning the created models relative to the points measured in reality. About 1 h was spent transferring the CAD model to the off-line programming tool and building a virtual controller based on the backup of the real station. We believe that the time required was relatively short, considering the complexity of the station (which includes two robots, a positioner, fences, a workpiece and equipment elements). Based on our experience with measurements using the tracker, we can state that factors such as noise and vibrations do not significantly impact the measurements. Due to the very high measurement frequency of 1000 Hz and the applied averaging, this type of interference does not affect the assumed accuracy of the geometric measurements. Due to the choice of an advanced tool for creating digital twins, it was possible to realize the latest solutions related to Industry 4.0 such as virtual and augmented reality.

Further research plans include integrating a tool for communication and operation of the laser tracker with RobotStudio software. This will allow real-time viewing of the retroreflector's position in a 3D environment. This will speed up the robot programming process and facilitate the assembly of components in the real station. As part of further work on the digital model of the developed robotic station, it is planned to import it into NVIDIA Isaac Sim software. It will make it possible to simulate artificial intelligence solutions and test safety solutions due to advanced models of people and their behavior.

Author Contributions: Conceptualization, D.S.; methodology, M.M. and A.B.; software, D.S., M.M. and P.P.; validation, A.B., K.K. and P.P.; formal analysis, D.S., M.M. and K.K.; investigation, D.S., M.M. and A.B.; resources, A.B. and P.P.; data curation, D.S. and K.K.; writing—original draft preparation, D.S., P.P. and A.B.; writing—review and editing, D.S. and P.P.; visualization, D.S., M.M. and K.K.; supervision, D.S. and A.B.; project administration, D.S. and A.B.; funding acquisition, A.B. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Data Availability Statement: The datasets used and analyzed during the current study are available from the corresponding author upon reasonable request.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Neto, P.; Mendes, N. Direct off-line robot programming via a common CAD package. *Robot. Auton. Syst.* **2013**, *61*, 896–910. [CrossRef]
2. Burghardt, A.; Szybicki, D.; Gierlak, P.; Kurc, K.; Pietruś, P.; Cygan, R. Programming of industrial robots using virtual reality and digital twins. *Appl. Sci.* **2020**, *10*, 486. [CrossRef]

3. 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]
4. 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]
5. Chryssolouris, G.; Alexopoulos, K.; Arkouli, Z. Artificial Intelligence in Manufacturing Equipment, Automation, and Robots. In *A Perspective on Artificial Intelligence in Manufacturing*; Springer International Publisher: Cham, Switzerland, 2013; Volume 436, pp. 41–78. [CrossRef]
6. Nutonen, K.; Kuts, V.; Otto, T. Industrial Robot Training in the Simulation Using the Machine Learning Agent. *Procedia Comput. Sci.* **2023**, *217*, 446–455. [CrossRef]
7. Kareemullah, H.; Najumnissa, D.; Shajahan, M.M.; Abhineshjayram, M.; Mohan, V.; Sheerin, S.A. Robotic Arm controlled using IoT application. *Comput. Electr. Eng.* **2023**, *105*. [CrossRef]
8. Mei, B.; Xie, F.; Liu, X.J.; Li, H. Calibration of a 6-DOF industrial robot considering the actual mechanical structures and CNC system. In Proceedings of the 2nd International Conference on Robotics and Automation Engineering (ICRAE), Shanghai, China, 29–31 December 2017; IEEE: Piscataway, NJ, USA, 2017; pp. 6–10. [CrossRef]
9. Novák, P.; Stoszek, Š.; Vyskočil, J. Calibrating industrial robots with absolute position tracking system. In Proceedings of the 2020 25th IEEE International Conference on Emerging Technologies and Factory Automation (ETFA), Vienna, Austria, 8–11 September 2020; IEEE: New York, NY, USA, 2020; Volume 1, pp. 1187–1190. [CrossRef]
10. Szybicki, D.; Obal, P.; Kurc, K.; Gierlak, P. Programming of industrial robots using a laser tracker. *Sensors* **2022**, *22*, 6464. [CrossRef]
11. Rajamurugu, N.; Karthik, M.K. Introduction, History, and Concept of Digital Twin. In *Digital Twin Technology: Fundamentals and Applications*; Wiley: Hoboken, NJ, USA, 2022; pp. 19–32. [CrossRef]
12. Lee, J.; Lapira, E.; Bagheri, B.; Kao, H.A. Recent advances and trends in predictive manufacturing systems in big data environment. *Manuf. Lett.* **2013**, *1*, 38–41. [CrossRef]
13. Gelernter, D. *Mirror worlds: Or the Day Software Puts the Universe in a Shoebox. . . How It Will Happen and What It Will Mean*; Oxford University Press: New York, NY, USA, 1993.
14. Segovia, M.; Garcia-Alfaro, J. Design, Modeling and Implementation of Digital Twins. *Sensors* **2022**, *22*, 5396. [CrossRef]
15. Fuller, A.; Fan, Z.; Day, C.; Barlow, C. Digital twin: Enabling technologies, challenges and open research. *Inst. Electr. Electron. Eng.* **2020**, *8*, 108952–108971. [CrossRef]
16. Mashaly, M. Connecting the twins: A review on Digital Twin technology & its networking requirements. *Procedia Comput. Sci.* **2021**, *184*, 299–305. [CrossRef]
17. Akroyd, J.; Harper, Z.; Soutar, D.; Farazi, F.; Bhave, A.; Mosbach, S.; Kraft, M. Universal digital twin: Land use. *Data-Centric Eng.* **2022**, *3*, 1–28. [CrossRef]
18. Petrik, D.; Herzwurm, G. iIoT ecosystem development through boundary resources: A Siemens MindSphere case study. In Proceedings of the 2nd ACM SIGSOFT International Workshop on Software-Intensive Business: Start-Ups, Platforms, and Ecosystems, ESEC/FSE'19: 27th ACM Joint European Software Engineering Conference and Symposium on the Foundations of Software Engineering (IWSiB 2019), Tallinn, Estonia, 26 August 2019; pp. 1–6. [CrossRef]
19. Narang, Y.; Sundaralingam, B.; Macklin, M.; Mousavian, A.; Fox, D. Sim-to-Real for Robotic Tactile Sensing via Physics-Based Simulation and Learned Latent Projections. In Proceedings of the 2021 IEEE International Conference on Robotics and Automation (ICRA), Xi'an, China, 30 May–5 June 2021; IEEE: New York, NY, USA, 2021; pp. 6444–6451. [CrossRef]
20. Niermann, D.; Doernbach, T.; Petzoldt, C.; Isken, M.; Freitag, M. Software framework concept with visual programming and digital twin for intuitive process creation with multiple robotic systems. *Robot. Comput. Integr. Manuf.* **2023**, *82*, 102536. [CrossRef]
21. Nie, Q.; Tang, D.; Liu, C.; Wang, L.; Song, J. A multi-agent and cloud-edge orchestration framework of digital twin for distributed production control. *Robot. Comput. Integr. Manuf.* **2023**, *82*, 102543. [CrossRef]
22. Kritzinger, W.; Karner, M.; Traar, G.; Henjes, J.; Sihn, W. Digital Twin in manufacturing: A categorical literature review and classification. *IFAC-PapersOnLine* **2018**, *51*, 1016–1022. [CrossRef]
23. Zhang, H.; Ma, L.; Sun, J.; Lin, H.; Thürer, M. Digital Twin in Services and Industrial Product Service Systems: Review and Analysis. *Procedia CIRP* **2019**, *83*, 57–60. [CrossRef]
24. Aras, A.; Ayaz, M.; Özdemir, E.; Abut, N. Virtual Reality Applications in Industrial Automation Systems: Industrial Robot Station Application. *Int. J. Comput. Exp. Sci. Eng.* **2020**, *6*, 189–194. [CrossRef]
25. 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]
26. 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]
27. Guo, D.; Zhong, R.Y.; Lin, P.; Lyu, Z.; Rong, Y.; Huang, G.Q. Digital twin-enabled Graduation Intelligent Manufacturing System for fixed-position assembly islands. *Robot. Comput. Integr. Manuf.* **2020**, *63*, 101917. [CrossRef]
28. Sun, X.; Bao, J.; Li, J.; Zhang, Y.; Liu, S.; Zhou, B. A digital twin-driven approach for the assembly-commissioning of high precision products. Robot. *Comput. Integr. Manuf.* **2020**, *61*, 101839. [CrossRef]
29. Zhang, X.; Yang, Y.; Zhang, X.; Hu, Y.; Wu, H.; Li, M.; Handroos, H.; Wang, H.; Wu, B. A multi-level digital twin construction method of assembly line based on hybrid worker digital twin models. *Adv. Eng. Inform.* **2024**, *62*, 102597. [CrossRef]

30. Zhang, X.; Wu, B.; Zhang, X.; Duan, J.; Wan, C.; Hu, Y. An effective MBSE approach for constructing industrial robot digital twin system. *Robot. Comput. Integr. Manuf.* **2023**, *80*, 102455. [CrossRef]
31. Nubiola, A.; Bonev, I.A. Absolute calibration of an ABB IRB 1600 robot using a laser tracker. *Robot. Comput. Integr. Manuf.* **2013**, *29*, 236–245. [CrossRef]
32. Al Khawli, T.; Anwar, M.; Sunda-Meya, A.; Islam, S. A calibration method for laser guided robotic manipulation for industrial automation. A calibration method for laser guided robotic manipulation for industrial automation. In Proceedings of the IECON 2018-44th Annual Conference of the IEEE Industrial Electronics Society, Washington, DC, USA, 21–23 October 2018; IEEE: New York, NY, USA, 2018; pp. 2489–2495. [CrossRef]
33. Schröer, K.; Bernhardt, R.; Albright, S.; Wörn, H.; Kyle, S.; van Albada, D.; Smyth, J.; Meyer, R. Calibration applied to quality control in robot production. *Control. Eng. Pract.* **1995**, *3*, 575–580. [CrossRef]
34. 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, Switzerland, 2019; Volume 568, pp. 585–593. [CrossRef]
35. Muszyńska, M.; Szybicki, D.; Gierlak, P.; Kurc, K.; Burghardt, A. The Use of VR to Analyze the Profitability of the Construction of a Robotized Station. *Adv. Manuf. Sci. Technol.* **2020**, *44*, 32–37. [CrossRef]
36. Robertson, B.F.; Radcliffe, D.F. Impact of CAD tools on creative problem solving in engineering design. *Comput. Aided Des.* **2009**, *41*, 136–146. [CrossRef]
37. Hunde, B.R.; Woldeyohannes, A.D. Future prospects of computer-aided design (CAD)—A review from the perspective of artificial intelligence (AI), extended reality, and 3D printing. *Results Eng.* **2022**, *14*, 100478. [CrossRef]
38. Szybicki, D.; Obal, P.; Penar, P.; Kurc, K.; Muszyńska, M.; Burghardt, A. Development of a Dedicated Application for Robots to Communicate with a Laser Tracker. *Electronics* **2022**, *11*, 3405. [CrossRef]
39. Kaknjo, A.; Rao, M.; Omerdic, E.; Robinson, L.; Toal, D.; Newe, T. Real-Time Video Latency Measurement between a Robot and Its Remote Control Station: Causes and Mitigation. *Wirel. Commun. Mob. Comput.* **2018**, *2018*, 8638019. [CrossRef]
40. Obal, P.; Burghardt, A.; Kurc, K.; Szybicki, D.; Gierlak, P. Monitoring the parameters of industrial robots. In *Methods and Techniques of Signal Processing in Physical Measurements*; Springer International Publisher: Berlin/Heidelberg, Germany, 2019; pp. 230–238. [CrossRef]
41. Khaled, T.A.; Akhrif, O.; Bonev, I.A. Dynamic path correction of an industrial robot using a distance sensor and an ADRC controller. *IEEE/ASME Trans. Mechatron.* **2020**, *26*, 1646–1656. [CrossRef]
42. Kihlman, H.; Loser, R.; Cooke, A.; Sunnanbo, A.; Von Arb, K. Metrology-integrated industrial robots: Calibration, implementation and testing. In Proceedings of the 35th ISR International Symposium on Robotics, Paris, France, 23–26 March 2004.
43. Liqiu, Z.; Juan, A.; Ronghao, Z.; Hairong, M. Trajectory planning and simulation of industrial robot based on MATLAB and RobotStudio. In Proceedings of the 2021 IEEE 4th International Conference on Electronics Technology (ICET), Chengdu, China, 7–10 May 2021; IEEE: New York, NY, USA, 2021; pp. 910–914. [CrossRef]
44. Hildenbrand, D.; Steinmetz, C.; Tichý, R. GAALOPWeb for MATLAB: An easy to handle solution for industrial geometric algebra implementations. *Adv. Appl. Clifford Algebras* **2020**, *30*, 1–18. [CrossRef]