ARTICLE IN PRESS

Measurement: Sensors xxx (xxxx) xxx

Contents lists available at ScienceDirect

Measurement: Sensors

journal homepage: www.sciencedirect.com/journal/measurement-sensors

Digital twin design of a 5 degrees of freedom industrial robotic arm for engineering education purposes

ARTICLE INFO

Keywords:
Digital twin
Photogrammetry
Robotics
Kinematics
Point cloud
Reverse engineering
Mitsubishi
Solid Edge
MATLAB

ABSTRACT

The main contribution of this paper is to transfer a Mitsubishi RV-2AJ MELFA type industrial robot with 5 degrees of freedom into a virtual space using photogrammetry. The mesh model of the robot was realised in Meshroom software while for the assembly model we used Sold Edge 3D designer software based on reverse engineering principles. The accuracy to the digital twin was tested by comparing the trajectory of the TCP with the physical robot. Moreover, forward kinematic calculations in MATLAB environment were used to validate the model on mathematical way. The digital twin has the opportunity to receive sensor data and coordinates from the physical robot and can be integrated into digital laboratories emphasizing and demonstrating industrial 4.0 technologies in modern engineering education.

1. Modern engineering education and industry 4.0

One of the biggest challenges of modern engineering education is to provide the right practical knowledge as well as theoretical knowledge. This is particularly true in the field of mechatronics engineering, which is a synergistic integration of mechanical, electrical, and informatical systems. The industry and market demand engineers who are familiar with the newest technologies and have strong practical skills. On the engineering university side, one of the conditions for ensuring these requirements is adequate laboratory facilities and sufficient time spent on practical tasks. Industrial development requires an increasing number of engineers, and universities are rapidly reaching the performance and capacity of their laboratories. This requires innovative thinking and the use of modern technologies presented by the Fourth Industrial Revolution.

Industry 4.0 is based on data and digitalization, cyber-physical systems, and digital twins. Moving real physical systems into virtual environment and using digital twins have many advantages such as real-time modeling and simulation of systems. This can minimize the risks and costs that real space represents. It allows systems to be designed, tested, and optimized more easily. The move to virtual space and the use of digital twins bring benefits not only for industry but also for education. The pandemic has also proved the huge need for virtual extension of university laboratories and virtualization of equipment. Digital twins of real devices can be part of the education as physical devices. The digital environment offers flexibility and provides an opportunity for engineering students to learn about the complexities of Industry 4.0 technologies gaining a complex understanding of modern manufacturing. However, it can only be effectively integrated into education if the similarity and functional accuracy of virtual twins meet the real devices.

This paper presents the realization of the digital twin of a Mitsubishi RV2-AJ MELFA industrial robotic arm and its validation based on the real equipment and kinematics. During the design process, a new technology called photogrammetry was used, which provided the possibility to create a highly accurate structural model of the robot. During the process, an appropriate number of images of the components with the right illumination, resolution, and quality were taken, from which a mesh cloud was generated. Then the 3D model of each component was realised by reverse engineering in Siemens Solid Edge 3D design software, thus the assembly model was created from the components, combining them into a complex machine. We tested the kinematic performance of the digital twin by comparing the Tool Center Point world coordinates with the forward kinematic model of the robot in a MATLAB environment. In addition to kinematics, the realistic 3D twin allows students to gain insight into the detailed structural design of industrial robots [1].

2. Digital twin technology

2.1. The general approach of digital twin

Nowadays, factories are first built in a virtual space, speeding up the design and then the construction process. The advantage of Digital Twin (DT) has made it possible to simulate entire production processes in real time. It processes and analyses information from the real world and provides real-time analysis. The concept was born in 2002 in the context of Product Lifecycle Management (PLM) by Michael Grieves at the University of Michigan. The model he proposed consists of three parts: real space, virtual space and the data/information flow medium that connects them. The model was then called the “Mirrored Spaces Model”. A similar model to the one mentioned above was already conceived in 1991 by David Gelernter, and described in the book “Mirror World”. In 2003, Kary Främling et al. came up with a similar idea, describing the relationship between the virtual equivalent of the process under

https://doi.org/10.1016/j.measen.2024.101323

investigation and the model that exists in reality, as a solution to increase the efficiency of paper-based transfer of manufacturing information to PLM. In 2006, the name of the conceptual model proposed by Grieves was changed from "Mirror Space Model" to "Information Mirroring Model". The model emphasized that the coupling between the two spaces should be bidirectional and that a model existing in real space should have several virtual counterparts, so that alternative ideas and designs could be explored simultaneously [2–4] (see Figs. 1).

Although both simulations and digital twins use digital models to reproduce different processes in the system, the digital twin is actually a virtual environment, which makes the study much richer. The difference between a digital twin and a simulation is largely a matter of scale: while a simulation typically studies a single process, a digital twin can itself run any number of useful simulations to study multiple processes. The differences do not end there. Simulations generally do not benefit from having real-time data. Digital twins, however, are designed around a two-way flow of information that occurs first when object sensors provide relevant data to the system processor, and then again when the insights generated by the processor are shared back with the original source object [5,6].

2.2. Classification based on complexity

There are many variations of digital twins, depending on how accurately we want to simulate a particular product. The biggest difference between them is the scope of application. It is common to have several coexisting but structurally different digital twins in a system [7,8].

- *Unit twins/component twins:* Unit twins are the basic unit of digital twins, these are the smallest building blocks. Component twins are roughly the same, but refer to slightly simpler, smaller devices. These can be motors, sensors, switches, valves.
- *Device twins:* When two or more component twins work together, they form a device. The device twins allow the interactions of these components to be studied, generating a wealth of performance data that can be processed and then converted into actionable insights.
- *System or unit twins:* It allows you to analyse how different devices fit together in a complete working system. System twins provide transparency into the interaction of assets and can suggest performance improvements.
- *Process twins:* Process twins are the largest, most complex twins that show how they work together to create a complete production facility. These systems are all synchronized with each other to operate at maximum efficiency.

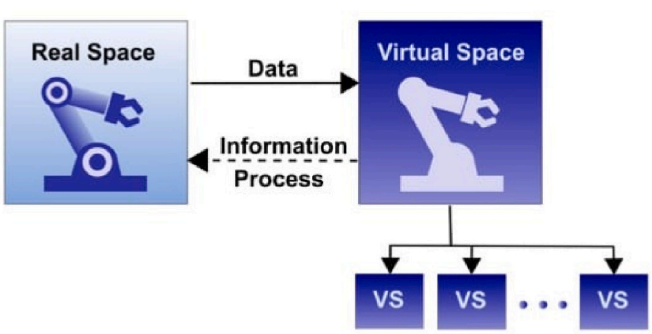

Fig. 1. Principle behind the Digital Twin technology [4].

3. Digital twin of Mitsubishi MELFA rv-2aj 5dof industrial robotic arm

3.1. Main specifications of the robot

The subject of the digitalization was a Mitsubishi MELFA RV-2AJ industrial robotic arm, that can be found at the Electrical and Mechatronics Department, Engineering Faculty, University of Debrecen, Hungary. The schematic and motion possibilities can be seen in Fig. 2:

It is a compact industrial robot that was introduced in 2002. The robot is equipped with 5 joint actuators and therefore has 5 degrees of freedom which are linear motion along the world coordinate system's *X*, *Y* and *Z* lines and rotation around *Z* and *Y* lines. The joints are numbered from *J1* to *J6*, but the robot does not have the *J4* joint, so the 5 degrees of freedom are given. The maximum payload of the robot is 2 [kg], which can only be used if the object has small dimensions. With a load of larger volume, the robot's payload is significantly reduced. This decrease is due to the fact that the larger the volume of the load, the more it affects the dynamic parameters of the effector [9].

The main specifications can be read in Table 1.:

3.2. Design using photogrammetry

The word of photogrammetry consists of 3 parts, which are: "photo", "gram" and "metry". Photogrammetry is the science of remote sensing, whereby photographs of objects or terrain/environment are used to determine the extent of real objects in the images by measurements and calculations. The basic principle of photogrammetry is that the same object can be reconstructed using photographs taken from several different angles, and the geometric information between the photographs can be used to determine the three-dimensional properties of the object. Photogrammetry is becoming more and more common nowadays, and open source software are available to generate three-dimensional models from photographs [10] (see Fig. 3).

To model the robotic arm as accurately as possible, we used photogrammetry. We divided the robot into its component parts and created a separate three-dimensional model of each segment. For this, an open source program was used called Meshroom, developed by AliceVision. Meshroom's interface is node base, which makes it user-friendly and gives you full control over the settings. In order to render the three-dimensional model with the right quality, several important rules

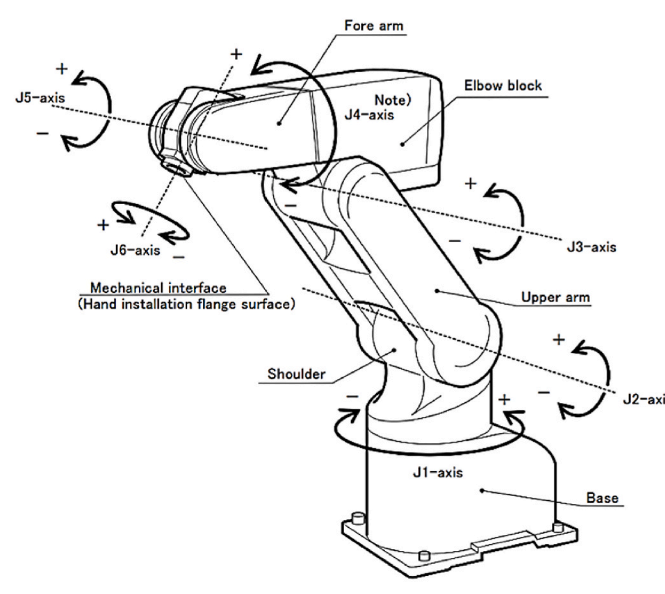

Fig. 2. The motion possibilities of the industrial robotic arm [9].

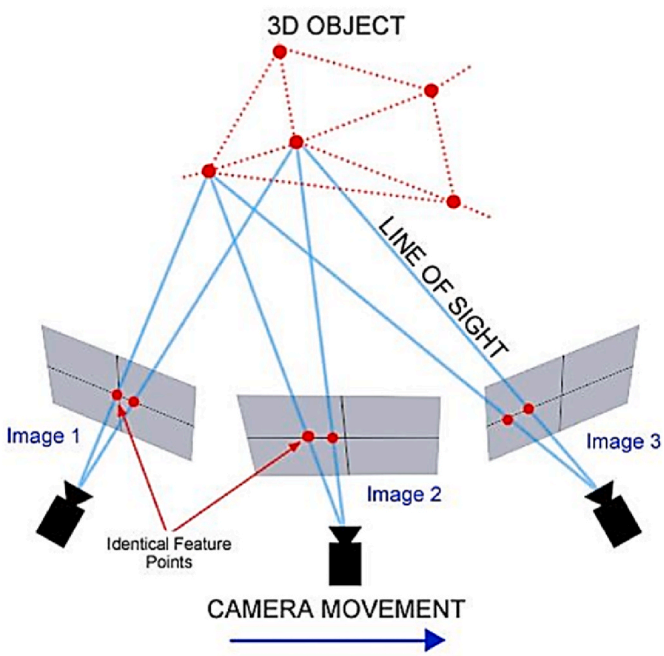

Fig. 3. Projection process in photogrammetry [10].

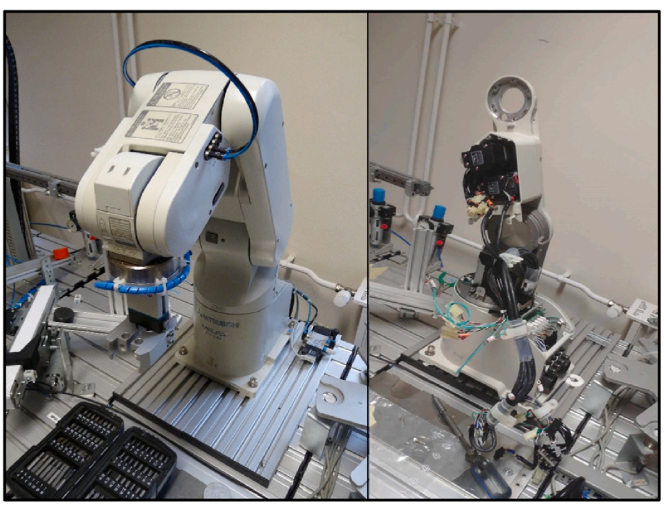

Fig. 4. The robot under disassembling process [11].

Table 1
RV-2AJ specifications [9].

Maximum load weight	2 [kg]
Maximum reach distance	410 [mm]
Repeatability	±0.02 [mm]
Type of the controller	MELFA CR1-571
Maximum speed	2000 [°/s]
Power supply requirements	180-253 [V] (AC)

must be followed:

- Accurate photogrammetry requires a large number of images, depending on the size and complexity of it. For our purpose, we needed to take approximately 80–150 images of a given robotic component.
- It is important that photos are clear, with no blurring, low noise and sufficient depth of field.
- Lighting should be constant, without shadows.
- The subject must not move when taking photos.
- At the beginning of taking a photo, it is necessary to decide whether lying or still images are required.
- The focal length should not be changed between photos [11–13].

To take the photos, a professional camera was used, which is a Panasonic Lumix DMC-FZ28. It has a 1/2.33″ CCD with a 10.7 Megapixel sensor. The effective resolution is 10.1 Megapixels, allowing the camera to capture 4K images. The best 3D scan can be achieved when everything in the photo is in contrast and clearly visible. The aperture matters a lot in this question, since the wider the aperture, the smaller the value gets, so the depth of field is reduced and only a certain part of the image will be sharp. For this reason a large aperture was set on the camera which is Panasonic Lumix DMC-FZ28 with the following settings: f/2.8 - f/4.2 closed: f/8 - f/8. The shutter speed controls how long the camera will expose. As the shutter speed is increased, the camera takes a longer exposure time and the camera has to be held still, which is hardly possible in many cases. ISO is the sensitivity of the sensor. The higher the value, the more sensitive it is to the light it is exposed to. The higher the ISO value, i.e. the higher the light sensitivity, the grainier and noisier the image will be. Finally, the applied setups were the following: *f-stop f/8, exposure time 1/40 [s], sensitivity ISO-200 and focal length 11 [mm]* [14] (see Fig. 4).

Once the photos were taken, we imported them into Meshroom. We took roughly 300–500 images of the wrist segments and the processing time for these was approximately 2.5 hours.

4. Extracting the information and geometry from images

4.1. Mesh generating from images using Meshroom

For generating the mesh, an open source software was used, called Meshroom. It may seem complicated at first look, until the user encounter a node base system (such a system is followed by Blender and other 3D drawing programs):

- Image Gallery, where images can be imported using drag & drop method,
- 3D Viewer and the display settings,
- Graph Editor, which is based on the Node base system mentioned above.

After importing the images, the software generated a preliminary point cloud where we see how accurately the mesh can be realised from the imported images. The point cloud was suitable, so starting the mesh generation was the next step. During the process, using *Bounding Box*, we created ROI (Region Of Interest) for the software in order to specify the most important segments of the image. In Fig. 5.: an accurate point cloud and mash can be seen [14].

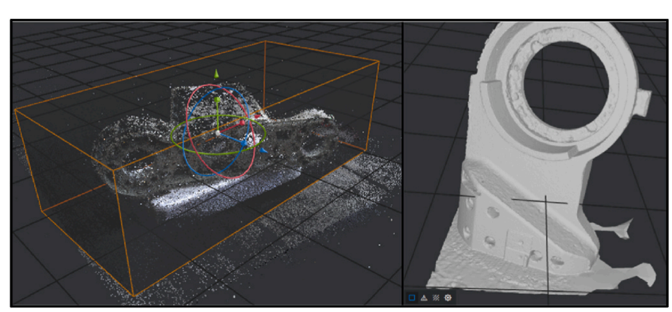

Fig. 5. An accurate point cloud and mesh generation [14].

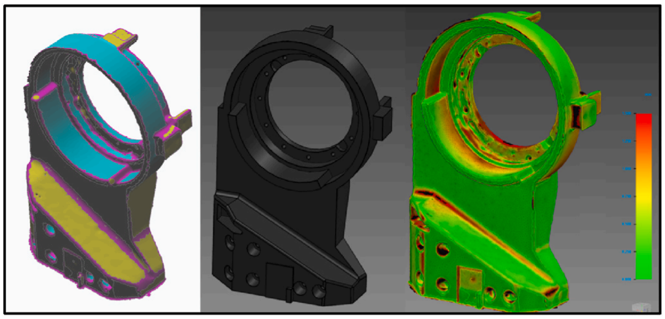

Fig. 6. The different stages of the 3D model of J2 joint [14].

4.2. Natural feature extraction, image- and feature matching

The purpose of the *Natural Feature Extraction* step is to extract distinctive sets of pixels that are to some extent invariant to changes in camera viewpoints during image capture. Therefore, an element of the image should have similar feature descriptions across all images. The best known feature detection method is the SIFT (Scale-Invariant Feature Transform) algorithm. The initial goal of SIFT is to extract distinctive patches from the first image that can be compared with distinctive patches from a second image, regardless of rotation, translation and scale. Since a relevant detail exists only at a certain scale, the extracted patches are centred on stable points of interest. To a certain extent, SIFT invariance can be used to handle image transformations that occur when viewpoints change during image acquisition. SIFT computes the scale space maximum of the flat representation, which is a specific image energy-based representation of the image, using so-called Gaussian differences. These maxima correspond to corner points. Each keypoint is associated with a descriptor of these corrections. The SIFT algorithm then performs the generation of the descriptor code corresponding to the corner points found. It assigns to each corner point, a 16 × 16 pixel environment, so that the descriptor size is a code of 128 digits. This 16 × 16 pixel area is always extracted from the resized image, thus ensuring the scalar invariance of the descriptor [14].

In the *Image Matching* step, the program searches for images that overlook the same areas of the photos. To do this, an image search technique is used that finds images that match in some content, without examining all the feature matches. The goal is to simplify the image into a compact image descriptor that allows efficient search across all image descriptors. One of the most widely used image descriptors is the so-called vocabulary tree topology. All the extracted feature descriptors are passed to the vocabulary tree, which performs classification by comparing their features with the features at each node of the tree. Each feature descriptor is stored in a leaf, which can be stored with a simple index [14].

During the *Feature Matching* the program searches for matches between the set of descriptors of the two input images. For each descriptor of image *A*, a list of candidate descriptors of image B is obtained. The descriptor space is not a well-defined linear space, so we cannot rely on absolute distance values to know whether a match is valid or not. Finding the two closest descriptors of the second image requires a large computational effort, and there are several algorithms to speed this up. The most commonly used algorithms are Approximate Nearest Neighbour and Cascading Hashing [14].

4.3. Structure from motion, depth maps estimation and Meshing

The purpose of the *Structure from Motion* step is to understand the geometric relationship behind all the observations provided by the input images. Its task is also to infer the rigid scene structure using the position, orientation and internal calibration of all the cameras. This is an incremental process, whereby it first computes a reconstruction consisting of two views, and then iteratively extends it by adding new views. As a first step, all the matches between pairs of images are merged. During the matching process, inconsistent tracks are removed and then the best initial image pair is selected. It is important to provide truly reliable geometric information, robust matches, and maximize the number of matches between image pairs. Meanwhile, to obtain reliable geometric information, the angle between the cameras must be large enough. Knowing the correct 2D features and the pose of the two front cameras, we can merge the features into 3D points using triangle method.

In cases of the camera images that *Structure from Motion* has processed, it is important to get the depth value for each pixel. There are several methods to do this, such as block matching. The algorithm consists of taking one environment of the pixel under consideration and moving it through all possible positions of the corresponding row of the other image, comparing at each step the environment of the original pixel with the environment of the current position. For each image. Since we know the transformation between each camera, we transform all camera positions into a common coordinate system. In this common frame of reference, the projection rays for the same spatial point will intersect at the position of the original spatial point. For three-dimensional points, there is a small probability that the rays from the two cameras will avoid each other during triangulation, so no intersection is obtained. It is therefore recommended to use filtering.

Finally, in the *Meshing* step, the elements of the created depth map are brought into a square grid layout and then convolution is performed. This involves finding the nearest neighbour in each direction and then considering this as the immediate neighbour on the grid. This is called χ-convolution. The convolution can be further modified here to take into account not only the values of the neighbouring points/features but also their distance. This solution is used in the so-called Point-CNN architecture [14].

5. Realising the 3d model of the robotic arm

5.1. Align Mesh to the world coordinate system, identify region and remesh procedure

Mesh in the STL file format for the joints can be easily imported into Solid Edge. Once imported, the Mesh will be visible in the 3D design window as a Construction Body. This body is not mapped to the Origo, thus manual position is necessary. Accordingly, there are several ways to move, such as: *use principal axes*, *use best-fit bounding box*, *use custom bounding box* and *use geometry references*.

Due to the large number of polygons, remesh is required, otherwise the computation capacity is extremely high. In order to reduce it, set the following properties:

- Target size: This defines the average size of the polygons on the selected body.
- Minimum size: The size of our smallest polygon. These are mainly found on curves and plane boundaries.
- Chordal Tolerance: How much the polygon should follow the arc.

In our case after specifying the target size, the minimum size was set to 50 % the target size, while the chordal tolerance was set to 10 % of the target size. Finally, colours were assigned to simple geometric shapes, which can be used to model any solid object. The colours are the following [14] (see Fig. 6):

- Lemon yellow: Plane
- Light blue: Cylinder
- Dark blue: Cone
- Red: Sphere
- Purple: It separates the previously listed simple geometric shapes from each other.

- White: Regions that will not be taken into consideration by the *Extract surfaces* operation.

Fig. 7: shows the stages of the procedure which consists of the simple shape association, the final 3D model and its comparison with the original mesh in order to see how accurately the 3D model generation was performed. From colour red to green the match is higher.

The parts that did not require photogrammetry were modelled in the conventional way. These parts were the harmonic drives between the joints, the motors and the gears on them. The assembly of the robotic arm was quick and easy, and we didn't encounter any faults in the component connections. This means that the three-dimensional parts created using photogrammetry were created exactly to the dimensions of the original part.

6. Accurcy test of the model

Since the industrial robots are precisely moving mechanism, one of the most important properties and requirement is the accuracy. Accuracy of an industrial robot can be affected by many factors. Estimating the accuracy is one of the tasks of kinematic analysis. Our digital twin can represent the real robot only if the kinematic calculations provides the same results as the real robot has.

6.1. *The kinematic representation of the robot*

The kinematic representation of the robot defines the motion possibilities based on the geometry of the mechanism. The kinematic calculations are generally conversion between the joint coordinates and world coordinates of the robot. Joint coordinates refer to the actual value of each of the joints (angle or linear displacement), however, world coordinates define the position and orientation of the end effector. When joint coordinates are inputs and world coordinates are outputs of the calculation, forward kinematic is used, otherwise inverse kinematic. The kinematic chain the Mitsubishi RV-2Aj robot is defined by the Denavit-Hartenberg parameters, shown in Table 2:

During the test we realised a motion sequence using the physical robot, then the same sequence was simulated using the 3D model, without direct connection to the robot. Finally, we used to forward kinematics representation of the robot in order to validate accuracy in mathematical way using MATLAB software environment. Our expectation was that the difference must be around the repeatability, since minor inaccuracies can be measured because of the aging of the physical components.

6.2. *Trajectory analysis of the Tool Center Point*

The TCP of the real and virtual robots is configured to the adapter on the J6 wrist. In the virtual space, it is possible to have a point in the space traced by the program and then visualized at the end of the simulation. This function is called *Trace Path*. The three-dimensional path generated by the *Trace Path* function can be saved in.csv file format (see Fig. 8).

To make sure that the robot accurately tracks our physical robot at every position, we simulated the motion on the real robot and recorded the TCP coordinates every 10°. First, the start and end positions between the two measurements were examined. For the real robot, we were able to adjust the articulations to within 0.01° (see Table 3).

7. Conclusions

It can be seen that the difference between the initial X and Y values is negligibly small, and our Z height is only very slightly off. Similarly expected results were obtained at the end point. To investigate the path of the TCP, we not only compared the measured positions on the physical robot, but also wanted to analyse the robot's motion mathematically. For this, we used a code written in MATLAB environment. The code represents the forward kinematics to calculate the TCP position [1]. To do this, the rotation of the joints in radians is given, and from this data the TCP can be determined(see Figs. 9).

The red continuous line is the result of a series of measurements provided by Solid Edge. The continuity is due to more than 400 measurement points. The circle in green is the results read from the robot,

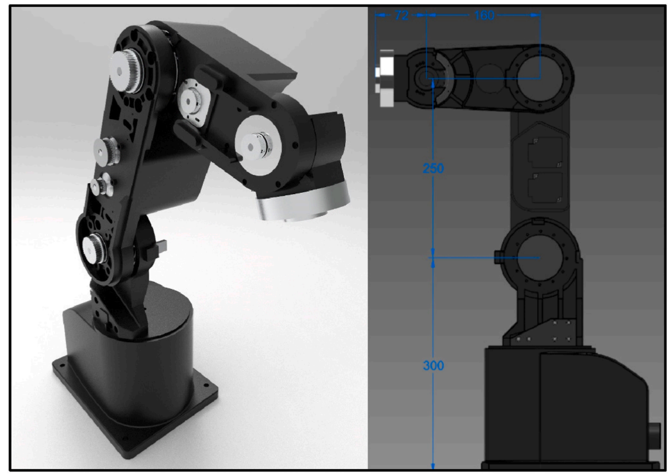

Fig. 7. The complete assembly model of the robot [14].

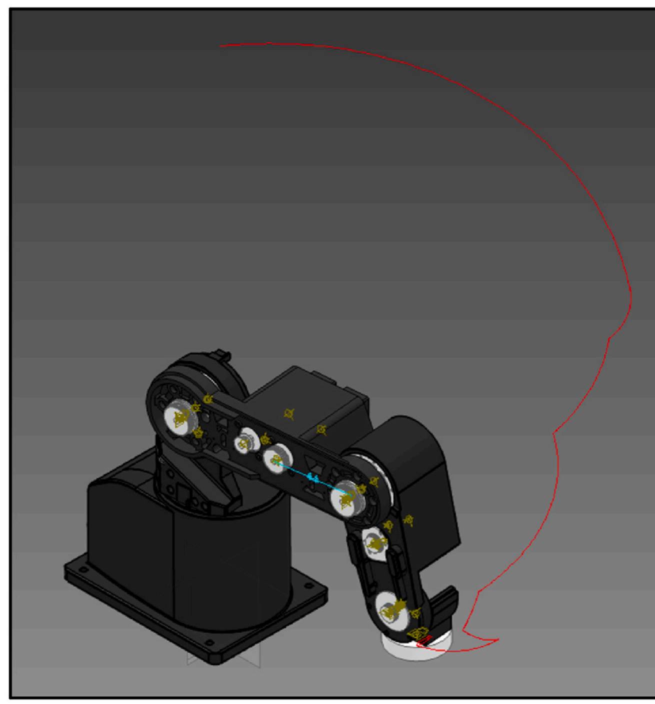

Fig. 8. The simulated trajectory of the TCP [14].

Table 2
The obtained Denavit-Hartenberg parameters [14].

Joint	θ_i [rad]	d_i [mm]	a_i [mm]	α_i [rad]	Offset
1	θ_1	300	0	$-\frac{\pi}{2}$	0
2	θ_2	0	250	0	$-\frac{\pi}{2}$
3	θ_3	0	160	0	0
5	θ_4	0	0	$-\frac{\pi}{2}$	$-\frac{\pi}{2}$
6	θ_6	178.5	0	0	π

Table 3
The obtained Denavit-Hartenberg parameters [14].

Position	X (mm)	Y (mm)	Z (mm)
Robot initial pos.	0,18	0	782
Model initial pos.	5,73E-06	−6,09E-12	782,0083
Difference	≅ 0	≅ 0	−0,0083
Robot end pos.	290,24	25,43	73,41
Model end pos.	290,2523	25,3961	73,4555
Difference	−0,0132	0,0339	−0,0455

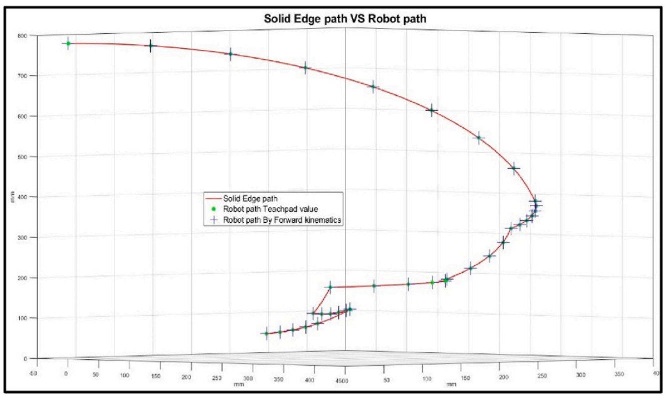

Fig. 9. The results from comparison of the robot, simulation and kinematics TCP [14].

while the cross in blue is the coordinates from the MATLAB code. It can be clearly seen that the plots are accurate and consistent, as the measurement points follow the red TCP path accurately [14].

The twin can be integrated into the virtual laboratory and, with appropriate software support, allows interactive learning of robot programming. The structure has the same weight as the real robot, so its motion performance and power consumption can be calculated and monitored. Using generative design, engineering students can learn optimization principles.

References

[1] C. Bai, P. Dallasega, G. Orzes, és J. Sarkis, Industry 4.0 technologies assessment: a sustainability perspective, Int. J. Prod. Econ. 229 (2020) 107776, https://doi.org/10.1016/j.ijpe.2020.107776.

[2] M.W. Grieves, Product lifecycle management: the new paradigm for enterprises, Int. J. Prod. Dev. 2 (1–2) (2005) 71–84, https://doi.org/10.1504/ijpd.2005.006669.

[3] M.W. Grieves, Product lifecycle management: the new paradigm for enterprises, Int. J. Prod. Dev. 2 (1–2) (2005) 71–84, https://doi.org/10.1504/ijpd.2005.006669.

[4] M. Singh, E. Fuenmayor, E.P. Hinchy, Y. Qiao, N. Murray, D. Devine, Digital twin: origin to future, Appl. Sys. Innov. 4 (2) (2021), https://doi.org/10.3390/asi4020036. MDPI AG.

[5] K. Främling, J. Holmström, T. Ala-Risku, M. Kärkkäinen, in: Product agents for handling information about physical objects, 2003.

[6] Michael Grieves, in: Back to the future: product lifecycle management and the virtualization of product information, 2009.

[7] A. Agrawal, M. Fischer, V. Singh, Digital twin: from concept to practice, J. Manag. Eng. 38 (3) (May. 2022), https://doi.org/10.1061/(asce)me.1943-5479.0001034.

[8] M. Liu, S. Fang, H. Dong, C. Xu, Review of digital twin about concepts, technologies, and industrial applications, J. Manuf. Syst. 58 (2021) 346–361, https://doi.org/10.1016/j.jmsy.2020.06.017.

[9] Mee, RV-1A/2AJ series instruction manual robot arm setup & maintenance".

[10] AliceVision, Photogrammetry pipeline natural feature extraction. Accessed: 2023. November 3. [Online]. Available: https://alicevision.org/#photogrammetry/natural_feature_extraction.

[11] Sampsa Kohtala, Jørgen F. Erichsen, Ole Petter Wullum, Martin Steinert, Photogrammetry-based 3D scanning for supporting design activities and testing in early stage product development, Procedia CIRP 100 (2021) 762–767, https://doi.org/10.1016/j.procir.2021.05.047. ISSN 2212-8271, https://www.sciencedirect.com/science/article/pii/S2212827121005096.

[12] Integrating 3D Modeling, Photogrammetry and Design, 2014, 978-4471-6328-2.

[13] D.W. James, F. Belblidia, J.E. Eckermann, J. Sienz, An innovative photogrammetry color segmentation based technique as an alternative approach to 3D scanning for reverse engineering design, Comput. Aided Des. Applica. 14 (1) (2017) 1–16, https://doi.org/10.1080/16864360.2016.1199751.

[14] T. Vasváry, Design of Mitsubishi RV-2AJ robot "digital twin", in: Solid Edge and Kinematic Matching, University of Debrecen, Mechatronics Department, 2023 (in Hungarian).

Gyula Korsoveczki[a,*,1], Tamás Vasváry[a,1], Angéla Váradiné Szarka[b,2], Kornél Sarvajcz[a,1]

[a] *University of Debrecen, Faculty of Engineering, Electronics and Mechatronics Department, Debrecen, Hungary*

[b] *Bay-Energy Energetic Division, Bay Zoltán Applied Research non-profit Ltd., Budapest, Hungary*

[*] Corresponding author.

E-mail address: korsoveczki.gyula@gmail.com (G. Korsoveczki).

[1] www.mechatronics.unideb.hu.

[2] www.bayzoltan.hu.