International Journal of
INTELLIGENT SYSTEMS AND APPLICATIONS IN ENGINEERING
ISSN:2147-6799 www.ijisae.org Original Research Paper

Modeling Robotic Arm with Six-Degree-of-Freedom Through Forward Kinematics Calculation Based on Deep Learning

Thanh Q. Nguyen[1], Khien B. Pham[2], Duong Thi Kim Chi[3]

Submitted: 11/11/2022 **Accepted:** 13/02/2023

Abstract: Modeling a robotic arm is one of the popular types of CNC (computer numerical controller) machines that are suitable for specialized training and meeting the high demand in today's manufacturing industry. However, research and development of robotic arm models in Vietnam are still limited and primarily concentrated in large foreign-invested factories. This research develops a forward kinematics problem model for a six-degree-of-freedom robotic arm, which is a common type of model in the industry today, using artificial intelligence (AI). This study details each step, from axis transformations, translations, and rotations to determine the position of each link at various times, based on deep learning. It establishes the relationship between each step of the robot designed from the virtual model by AI. Furthermore, the study will use calculations and simulations to compare and contrast the deviations and verify the results. In the future, the study will incorporate inverse kinematics and dynamics problems to create a comprehensive study of the six-degree-of-freedom robotic arm model.

Keywords: *Modeling robotic arm, robots with six-degree-of-freedom, kinematics, forward kinematics, reverse kinematics, artificial intelligence, deep learning*

1. Introduction

With the advancement of science and technology, the robotic arm model has always had fast, accurate, stable, and particularly time-saving operations when compared to work done by human hands. The robot's basic movements typically include rotation, translation, and parallelism in order to perform some basic operations such as holding, grasping, lifting, shifting, turning, and flipping [1] [2] [3] [4]. Through these movements, the robotic arm model can perform welding and assembly operations throug human control. The robotic arm model is frequently programmed and simulated in the virtual space of specialized software to save time during evaluation, fabrication, or trial production [5] [6] [7]. They can simulate the model's actual operation to anticipate the risks of collision, overload, or irregularities in motion. As a result, every actual movement of the robot during the working process will be precise and smooth, with no redundant movements. The movements of a robotic arm modeling are always the same and repeat exactly the same for the given cycles. As a result, these movements produce the exact results that each product requires, thereby reducing the number of products that are damaged, damaged as a result of carelessness or mishandling during the same operation as in humans, especially in jobs requiring a high level of concentration and high accuracy. As previously discussed, the robotic arm model is always monitored and operated correctly, it is resulting in high productivity from the industrial robot arm. They are especially capable of working around the clock (24/7) and bringing the highest productivity to businesses. Investors will cut many costs incurred during the manufacturing process, particularly for toxic, dangerous, and risky working environments such as dust, heat, radiation... As a result, the main robot is an ideal substitute for humans. Robots have helped us reduce a significant number of workplace accidents over the years. With the objective as analyzed above, those studies calculated the kinematics for the robotic arm model based on deep learning, which serves the assembly on automatic lines via a computational model based on the angular coordinate system. Thanks to the artificial intelligence model, robotic systems have been operated faster, more accurately and with a high degree of automation in operation, which can be applied in industrial environments [8] [9] [10] [11] [12]. Currently, the applications of machine learning models to robot operating systems are increasing and it is changing the image of factories and industrial parks around the world.

[1]*Faculty of Engineering and Technology, Thu Dau Mot University, Binh Duong Province, Vietnam*
ORCID ID : 0000-0003-4898-091X
[2]*HUTECH Institute of Engineering, Ho Chi Minh City University of Technology, Ho Chi Minh City, Vietnam*
ORCID ID : 0000-0001-9120-0523
[3]*Faculty of Engineering and Technology, Thu Dau Mot University, Binh Duong Province, Vietnam*
ORCID ID : 0000-0003-1744-3249
** Corresponding Author Email: [2]pb.khien@hutech.edu.vn ; [3]chidtk@tdmu.edu.vn*

Fig. 1. The Denso VS-6556 robot. There are six-degree-of-freedom

The angular coordinate system, also known as the simulated coordinate system, is used to calculate the robot [13] [14] [15]. This is the coordinate system used in many current studies because of its ability to calculate accurately, simply, and efficiently. Rotation angles perform the three basic movements [16] [17] [18] [19]. The basic movements are determined by six rotation angles in the angular coordinate system. Because all of the robotic arm parts in this study are on the vertical plane, the basic calculations are all planar. When this plane rotates, this robotic arm model can be placed anywhere in the work area where human control is required [20] [21]. The compact model size and smooth control of this method for robots means that the operation area is relatively large and optimal compared to the size of the robot itself. This study discovered through a field survey that in order to improve the flexibility of the robotic arm model used in industry, manipulators must have a high number of degrees of freedom of movement. This demonstrates that the robotic arm model with six-degree-of-freedom efficiency improves in the industrial use process. However, the number of degrees of freedom in this robotic arm model should not exceed six due to the complexity of the control process. This can be shown with six-degree-of-freedom of movement, if properly arranged, it will be sufficient to create the flexibility of the final acting stage to be able to reach the manipulated object (located in its action) in all directions. The kinematic diagram of robotic arm model with six-degree-of-freedom obtained using the Denso VS-6556 robot model as shown in Figure 1 is as follows in Figure 2:

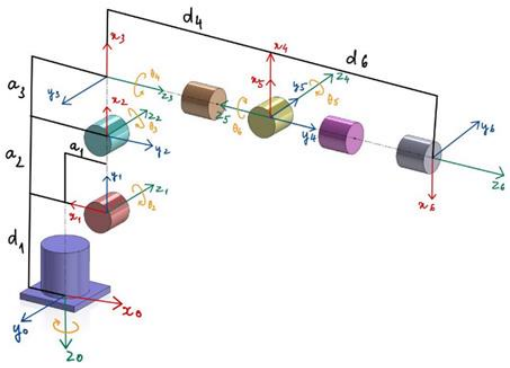

Fig. 2. Kinematic diagram of the robot manipulator and DH coordinate systems for each stage.

This research, based on artificial intelligence model, the manuscript calculates a forward kinematics problem model for a six-degree-of-freedom robotic arm model. The results obtained from the manuscript will solve many problems in controlling six -degree-of-freedom robot systems. This study details each step, from axis transformations, translations, and rotations to determine the position of each link at various times based on deep learning. It is done explicitly in establishing the relationship between each step of the robot that was designed from the virtual model by AI. Furthermore, the study will use calculations and simulations to compare and contrast the deviations and to verify the results. In the future, the study will incorporate inverse kinematics and dynamics problems in order to create a comprehensive study for the six-degree-of-freedom robotic arm model.

2. Computational Model and Results Obtained

2.1. Kinetic parameters Denavit – Hartenberg [22]:

Stages will be n+1 on a robot with n joints. Thus, if the number of stages begins at 0 for the base stages (the model's pedestal) that are not moving and gradually increases up to n for the stages in the most recently manipulated position [23] [24] [25]. Number the joints from 1 to n, beginning with the joints associated with the first movable stages to the base. Thus, joint i is connected to its lower suture (i-1), on its proximal side by joint i and joint (i+1) is connected to its higher suture (i+1) on its distal end.

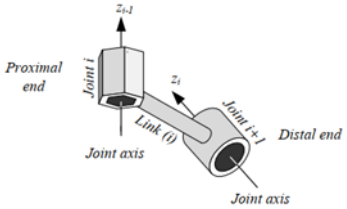

Fig. 3. Stage (i) and end joints (i-1) and (i).

- The z_i axis aligns with the joint axis $i+1$
- The x_i axis is defined along the common normal between the z_{i-1} and z_i axis, in the direction from z_{i-1} to z_i
- The y_i axis is determined by the right hand rule with $y_i = z_i \times x_i$
- The DH coordinate frame is defined by a_i, α_i, θ_i và d_i

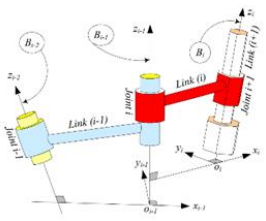

Fig. 4. Stage (i-1), i, (i+1) together with B_i and B_{i+1}, coordinate frames

The stage length a_i is the distance along the x_i axis between the z_{i-1} and z_i axes. The kinetic length of the bond i is then given by a_i. Furthermore, the bond i is the rotation of the z_{i-1} axis around the x_i axis required to become parallel to the z_i axis. The d_i joint distance is equal to the distance between x_{i-1} and x_i along the z_{i-1} axis. Joint displacement is also known as distance between joints, where joint angle i is the required rotation of the x_{i-1} axis around the z_{i-1} axis to become parallel to the x_i axis.

2.2. Setting the kinetic parameters Denavit - Hartenberg

The study will establish a table of DH parameters through table 1

Table 1. Denavit – Hartenberg table of kinematic parameters of the robot

Elements		d_1	a_i	α_i
1	θ_1	335	75	-90°
2	θ_2	0	270	0
3	θ_3 - 90^0	0	90	90°
4	θ_4	-295	0	-90°
5	θ_5	0	0	90°
6	$\theta_6 + 180^0$	-80	0	90°

Where the variables of the match are $\theta_1,\theta_2,\theta_3,\theta_4,\theta_5,\theta_6$, and the constants are d_1,d_4,d_6,a_2. Set the rotation angles at the joint variables to $q_1 = \theta_1$, $q_2 = \theta_2$, $q_3 = \theta_3$, $q_4 = \theta_4$, $q_5 = \theta_5$, $q_6 = \theta_6$, and , $q_6 = \theta_6$ for ease of calculation and programming. The Denavit - Hartenberg uniform coordinate transformation matrix in its general form for the following stages:

$$^{i-1}\mathrm{T_i} = \begin{bmatrix} \cos q_i & -\sin q_i \cdot \cos \alpha_i & \sin q_i \cdot \sin \alpha_i & a_i \cdot \cos q_i \\ \sin q_i & \cos q_i \cdot \cos \alpha_i & -\cos q_i \cdot \sin \alpha_i & a_i \cdot \sin q_i \\ 0 & \sin \alpha_i & \cos \alpha_i & d_i \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (1)$$

The matrix $^{i-1}T_i$ is divided into two sub-matrixes, and the research presents a unique rotation and translation to generate the fixed motion required to move from stage *i-1* to *i*.

$$^{i-1}T_i = \begin{bmatrix} ^{i-1}R_i & ^{i-1}d_i \\ 0 & 1 \end{bmatrix} \quad (2)$$

with

$$^{i-1}R_i = \begin{bmatrix} cos\ q_i & -sin\ q_i \cdot cos\ \alpha_i & sin\ q_i \cdot sin\ \alpha_i \\ sin\ q_i & cos\ q_i \cdot cos\ \alpha_i & -cos\ q_i \cdot sin\ \alpha_i \\ 0 & sin\ \alpha_i & cos\ \alpha_i \end{bmatrix} \quad (3)$$

and

$$^{i-1}d_i = \begin{bmatrix} a_i \cdot cos\ p_i \\ a_i \cdot sin\ p_i \\ d_i \end{bmatrix} \quad (4)$$

Based on Table 1, kinematic parameters are determined by working conditions, the manuscript can find the transformation matrices by directly replacing the parameters in (1) to obtain a set of matrices, similar to Eq. (5)

$$^0T_1 = \begin{bmatrix} cos\ q_1) & 0 & -sin(q_1) & a_1.cos(q_1) \\ sin(q_1) & 0 & cos(q_1) & a_1.sin(q_1) \\ 0 & -1 & 0 & d_1 \\ 0 & 0 & 0 & 1 \end{bmatrix};$$

$$^1T_2 = \begin{bmatrix} cos(q_2) & -sin(q_2) & 0 & a_2.sin(q_2) \\ sin(q_2) & cos(q_2) & 0 & a_2.cos(q_2) \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix};$$

$$^2T_3 = \begin{bmatrix} sin(q_3) & 0 & -cos(q_3) & a_3.cos(q_3) \\ -cos(q_3) & 0 & -sin(q_3) & -a_3.sin(q_3) \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$^3T_4 = \begin{bmatrix} cos(q_4) & 0 & -sin(q_3) & 0 \\ sin(q_4) & 0 & cos(q_3) & 0 \\ 0 & -1 & 0 & -d_4 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$^4T_5 = \begin{bmatrix} cos(q_5) & 0 & sin(q_5) & 0 \\ sin(q_5) & 0 & -cos(q_5) & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$^5T_6 = \begin{bmatrix} -cos(q_6) & 0 & sin(q_6) & 0 \\ -sin(q_6) & 0 & -cos(q_6) & 0 \\ 0 & 1 & 0 & -d_6 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (5)$$

The research will derive a homogeneous coordinate transformation matrix $^0T_6$, which represents the final state of operation from the matrices shown above, as shown in Eq. (6):

$$^0T_6 = {}^0T_1\,{}^1T_2\,{}^2T_3\,{}^3T_4\,{}^4T_5\,{}^5T_6 = {}^0T_3\,{}^3T_6 = \begin{bmatrix} t_{11} & t_{12} & t_{13} & t_{14} \\ t_{21} & t_{22} & t_{23} & t_{24} \\ t_{31} & t_{32} & t_{33} & t_{34} \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (6)$$

2.3. Deep learning model in manuscript

Deep learning is an artificial neural network where neurons are connected to each other, as shown in Figure 5, by weights as in Eq. (7). The output of the network is the sum of the inputs expressed in Eq. (8). Finally, the values of each neuron and their weights are fed into the activation function to turn them into the nonlinear function shown in Eq. (9).

$$u = \sum_{i=1}^{m} w_i q_i + b \quad (7)$$

$$q_i = [q_1 q_2 q_3 \ldots q_{m-1} q_m]^T \quad (8)$$

$$y = \varphi(u + d) \tag{9}$$

which, w_i is the weight corresponding to q_i, b is the coefficient of freedom

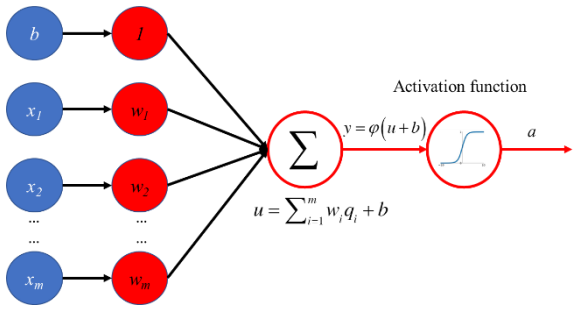

Fig. 5. Deep learning model in manuscript

Usually, in a deep learning network, each neuron in the hidden layer is connected to all the neurons in the network before and after it, so the deep learning network is said to be fully connected. A valid deep learning network must consist of at least three layers, including an input and an output layer and at least one hidden layer, shown as Eqs. 10–14

$$s_j = \sum_{i=1}^{m} w_{ij}\, x_i - b_j \tag{10}$$

$$u_k = f\left(\sum_{j=1}^{H} v_{kj} y_j - d_k\right); k = 1,2,3,\ldots O \tag{11}$$

$$y_j = f(s_j) \tag{12}$$

$$y_j = f\left(\sum_{i=1}^{m} w_{ij}\, x_i - b_j\right) \tag{13}$$

$$y_j = \frac{1}{1+exp\left(-\left(\sum_{i=1}^{m} w_{ij} x_i - b_j\right)\right)}; j = 1,2,\ldots,H \tag{14}$$

where **W** is the input weight matrix as Eq. 15, v is the output weight matrix in Eq. 16, *b* is the increment matrix input as Eq. 17, and *d* is the increment matrix output in Eq. 18

$$W = (w_1 w_2 w_3 \ldots w_{m-1} w_m) \tag{15}$$

$$b = \begin{pmatrix} b_1 \\ b_2 \\ \ldots \\ b_{m-1} \\ b_m \end{pmatrix} \tag{16}$$

$$v = \begin{pmatrix} v_{1,1} & w_{2,1} & w_{3,1} & \ldots & w_{k-1,1} & w_{k,1} \\ v_{1,2} & w_{2,2} & w_{3,2} & \ldots & w_{k-1,2} & w_{k,2} \\ \ldots & \ldots & \ldots & \ldots & \ldots & \ldots \\ v_{1,j-1} & w_{2,j-1} & w_{3,j-1} & \ldots & w_{k-1,j-1} & w_{k,j-1} \\ v_{1,j} & w_{2,j} & w_{3,j} & \ldots & w_{k-1,j} & w_{k,j} \end{pmatrix} \tag{17}$$

$$d = \begin{pmatrix} d_1 \\ d_2 \\ \ldots \\ d_{k-1} \\ d_k \end{pmatrix} \tag{18}$$

The output vector of the deep learning network is influenced by the input vector and the activation function used. Specifically, the output value of the deep learning is calculated as the sum of the values of the input and the value of each neuron multiplied by their weight as shown in Figure 6.

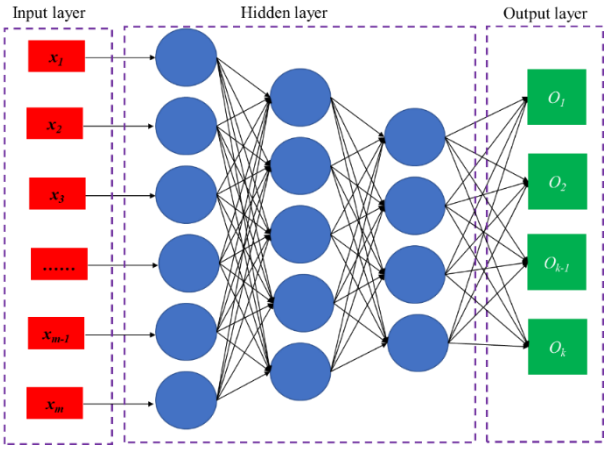

Fig. 6. Fully connected network with 3 hidden layers

3. Kinetic model of the robotic arm model based on deep learning

The transformation of kinematic information from joint variable space to Cartesian coordinate space is known as forward kinematics [26] [27] [28]. The research will then determine the position and direction of the final manipulation based on a specific set of matching variables. Kinetic information investigated in this study includes: link position, link speed, and link acceleration in Figure 7. However, in robotic arm modeling, the forward kinematics problem frequently deals with analyzing the position of the links at different times. As a result, the study of paramagnetic kinematics in this study will investigate the forward positions of the links by deep learning as shown in Figure 8, resulting in the determination of a combined transformation matrix between the links.

$${}^{0}T_n = {}^{0}T_1(q_1)\,{}^{1}T_2(q_2)\,{}^{2}T_3(q_3)\ldots{}^{n-1}T_n(q_n) \tag{19}$$

To find the coordinates of the point P (x_0, y_0, z_0) in the base coordinate system, as its coordinate system is given in the last step:

$${}^{0}r_P = {}^{0}T_n.\,{}^{n}r_P \tag{20}$$

We can deduce the position of the active link's endpoint from this:

$${}^{0}r_P = {}^{0}T_6.\begin{bmatrix} 0 \\ 0 \\ 0 \\ 1 \end{bmatrix} = \begin{bmatrix} t_{14} \\ t_{24} \\ t_{34} \\ 1 \end{bmatrix} \tag{21}$$

At the robotic arm model's resting position in the z and x directions with: with $q_1 = 0$, $q_2 = 90^o$, $q_3 = 90^o$, $q_4 = 0$, $q_5 = 0$, $q_6 = 0$, ${}^{0}T_6$, it becomes:

$$^{0}T_{6} = \begin{bmatrix} 0 & 0 & 1 & d_6 + d_4 + a_1 \\ -1 & 0 & 0 & 0 \\ 0 & -1 & 0 & a_3 + a_2 + d_1 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (22)$$

3.1. Create the motion rule, draw the final operation's trajectory, and test the results by deep learning

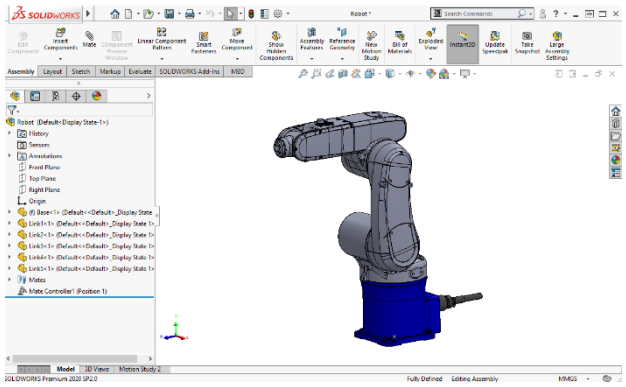

Fig. 7. Denso robot model on Soilidworks software.

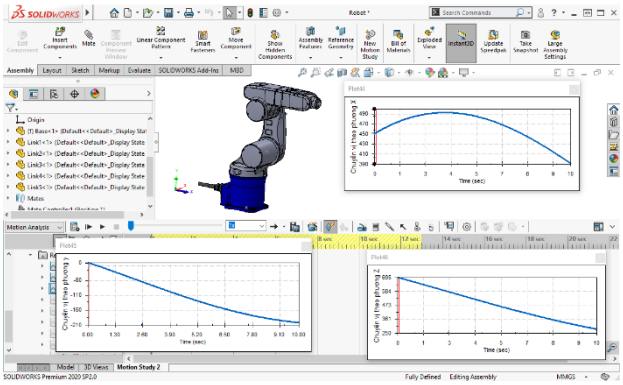

Fig. 8. Using Motion Analysis to generate displacement results in three directions x, y, and z by deep learning

To investigate the above results in Figures 7-8, the manuscript constructs the following movement law of the matching variables q over time t as follows:

$$q = \begin{cases} q_1 = q_4 = q_5 = q_6 = 3t \\ q_2 = -90 + 3t \\ q_3 = 90 + 3t \end{cases} \quad (23)$$

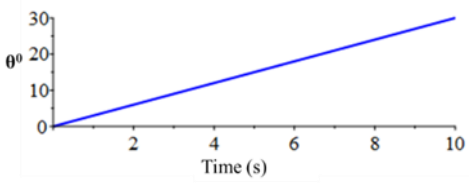

Fig. 9. The graph of the angle q_1, q_4, q_5, q_6 varies with time t

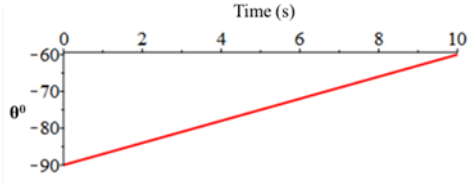

Fig. 10. Graph of angle q_2 varies with time t

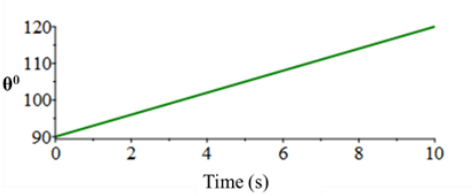

Fig. 11. The graph of angle q_3 varies with time t

We obtain the corresponding translational displacement graphs in the coordinate planes in the space of *Oxy*, *Oxz*, and *Oyz* in the time t = 0....10 (s), which are programmed, calculated, and drawn by Maple software in Figures 9-11.

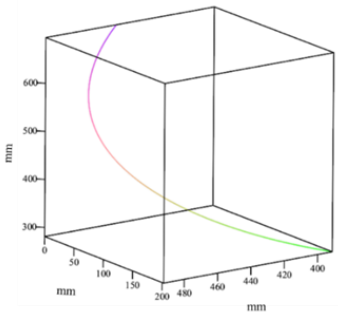

Fig. 12. Graph of the last manipulator's motion trajectory in space by deep learning

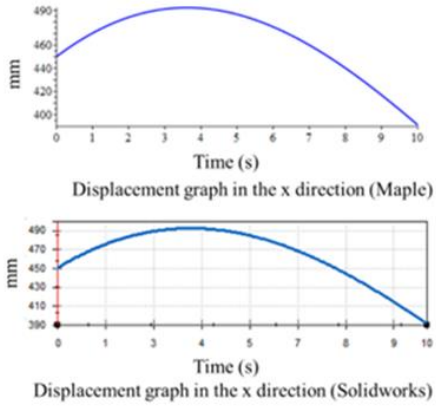

Fig. 13. Compare the displacement graphs in the y direction of the two methods by deep learning

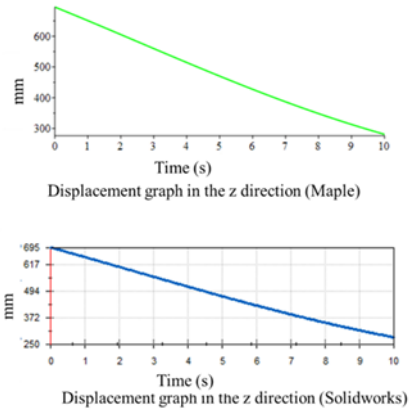

Fig. 14. Compare the displacement graphs in the z direction of the two methods by deep learning

Because the conventional coordinate system in Maple and the system coordinate system in Solidworks are opposite, the graph in Figures 12-14 will produce results with opposite signs. This manuscript get a table of data from the graphs that describes the difference between the two methods.

Table 2: A comparison table of location data between Solidworks and Maple software

N₀	**X'(slw)**	**X (maple)**	**Y' (slw)**	**Y (maple)**	**Z' (slw)**	**Z (maple)**
0	**449.9999801**	**449.9999999**	**-0.0004610**	**0**	**695.0000131**	**695.0000000**
1	**470.3039651**	**470.303960**	**-24.427700**	**24.42816125**	**650.7920117**	**650.7919974**
2	**484.008174**	**484.0081442**	**-49.991935**	**49.99239506**	**605.5437109**	**605.5436959**
3	**491.1268152**	**491.1267604**	**-75.804236**	**75.80469427**	**559.9382375**	**559.9382213**
4	**491.8543813**	**491.8543018**	**-101.010965**	**101.0114197**	**514.6408764**	**514.6408597**
5	**486.5522778**	**486.5521736**	**-124.822784**	**124.8232337**	**470.2847361**	**470.2847182**
6	**475.7294881**	**475.7293596**	**-146.540939**	**146.5413824**	**427.4576682**	**427.4576500**
7	**460.0182938**	**460.0181415**	**-165.578819**	**165.5792547**	**386.6908230**	**386.6908038**
8	**440.1462744**	**440.1460987**	**-181.477977**	**181.4784032**	**348.4491604**	**348.4491407**
9	**416.9059558**	**416.9057570**	**-193.918100**	**193.9185162**	**313.1241576**	**313.1241371**
10	**391.123541**	**391.1233195**	**-202.720739**	**202.7211432**	**281.0288781**	**281.0288565**

Evaluation of the result: At each time unit t(s), the manuscript has displacement error:

$$\Delta x_i = \left||x_i'| - |x_i|\right|; \Delta y_i = \left||y_i'| - |y_i|\right|; \Delta z_i = \left||z_i'| - |z_i|\right| \quad (24)$$

Table 3: Displacement error in the *x*, *y*, and *z* directions

N_o	Δx (mm)		Δy (mm)		Δz (mm)	
	Normal	Deep learning	Normal	Deep learning	Normal	Deep learning
0	0.01980	0.00841	0.46100	0.12451	0.01310	0.01024
1	0.00510	0.00212	0.46125	0.13152	0.01430	0.01147
2	0.02980	0.01254	0.46006	0.12541	0.01500	0.01004
3	0.05480	0.01210	0.45827	0.12879	0.01620	0.01078
4	0.07950	0.04124	0.45470	0.13024	0.01670	0.01012
5	0.10420	0.08452	0.44970	0.12981	0.01790	0.01168
6	0.12850	0.08542	0.44340	0.13145	0.01820	0.01024
7	0.15230	0.09120	0.43570	0.14574	0.01920	0.01231
8	0.17570	0.10852	0.42620	0.13011	0.01970	0.01033
9	0.19880	0.14217	0.41620	0.12755	0.02050	0.01174
10	0.22150	0.15421	0.40420	0.12452	0.02160	0.01424

The average displacement error in each direction is as follows:

$$\delta_{x-deep\ learning} = \frac{\Delta_{x_{tb}}}{x'_{tb}} = 0{,}00023133 \approx 0{,}23\% \qquad \delta_{y-deep\ learning} = \frac{\Delta_{y_{tb}}}{y'_{tb}} = 0{,}000384640 \approx 0{,}38\%$$

$$\delta_{z-deep\ learning} = \frac{\Delta_{z_{tb}}}{z'_{tb}} = 0{,}000366312 \approx 0{,}37\%$$

The study's findings have low error rates and are reliable, indicating two issues:

- The forward kinematics problems of the robotic arm model include the creation of matrices in the manuscript that show the relationship between quantities such as coordinates, velocity, and acceleration between machine parts. As a result, they have high accuracy and low error and can be applied to real-world control models based on deep learning.

- The machine learning application model in the manuscript gives much smaller errors than those commonly used today. The specific appropriate magnetic field range of this model, some basic settings, and the process of comparing and contrasting the results between the virtual simulation (Solidworks) and the calculated results on the algorithm have been built, which is effective and useful during implementation.

4. Conclusion

The study developed a forward kinematics model for a six-degree-of-freedom robotic arm using deep learning, with real-world working conditions. The model utilizes axis transformations, translations, and rotations to determine the position of each link at various times. It was designed from a virtual model, eliminating the need to build a physical model. The study conducted calculations and simulations to compare and verify the results between the virtual software, improving the efficiency of the process. By simulating the movement of the links with specific examples, the manuscript provides specific error cases for each corresponding link, resulting in a more accurate determination of the model's motion laws. In the future, the research will focus on inverse kinematics and dynamics problems, providing a more comprehensive study for the six-degree-of-freedom robotic arm model.

Acknowledgements

This research was supported by AI to correct English mistakes, check grammar, and suggest new words because we come from countries where English is not a native language. However, the entire content of the research was developed by the research team.

Author contributions

Thanh Q. Nguyen: Conceptualization, Methodology, Software, Field study

Khien B. Pham: Data curation, Writing-Original draft preparation, Software, Validation., Field study

Kim-Chi. Duong: Visualization, Investigation, Writing-Reviewing and Editing.

Conflicts of interest

The authors declare no conflicts of interest.

References

[1] Michael Beetz, Ulrich Klank, Ingo Kresse, Alexis Maldonado, Lorenz Mösenlechner, Dejan Pangercic, Thomas Rühr and Moritz Tenorth, "Robotic roommates making pancakes," *in 11th IEEE-RAS International Conference on Humanoid Robots,* Bled, Slovenia – October 26 – 28, 2011, 2011. W.-K. Chen, Linear Networks and Systems. Belmont, CA, USA: Wadsworth, 1993, pp. 123–135.

[2] Hanna Yousef, Mehdi Boukallel and Kaspar Althoefer, "Tactile sensing for dexterous in-hand manipulation in robotics - A review," *Sensors and Actuators A: Physical,* vol. 167, no. 2, pp. 171-187, 2011.

[3] K. Yamazaki and T. Abe, "A Versatile End-Effector for Pick-and-Release of Fabric Parts," *IEEE Robotics and Automation Letters,* vol. 6, no. 2, pp. 1431-1438, 2021.

[4] Shirine El Zaatari, Mohamed Marei, Weidong Li and Zahid Usman, "Cobot programming for collaborative industrial tasks: An overview," *Robotics and Autonomous Systems,* vol. 116, pp. 162-180, 2019.

[5] Sivadas Chandra Sekaran, Hwa Jen Yap, Siti Nurmaya Musa, Kan Ern Liew, Chee Hau Tan and Atikah Aman , "The implementation of virtual reality in digital factory—a comprehensive review," *The International Journal of Advanced Manufacturing Technology,* vol. 115, p. 1349–1366, 2021.

[6] Evgenia Manou, George-Christopher Vosniakos and Elias Matsas , "Off-line programming of an industrial robot in a virtual reality environment," *International Journal on Interactive Design and Manufacturing (IJIDeM),* vol. 13, p. 507–519, 2019.

[7] Hadi Alasti, Behin Elahi and Atefeh Mohammadpour , "Interactive Virtual Reality-Based Simulation Model Equipped with Collision-Preventive Feature in Automated Robotic Sites," in *Simulation for Industry 4.0*, Springer, Cham, Springer Series in Advanced Manufacturing, 2019, p. 111–128.

[8] Michael Brady, "Artificial intelligence and robotics," *Artificial Intelligence,* vol. 26, no. 1, pp. 79-121, 1985.

[9] Phillip Phan, Michael Wright and Soo-Hoon Lee, "Of Robots, Artificial Intelligence, and Work," *Academy of Management Perspectives,* vol. 31, no. 4, pp. 253-255, 2017.

[10] Woodrow Barfield, "Liability for Autonomous and Artificially Intelligent Robots," *Paladyn, Journal of*

Behavioral Robotics, vol. 9, no. 1, p. 193–203, 2018.The Terahertz Wave eBook. ZOmega Terahertz Corp., 2014. [Online]. Available: http://dl.z-thz.com/eBook/zomega_ebook_pdf_1206_sr.pdf. Accessed on: May 19, 2014.

[11] S. H. Alsamhi, Ou Ma and Mohd. Samar Ansari , "Survey on artificial intelligence based techniques for emerging robotic communication," *Telecommunication Systems* , vol. 72, p. 483–503, 2019.

[12] Sandip Panesar, Yvonne Cagle, Divya Chander, Jose Morey, Juan Fernandez-Miranda and Michel Kliot, "Artificial Intelligence and the Future of Surgical Robotics," *Annals of Surgery,* vol. 270, no. 2, pp. 223-226, 2019.

[13] Kuts, Vladimir, Cherezova, Natalia, Sarkans, Martins, Otto and Tauno, "Digital Twin: industrial robot kinematic model integration to the virtual reality environment," *Journal of Machine Engineering,* vol. 20, no. 2, pp. 53--64, 2020.

[14] P. Aivaliotis, G. Michalos and S. Makris, "Cooperating robots for fixtureless assembly: modelling and simulation of tool exchange process," *International Journal of Computer Integrated Manufacturing,* vol. 31, no. 12, pp. 1235-1246, 2018.

[15] Ziwen Yang, Shanying Zhu, Cailian Chen, Gang Feng and Xinping Guan, "Leader-follower formation control of nonholonomic mobile robots with bearing-only measurements," *Journal of the Franklin Institute,* vol. 357, no. 3, pp. 1628-1643, 2020.

[16] Thomas Dos'Santos, Christopher Thomas, Paul Comfort and Paul A. Jones , "The Effect of Angle and Velocity on Change of Direction Biomechanics: An Angle-Velocity Trade-Off," *Sports Medicine,* vol. 48, p. 2235–2253, 2018.

[17] Nabeel Abdulkadhim Athab, Wissam Riad Hussein and Ahmed Amer Mohamed Ali, "A Comparative Study for Movement of Sword Fencing Stabbed According to the Technical Programming in the Game of Fencing Wheelchairs Class B," *Indian Journal of Public Health Research & Development,* vol. 10, no. 5, pp. 1344-1347, 2019.

[18] Salwa M. Al-Masrani and Karam M. Al-Obaidi, "Dynamic shading systems: A review of design parameters, platforms and evaluation strategies," *Automation in Construction,* vol. 102, pp. 195-216, 2019.

[19] Yucheng Zhu, Guangtao Zhai and Xiongkuo Min, "The prediction of head and eye movement for 360 degree images," *Signal Processing: Image Communication,* vol. 69, pp. 15-25, 2018.

[20] Samuel Poirier, François Routhier and Alexandre Campeau-Lecours, "Voice Control Interface Prototype for Assistive Robots for People Living with Upper Limb Disabilities," in *2019 IEEE 16th International Conference on Rehabilitation Robotics (ICORR)*, Toronto, ON, Canada, 2019.

[21] Yen, Shih-Hsiang, Pei-Chong Tang, Yuan-Chiu Lin and Chyi-Yeu Lin, "Development of a Virtual Force Sensor for a Low-Cost Collaborative Robot and Applications to Safety Control," *Sensors,* vol. 19, no. 11, p. ID. 2603, 2019.

[22] Lei Pei and Wei Zhang, Setting and Calculating the Kinetic Parameters of a Hybrid Excavator's Working Unit, Applied Mechanics and Materials, 2011.

[23] Sadat Foumani M., Khatibi Mohammad Mahdi, Moradi Mahdi and Mahdiabadi Morteza, " Kinematic-Kinetic Analysis Of Humanoid Robot Straight Motion," *JOURNAL OF MODELING IN ENGINEERING,* vol. 7, no. 17, p. 0, 2009.

[24] Arijit, Abhishek and Pratihar, Dilip Kumar*, "Inverse dynamics learned gait planning of an exoskeleton to negotiate uneven terrains using neural networks," *International Journal of Hybrid Intelligent Systems,* vol. 13, no. 1, pp. 49-62, 2016.

[25] K. K. Rohith, Navaneeth Varma, A. P. Sudheer and M. L. Joy, "Mathematical Modeling and Comparative Study of 12-DoF Biped Robot Using Screw Theory and Denavit–Hartenberg Convention," in *Innovative Product Design and Intelligent Manufacturing Systems*, Springer, Singapore, Lecture Notes in Mechanical Engineering , 2020, p. 979–989.

[26] M. Giorelli, F. Renda, M. Calisti, A. Arienti, G. Ferri and C. Laschi, "A two dimensional inverse kinetics model of a cable driven manipulator inspired by the octopus arm," in *IEEE International Conference on Robotics and Automation*, Saint Paul, MN, USA, 2012.

[27] Federico Renda, Michele Giorelli, Marcello Calisti, Matteo Cianchetti and Cecilia Lasch, "Dynamic Model of a Multibending Soft Robot Arm Driven by Cables," *Dynamic Model of a Multibending Soft Robot Arm Driven by Cables,* vol. 30, no. 5, pp. 1109-1122, 2014

[28] M. Kawato, Y. Maeda, Y. Uno and R. Suzuki , "Trajectory formation of arm movement by cascade neural network model based on minimum torque-change criterion," *Biological Cybernetics,* vol. 62, p. 275–288, 1990.