Experimental Validation for Inverse Kinematics of a Five Axis Robotic Manipulator Using Deep Artificial Neural Network

Shubham Kamlesh Shah, Ruby Mishra

Abstract*: The use of robotics is to improvise and simplify human life. Robotic manipulators have been around for a while now and are being used in many different sectors such as industries, households, warehouses, medicine etc. Solving of inverse kinematics is one of the most complex issues faced while designing the robotic manipulator. In this research a Deep Artificial neural network (D-ANN) model is proposed to solve inverse kinematics of a 5-axis robotic manipulator with rotary joints. The D-ANN model is trained in MATLAB. Training dataset was generated using forward kinematics equations obtained easily from transformation matrix of the robotic manipulator. To validate predictions made by this model an experimental robotic arm manipulator Is fabricated. A smart camera setup has been linked to MATLAB for real time image processing and calculating the deviation of the end needle in reaching the desired target coordinate. The trained model yielded satisfactory results with ±0.03 radians error and this was also validated experimentally. This research will help the robotic manipulator reach the desired target coordinates even when one does not have enough input data.Paper Setup must be in A4 size with Margin: Top 0.7", Bottom 0.7", Left 0.65", 0.65", Gutter 0", and Gutter Position Top. Paper must be in two Columns after Authors Name with Width 8.27", height 11.69" Spacing 0.2". Whole paper must be with: Font Name Times New Roman, Font Size 10, Line Spacing 1.05 EXCEPT Abstract, Keywords (Index Term), Paper Tile, References, Author Profile (in the last page of the paper, maximum 400 words), All Headings, and Manuscript Details (First Page, Bottom, left side).

***Keywords* : Deep Artificial neural network (D-ANN), Image Processing, Inverse Kinematics, MATLAB, Robotic manipulator.**

I. INTRODUCTION

The robotic technology has become one of the leading technologies in the world providing assistance to humans in almost every field. Robots are being utilized for daily life house duties like cleaning floors with vacuum cleaners, cleaning litter and lawn mower, etc. Nowadays, the necessity of robots in industries is more because large quantities of products can be manufactured in a very less time. Robots help in making high quality products in a precise manner. Robots which can work for 24 hours are used in big industries. These robots can work in place of hundreds of workers at a time. Electronic and vehicle manufacturing companies are also using robots in assembling and in other purposes which is time consuming and difficult to perform by workers.

An attempt was made to develop a nature inspired robotic arm that could mimic the motions of an octopus tentacle. The experimental model was made to replicate standard movements of octopus tentacle such as elongation, shortening, reaching and bending. At the end controls were simplified by muscle arrangement and by muscular hydrostat properties [1]. A 6 degree of freedom open source robotic arm platform was designed for training and vocational purposes. Kinematics and the dynamics of the model was derived using forward and inverse equations [2]. Inverse kinematics of a 3-link manipulator was solved using feed forward neural network method. Trajectories of the path traced were generated in a 2D Cartesian space [3]. Genetic algorithm was used as the proposed approach for controlling trajectory of a robotic arm. Various criterions such as operating time, energy conversion and sum of all rotation angles had been minimized by optimization. The genetic algorithm approach was proved by comparing all the above criterions on an industrial robot ABB IRB 6400FHD with 6 degrees of freedom [4].

Solving of inverse kinematics has been one of the most challenging tasks in designing and controlling a robotic manipulator. In this research a novel method for solving inverse kinematics is proposed. Deep Artificial Neural Network (D-ANN) is a type of machine learning technique which learns and predicts the output based on input. Hence, application of ANN technique is explored as an alternative for solving inverse kinematics of a 5-axis robotic manipulator. The results obtained from the D-ANN is applied on an experimental model of the robotic manipulator and image processing technique is used to test genuine accuracy of the robotic manipulator.

II. METHODOLOGIES

A.Designing

A 5-axis robotic manipulator is designed for the purpose of achieving the aim of this research. This robotic arm has 5 degrees of freedom. Frames are assigned to the joints of the robotic manipulator according to the DH convention [5]. The figure 1 shows the frames assigned to the robotic manipulator.

Revised Manuscript Received on November 10, 2019.

***** Correspondence Author

Shubham Kamlesh Shah, is currently pursuing PhD program in Mechanical engineering at KIIT Deemed to be University, India, PH-+91-9861501763. E-mail: shubham.prasad1@gmail.com

Ruby Mishra*, is currently an associate professor in Mechanical engineering at KIIT Deemed to be University, India, PH-+91-9437678405. E-mail: rubymishrafme@kiit.ac.in

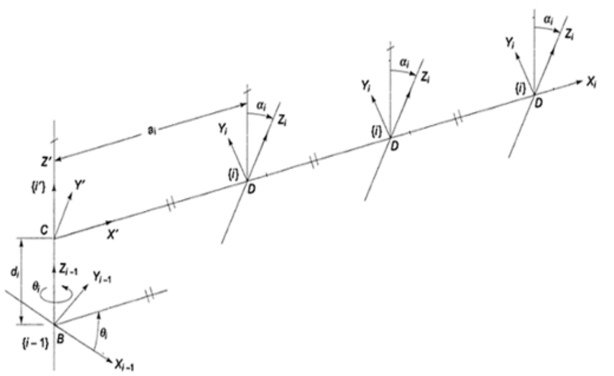

Fig. 1. Convention of frame assignment to each joint.

Table 1 shows the standard DH parameters of the robotic manipulator designed for this research. link lengths are constrained be the dimension of the actuators used for joint motion.

Table 1 Dh Parameters Of The Robotic Arm

Links	*Link length* a_i (cm)	*Link Twist* α_i	*Link Displacement* d_i (cm)	*Joint Angle* θ_i
Waist Link (1)	0	-90^0	$l_1 = 12.35$	θ_1
Shoulder Link (2)	$l_2 = 8.45$	0	$d_2 = -5.9$	θ_2
Arm Link (3)	$l_3 = 7.18$	0	$d_3 = -8.6$	θ_3
Forearm Link (4)	$l_4 = 6.06$	0	$d_4 = -7.0$	θ_4
End needle Link (5)	$l_5 = 4.62$	0	$d_5 = -5.17$	θ_5

B. Experimental Model

The experimental model of the robotic arm designed here has 5 joints 6 links including the base and the end effector. The sixth link is considered as end effector which is a needle used for reaching the target point [6]. Actuators used in this robotic manipulator is dynamixel servo motors with maximum 15 kg-cm torque. Figure 2 shows the experimental robotic manipulator mounted on a wooden base

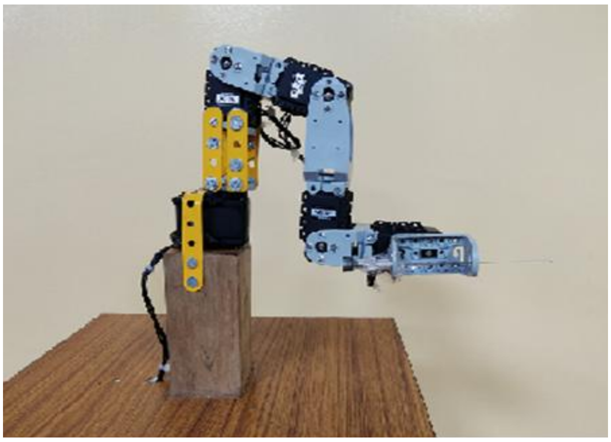

Fig. 2. Complete model of the experimental robotic arm.

In figure 2 A needle is placed at the end of the robotic manipulator for measuring the exact deviation using smart camera and image processing in the above-mentioned model.

III. EQUATIONS

Initially Artificial Neural network needs to be trained in order to predict output values. Forward kinematics is used to generate training dataset. Equation 1 shows the transformation matrix of the robotic manipulator with respect to its base frame.

$$\begin{pmatrix} C\theta_1 \times C(\theta_2+\theta_3+\theta_4+\theta_5) & -C\theta_1 \times S(\theta_2+\theta_3+\theta_4+\theta_5) & -S\theta_1 & X \\ S\theta_1 \times C(\theta_2+\theta_3+\theta_4+\theta_5) & -S\theta_1 \times C(\theta_2+\theta_3+\theta_4+\theta_5) & -C\theta_1 & Y \\ S(\theta_2+\theta_3+\theta_4+\theta_5) & -C(\theta_2+\theta_3+\theta_4+\theta_5) & 0 & Z \\ 0 & 0 & 0 & 1 \end{pmatrix} = T = {}^0_5T \tag{1}$$

$$\begin{aligned} X = &\left(cos(\theta_1) \times l_2 \times cos(\theta_2)\right) - \left(sin\theta_1 (d_2+d_3+d_4+d_5)\right) + (l_3 \times cos(\theta_1) \times cos(\theta_2+\theta_3) \\ &+ (l_4 \times \left(cos(\theta_1) \times cos(\theta_2+\theta_3+\theta_4)\right) \\ &+ (l_5 \times \left(cos(\theta_1) \times cos(\theta_2+\theta_3+\theta_4+\theta_5)\right) \end{aligned} \tag{2}$$

$$\begin{aligned} Y = &cos(\theta_1)(d_2+d_3+d_4+d_5) + (sin(\theta_1) \times l_2 \times cos(\theta_2) + (l_3 \times sin(\theta_1) \times cos(\theta_2+\theta_3) \\ &+ (l_4 \times \left(sin(\theta_1) \times cos(\theta_2+\theta_3+\theta_4)\right) \\ &+ (l_5 \times \left(sin(\theta_1) \times cos(\theta_2+\theta_3+\theta_4+\theta_5)\right) \end{aligned} \tag{3}$$

$$\begin{aligned} Z = &(l_1) - (l_2 \times sin\theta_2)) - (l_3 \times \sin(\theta_2+\theta_3) \\ &- l_4 \times \sin(\theta_2+\theta_3+\theta_4)) - l_5 \times sin(\theta_2+\theta_3+\theta_4+\theta_5)) \end{aligned} \tag{4}$$

Equation (2) (3) and (4) shows relationship between cartesian coordinates and joint angles. Just from these three equations it is observed that there could be multiple solutions for reaching a target point. To avoid this issue constraints to each joint angle were provided according to quadrants in three-dimensional axis. Separate ANN can be trained for each quadrant and unique solution can be achieved for each quadrant. Using this logic, joint angles were provided and X, Y and Z axis coordinates were generated using above equations (2), (3) and (4) for training data.

Cartesian space coordinates generated were used as input for training the D-ANN model and joint angles were designated as output. Training of D-ANN was done using MATLAB software. Network architecture of the proposed D-ANN is shown in figure 3.

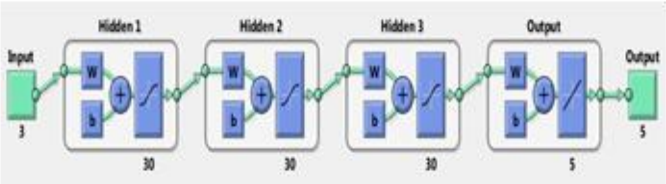

Fig. 3. Network architecture of D-ANN model.

As seen in figure 3 the D-ANN has an input layer and output layer and 3 hidden layers with 30 neurons in each layer. There are three inputs and five outputs signifying cartesian coordinates and joint angles respectively. Training of the D-ANN is done for 500 epochs. Training algorithm used for this research is Levenberg-Marquardt (LM) function and performance is calculated using mean squared error (MSE).

IV. RESULTS

After training was completed, performance of the training was observed. Figure 4 shows MSE convergence graph of the training validation and testing data obtained after training. Training, validation and testing data were split randomly in 60%, 20% and 20% respectively.

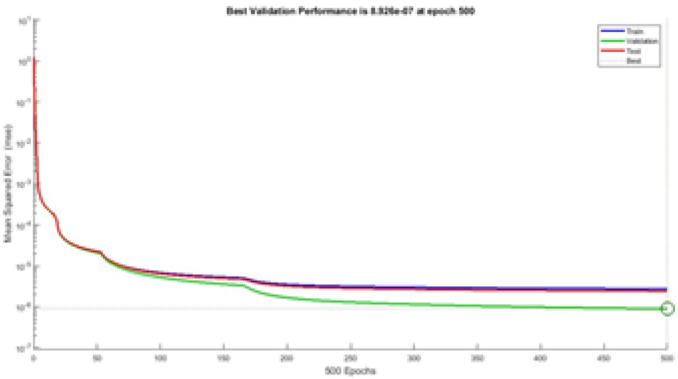

Fig. 4. MSE Convergence graph after training.

From figure 4 it is observed that the training, validation and testing data of the model is tending towards 0 and is achieving near convergence at 8.926 × 10-07 value. Lesser the value of convergence better the model is trained.

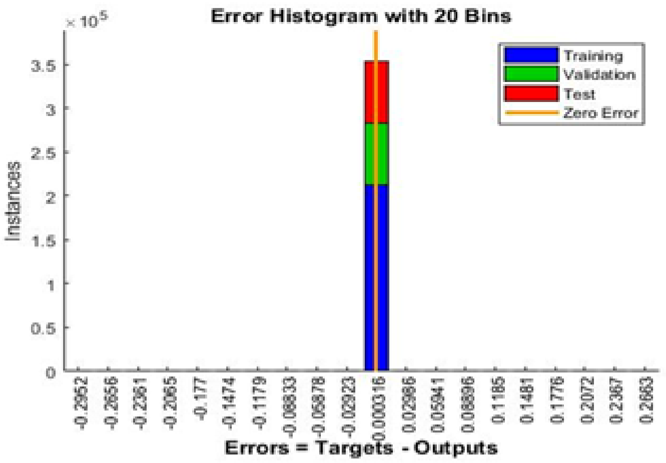

Fig. 5. Error histogram plot

Figure 5 shows a histogram plot between number of training datasets and error found after validating and testing. This graph shows the maximum and minimum error one will get after prediction as compared to the actual value found from inverse kinematics. Here maximum and minimum value found from the graph will be 0.03 to -0.03 radians.

For testing the accuracy of the experimental robotic arm, a point is marked on an apple, which is then placed in trained D-ANN quadrant of the workspace. The target coordinate points are fed as input to the trained D-ANN model and joint angles were received as output in return using MATLAB. After getting the required joint angles, the robotic arm is programmed, to reach for that target point marked on the apple. Figure…. Shows the experimental setup of the robotic manipulator at starting position.

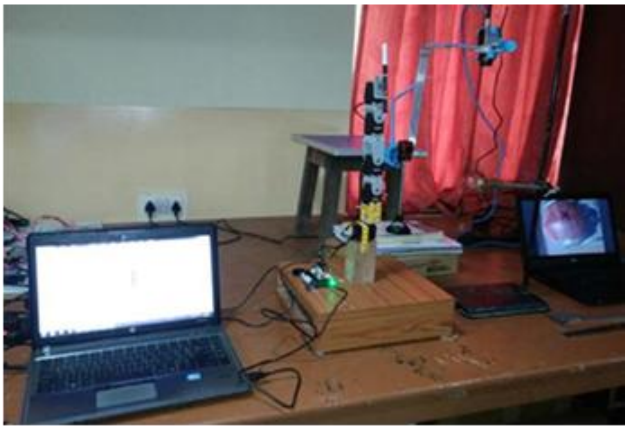

Fig. 6. Robotic manipulator at starting position

The deviation in reaching the target point is measured in 3 dimensions using two Basler smart Cameras which is interfaced with MATLAB in real time. Using the image processing technique for measuring the pixel distance and converting it to real world dimension, actual deviation is found.

To calculate the size of each pixel captured by the cameras in mm a simple mathematical formula is used which is shown in equations (5), (6) and (7).

Focal length of the lens / CCD sensor width = Standoff distance / Field of view (5)

Size of one pixel in mm = Field of view in mm / 1278 (6)

Deviation in the respective axis in mm = Size of one pixel in mm × no. of pixels found through image processing (7)

Fig. 7. Deviation analysis using image processing

This process validates the application of D-ANN model for solving inverse kinematics of a 5-axis robotic manipulator in real world application. Multiple experiments were performed to test the repeatability and accuracy of the D-ANN model and robotic manipulator for application in real world. Table 2 shows all the five experiments performed for two different target coordinates and the deviations in X, Y and Z axes that are calculated using image processing done in MATLAB. The deviation is found approximately in the order of 0.3 to 0.4mm in X axis, 0.1 to 0.3 mm in Y axis and 0.2 mm in Z axis.

Table 2 Deviations In X, Y And Z Axis

Sl. No.	X-axis (mm)	Y-axis (mm)	Z-axis (mm)	Deviation in each axis (mm)		
				X- axis	Y- axis	Z- axis
1	260	0	0	0.32054	0.31772	0.196
2	260	0	0	0.329	0.29082	0.184
3	260	0	0	0.311	0.304	0.126
4	260	0	0	0.306	0.296	0.154
5	260	0	0	0.36848	0.2645	0.193
1	256	-17	80	0.293	0.287	0.172
2	256	-17	80	0.2415	0.238	0.115
3	256	-17	80	0.238	0.244	0.13
4	256	-17	80	0.246	0.339	0.145
5	256	-17	80	0.29175	0.36275	0.124

V. CONCLUSION

In this research a 5-axis robotic arm manipulator was designed successfully for experimental purposes. A novel method for solving the inverse kinematics of a robotic manipulator is proposed using D-ANN machine learning technique. D-ANN model, after training yielded satisfactory results as the error was within the range of ±0.03 radians. To test this theoretical D-ANN model, arbitrary coordinate points were taken within the quadrant of the trained model in the workspace and robotic manipulator was programmed to reach that target point. Validation was done using smart camera and image processing technique was used to calculate the deviation of the end needle from the e=intended target point. Deviation was found to be in the range of 0.3 to 0.4mm in X axis, 0.1 to 0.3 mm in Y axis and 0.2 mm in Z axis. These deviations are negligible and could also occur due to actuator limitations. This research validates that D-ANN can be used as an alternative for solving inverse kinematics.

To reduce the deviation more training is required. Training of the datasets in different quadrants will unlock more dexterity of the robotic manipulator.

ACKNOWLEDGMENT

We wholeheartedly thank KIIT Deemed to be University for providing the resources and infrastructure for this research.

REFERENCES

1. Cianchetti, M., Arienti, A., Follador, M., Mazzolai, B., Dario, P., & Laschi, C. (2011). Design concept and validation of a robotic arm inspired by the octopus. Materials Science and Engineering: C, 31(6), 1230-1239
2. Manzoor, S., Islam, R. U., Khalid, A., Samad, A., & Iqbal, J. (2014). An open- source multi-DOF articulated robotic educational platform for autonomous object manipulation. Robotics and Computer-Integrated Manufacturing, 30(3), 351-362.
3. Guiochet, J., Tondu, B., & Baron, C. (2003, October). Integration of UML in human factors analysis for safety of a medical robot for tele-echography. In Intelligent Robots and Systems, 2003.(IROS 2003). Proceedings. 2003 IEEE/RSJ International Conference on (Vol. 4, pp. 3212-3217). IEEE.
4. Števo, S., Sekaj, I., & Dekan, M. (2014). Optimization of robotic arm trajectory using genetic algorithm. IFAC Proceedings Volumes, 47(3), 1748-1753.
5. R. K. Mittal, I. J. Nagrath, Book: Robotics and Control, Publisher: Tata Mc Graw Hill.
6. Shah Shubham k., Mishra, R. and Mohapatro G.S., "experimental and theoretical design analysis and modeling of a CT image guided robotic arm" in International Conference on engineering applied science and technology. 978-1-5386-4956-5/18, July 2018

AUTHORS PROFILE

Shubham Kamlesh Shah is currently pursuing his PhD in Mechanical engineering with specialization in robotics from KIIT Deemed to be University, Bhubaneswar, India. He specializes in kinematics and dynamics of robotic manipulator, machine learning, image processing and robotic design.

Dr. Ruby Mishra is currently an associate professor at school of mechanical engineering, KIIT Deemed to be University, Bhubaneswar, India. Her specialization is joint link kinematics from KIIT Deemed to be University She specializes in kinematics and dynamics of robotic manipulator.