Research Article

A New Artificial Neural Network Approach in Solving Inverse Kinematics of Robotic Arm (Denso VP6242)

Ahmed R. J. Almusawi,[1,2] L. Canan Dülger,[1] and Sadettin Kapucu[1]

[1]*Mechanical Engineering Department, University of Gaziantep, Gaziantep, Turkey*
[2]*Mechatronics Engineering Department, University of Baghdad, Baghdad, Iraq*

Correspondence should be addressed to Ahmed R. J. Almusawi; ahmed.almusawi@mail2.gantep.edu.tr

This paper presents a novel inverse kinematics solution for robotic arm based on artificial neural network (ANN) architecture. The motion of robotic arm is controlled by the kinematics of ANN. A new artificial neural network approach for inverse kinematics is proposed. The novelty of the proposed ANN is the inclusion of the feedback of current joint angles configuration of robotic arm as well as the desired position and orientation in the input pattern of neural network, while the traditional ANN has only the desired position and orientation of the end effector in the input pattern of neural network. In this paper, a six DOF Denso robotic arm with a gripper is controlled by ANN. The comprehensive experimental results proved the applicability and the efficiency of the proposed approach in robotic motion control. The inclusion of current configuration of joint angles in ANN significantly increased the accuracy of ANN estimation of the joint angles output. The new controller design has advantages over the existing techniques for minimizing the position error in unconventional tasks and increasing the accuracy of ANN in estimation of robot's joint angles.

1. Introduction

Artificial intelligence has become the most modern technology of robotic control. It has many advantages in performance such as precise control and less computing time, in addition to overcoming some mathematical problems in motion and path generation. The main problem of motion control in robotic arm is to find the accurate and reliable solution for inverse kinematics. The calculation of inverse kinematics is necessary in real-time control; the solving of inverse kinematics is computationally complex and requires a very long processing time. Most applications of motion in robotic manipulation require Cartesian space motion control. In inverse kinematics, desired position and orientation of the end effector in Cartesian space are given while the set of robot's joint angles in joint space is calculated.

In general, the solutions of inverse kinematics of a robotic manipulator are geometric, iterative, analytic, or algebraic approaches. Recently, a high focus has been applied on artificial intelligence based methods for inverse kinematics problem solution of general purpose robot. Many studies were done on the implementation of artificial intelligence on a robotic arm to overcome the singular configuration problem of robotic arm. The inverse kinematics of three DOF robotic arm was solved by multilayer network inversion method; the joint angles were estimated for given end effector position in a simulation of three-link robotic arm. The results showed an approximation solution for inverse kinematics [1]. Singularities and uncertainties in arm configurations are the main complications in the kinematics of robot control, in order to have a realistic solution based on one of the heuristic methods; artificial neural network (ANN) was suggested for a nonsurgical robot. The main idea of this approach was the use of ANN to learn the robot system characteristics rather than having to specify an explicit robot system model [2]. A neural network and genetic algorithms were used together to solve the inverse kinematics problem of the nonsurgical robotic manipulator to minimize the error at the end effector and improve the precision of the inverse kinematics solution [3]. The inverse kinematic of redundant manipulators was presented by neural networks (NNs) to obtain the joint angles of the robot using the Cartesian coordinate of the end effector.

Position errors of end effector and feasibility of the joint angles were obtained [4]. An artificial neural network was used for controlling 3 DOF robotic manipulator. The methods introduced a nonlinear relation between Cartesian and joint coordinates using multilayer perceptron in artificial neural network. A simulation test was implemented [5]. A neural network architecture was introduced to solve the inverse kinematics problem for robotics manipulators with two degrees of freedom. The neural networks were multilayered perceptron (MLP) with a backpropagation training algorithm for reducing the complexity of the algorithm and calculation (matrix inversion) of inverse geometric of robotic arm. The result showed a mean squared error (MSE) in performance near to 10^{-5} [6]. A high-order-logic theorem was used for solving the kinematics analysis of six-axis revolute joint robot. The approach required an enormous amount of user intervention to overcome the limitation of kinematic analysis [7]. The inverse kinematics problem of the 6 DOF robot is solved by using curved-surface scanning to carry the ultrasonic testing task. Many results for the joint angles were acquired; the method of the shortest distance was assumed to solve the inverse problem of the robot system. A 3D application software was introduced to simulate the motion of ultrasonic trajectory and path planning [8]. The kinematics and singularities of an asymmetrical parallel robotic wrist were investigated by using the method of Lagrange multipliers and considering all the mobile components. The designed model was numerically illustrated to show its computation accuracy [9]. A technique for solving the inverse kinematics problem using artificial neural networks was introduced for a PUMA 560 robot. An inverse kinematic solution was studied by training the neural network with the robot's end effector Cartesian coordinates and its corresponding joint configurations. Results showed mean square error (MSE) of 1.2178, the regression value obtained was 0.87527, and the position errors in x-, y-, and z-axis were 4.93%, 7.29%, and 3.73%, respectively [10]. A solution to the inverse kinematics was required for generating desired trajectories in the Cartesian space (2D). A feedforward neural network was used for planar of three-link manipulators. The result showed the best performance at epoch 9 with mean squared error (MSE) of 0.0054387 [11]. The kinematics of three DOF was introduced for the lower limb of the humanoid robot. Decoupled closed-form solution for the position and orientation was the solution of kinematics; the joint sequences were presented by Denavit-Hartenberg (DH) transformation matrices. Swing phase equations were developed to avoid matrix inversion problems [12]. The kinematic parameters on industrial robot were affected by vibrations disturbance. The error in motion was improved by sensors of accelerometer and gyroscope. The motion profile was analyzed for joint; then, the path tracking of welding task was estimated [13]. Human robot kinematics was identified by geometry kinematics approach to map human arm configuration and stiffness controlled index by hand gesture. The human arm stiffness was estimated within robot experiential stability region. A moving task was implemented to test the performance of geometry kinematics approach on Baxter robot simulator [14]. The geometric approach was used to solve the kinematics of the autonomous positioning of a robotic arm. This modeling and analysis approach was tested by using a five DOF arm with a gripper mounted to the *iRobot* mobile platform [15]. Online robot kinematics parameter errors estimation based on inertial measurement unit (IMU) was presented. It obtained the orientation of the manipulator with the orientation of the IMU in real time. This approach incorporated Factored Quaternion Algorithm (FQA) and Kalman Filter (KF) to the orientation of the IMU [16]. An analytical solution of inverse kinematics for a five DOF spatial parallel micromanipulator was presented. A geometrical mode and structural of system were introduced for the microrobot's task [17]. Forward kinematics and inverse kinematics were calculated and simulation was done for joints and link parameters of six-axis robotic arm. Trajectory planning was described for the requisite motion of the manipulator as a time sequence task [18]. Forward and inverse kinematics of a KUKA robotic arm in the application of a simple welding process were introduced. A general DH representation of forward and inverse matrix was obtained. A movement flow planning was designed and developed for the programming of the robot [19]. The mobile robot with arm (KUKA youBot) and the solving of inverse kinematics problem were introduced. The robot was presented as 8 DOF. The kinematics redundancy of the holonomic platform was presented. Including redundancy parameters, the inverse kinematics solution was suggested [20]. The end effector position and orientation error of a space robot were studied. A geometric parameter identification method was presented based on a laser ranger attached to the end effector. The independence of the geometric parameters was analyzed. Identification equations were derived by simulation which was implemented for different types of robot configuration [21].

This paper introduces a novel solving method for six-axis manipulator robot based on ANN to be used in motion control. It is clear from previous survey that no study has included current joint angles of robot in their ANN. In this study, the ANN architecture has included current joint angles of robot in the input pattern and it improves the performance of proposed ANN in solving inverse kinematics. It is the first ANN that fulfils the requirement of robot precise motion and reduces joint angles error and outcomes in some aspects of the robot tasks.

This paper is organized as follows. Section 2 presents the kinematics analysis and the required parameters for motion control followed by an explanation of problem formulation. The proposed artificial neural network is described in Section 3. Section 4 illustrates the system setup. Section 5 presents the experimental work and discussion of results. Finally, Section 6 concludes this paper.

2. Kinematics Analysis

The kinematics of serial manipulator describe the relationship between the joint angles and the position and orientation of its end effector. The kinematics of robot is required in trajectory generation and motion control. The transformation
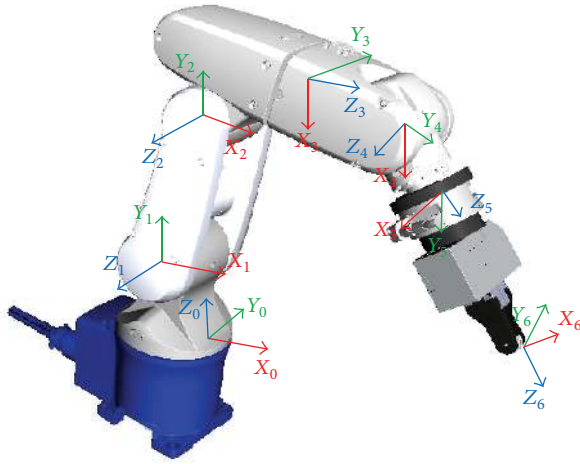
FIGURE 1: Reference coordinates for the system.

TABLE 1: DH parameters of the Denso robot.

Link i	θ_i	d_i	a_i	α_i
1	q_1	0.125	0	pi/2
2	q_2	0	0.21	0
3	q_3	0	−0.075	−pi/2
4	q_4	0.21	0	pi/2
5	q_5	0	0	−pi/2
6	q_6	0.07	0	0

The system has six links and a gripper. The homogeneous transformation matrix is calculated by multiplication of matrices as follows:

$$T^0_{\text{gripper}} = A_1 A_2 A_3 A_4 A_5 A_6 A_{\text{gripper}}, \tag{3}$$

where

matrices have been used for control. The robotic system is the Denso robot with 6 revolute joints. The kinematics analysis is done after system coordinate frame has been performed, the coordinates O_0, x_0, y_0, z_0 are fixed to the base which is the base frame. The other coordinate frames are attached to the corresponding links. The reference coordinates of the system are shown in Figure 1.

The homogeneous transformation matrix is stated to represent the position and orientation of end effector with respect to base coordinate; a homogeneous transformation matrix T^0_6 for overall system is as follows:

$$T^0_6 = \begin{bmatrix} R^0_6 & P^0_6 \\ 0 & 1 \end{bmatrix}, \tag{1}$$

where R^0_6 is a rotation matrix 3×3 and P^0_6 is a position vector of the end effector in the base frame coordinate. The Denavit-Hartenberg DH method is used to analyze the kinematics of Denso robot. The robot transformation matrix has been denoted. The single link homogenous transformation matrix A_i is

$$A_i = \text{Rot}_{z,\theta i} \text{Trans}_{x,di} \text{Trans}_{x,\alpha i} \text{Rot}_{x,\alpha i},$$

$$A_i = \begin{bmatrix} C\theta_i & -S\theta_i C\alpha_i & S\theta_i S\alpha_i & a_i C\theta_i \\ S\theta_i & C\theta_i C\alpha_i & -C\theta_i S\alpha_i & a_i S\theta_i \\ 0 & S\alpha_i & C\alpha_i & d_i \\ 0 & 0 & 0 & 1 \end{bmatrix}, \tag{2}$$

where i is link number, $S\theta_i = \sin\theta_i$, $C\theta_i = \cos\theta_i$, θ_i is the joint rotation angle, a_i is the length of links, α_i is the twist angles, d_i is the link offsets, and θ is the joint angles.

The Denavit-Hartenberg HD parameters of the robot are shown in Table 1.

$$A_1 = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & -1 & 0 \\ 0 & 1 & 0 & .125 \\ 0 & 0 & 0 & 1 \end{bmatrix},$$

$$A_2 = \begin{bmatrix} 1 & 0 & 0 & .21 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix},$$

$$A_3 = \begin{bmatrix} 1 & 0 & 0 & -.075 \\ 0 & 0 & 1 & 0 \\ 0 & -1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix},$$

$$A_4 = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & -1 & 0 \\ 0 & 1 & 0 & .21 \\ 0 & 0 & 0 & 1 \end{bmatrix},$$

$$A_5 = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & -1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix},$$

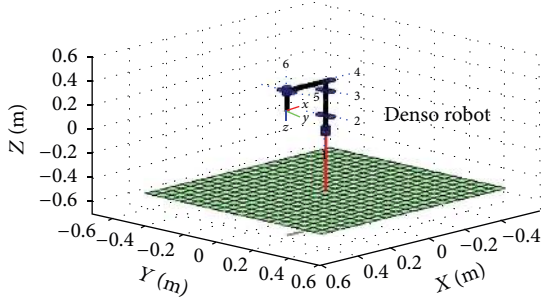$$A_6 = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & .17 \\ 0 & 0 & 0 & 1 \end{bmatrix}. \tag{4}$$

FIGURE 2: The graphical representation for the robotic system.

The transformation matrix of the gripper is A_{gripper}; the joint angles and gripper transformation matrix are given; then, the transformation matrix T_{Home} of home position is calculated:

$$T_{\text{Home}} = \begin{bmatrix} -1 & 0 & 0 & .21 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & -1 & .24 \\ 0 & 0 & 0 & 1 \end{bmatrix}. \tag{5}$$

The graphical representation of Denso robot is done by MATLAB programming robotics toolbox as shown in Figure 2.

The inverse kinematics might got several solutions produced for each of the joint angles because these are corresponding to many robot configurations such as elbow up, elbow down, wrist up, wrist down, shoulder forward, and shoulder back. The position and orientation of end effector are obtained by forward kinematics as follows:

$$T_6^0 = \text{forward kinematics} \, [q]. \tag{6}$$

The joint angles are calculated by inverse kinematic for desired position/orientation of end effector:

$$[q] = \text{inverse kinematics} \left(T_6^0 \right). \tag{7}$$

In this study, a new artificial neural network solution for inverse kinematics of (8) is introduced in Section 3:

$$\begin{bmatrix} q_1 \\ q_2 \\ q_3 \\ q_4 \\ q_5 \\ q_6 \end{bmatrix} = \text{inverse kinematics} \begin{pmatrix} p_x \\ p_y \\ p_z \\ \theta_x \\ \theta_y \\ \theta_z \end{pmatrix}. \tag{8}$$

2.1. The Problem Statement. In general, the desired motion of robot is carried out in the Cartesian coordinate, while the robotic arm motion is controlled by joint coordinate; a solution for the inverse kinematics is very important to be calculated. Solving the inverse kinematics problem for robotic manipulators is a difficult and also quite challenging task. The difficulty of this problem is given by the robot's

geometry and the nonlinear trigonometric equations that describe the relationship between the Cartesian space and the joint space. Although a closed-form solution to this problem is preferable in robotics, sometimes it is impossible to find. Therefore, various other ways to determine the solution for inverse kinematics problem were studied such as geometrical solutions and numerical algorithms. This task depends on the designed structure of the robot while many robots such as redundant manipulators do not have an analytical solution for the inverse kinematics.

In this study, two ANNs are designed for inverse kinematics of robotic arm. The first one is the traditional ANN as used in serial robotics inverse kinematics analysis, and the second is the proposed ANN by considering the feedback of current robot configuration (current joint angles) in the design of ANN.

3. Traditional Design of Artificial Neural Network

A traditional design for ANN is used in many studies [6, 10, 11]. In order to utilize the advantages of this proposed method, traditional ANN is designed in this study to solve the inverse kinematics. In this ANN, the elements in the input layer are six variables, which are the position P and orientation R of gripper in Cartesian coordinates. The number of hidden layers is ten. The output layer has six elements of the angles of joint Q. MATLAB/neural network toolbox is used for training, validation, and testing. Figure 3 shows a block diagram for traditional ANN and its model as follows:

$$[Q] = \text{ANN\_Traditional\_Net}\,(P, R). \tag{9}$$

The inputs are uniformly enclosed with the workspace of specified position; the corresponding inputs/outputs are computed by solution of forward kinematics. In this way, each position of the robot has a unique joint configuration in the neural network inputs/outputs set.

The training algorithm is the Levenberg-Marquardt backpropagation; it is used to assure fast convergence of the training error and is also a very popular curve-fitting algorithm. Figure 4 shows the performance of the traditional neural network; the MSE of training is decreased until the validation error is stopped at epoch 121 and MSE was $1.1892e^{-5}$ in the best performance. A closed relation was between the output and target samples at correlation of 0.99758.

4. Proposed Artificial Neural Network Design

Always, robot starts motion from current position, and in most applications robot moves on trajectory on sequential point's path. So the inclusion of current joint configuration in ANN has a positive effect in the estimation of joint angles for the next desired position. In this paper, a novel neural network design is proposed and used to solve the inverse kinematics problem of robotic arm. The proposed method relies on the constraints of the kinematics of robotic arm to achieve robot's motion in an intelligent way with high
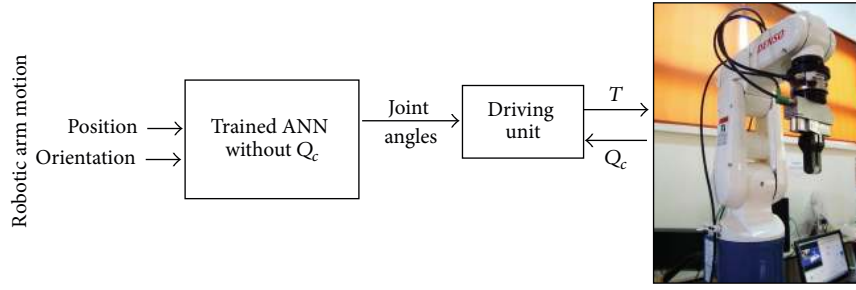
Figure 3: Block diagram of robot motion control by traditional ANN.
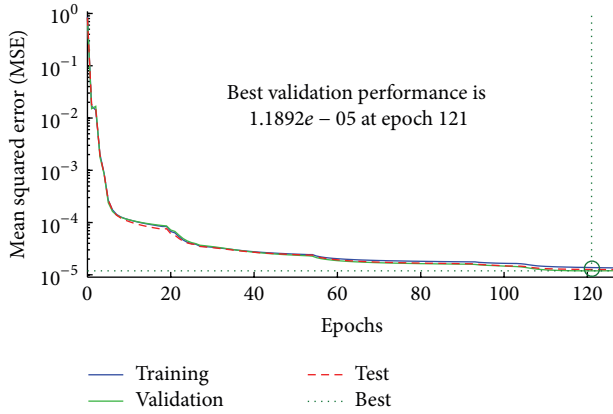

Figure 4: The performance of the traditional neural network.

accuracy in position. The contribution of proposed design is considering the current joint angles of the robot in solving the inverse kinematics of robot by ANN. The inclusion of the current joint angles during training produces a strong network and adjusts the weights with very low error.

The forward kinematics is used to generate the input/target data set which is used in training; the inputs of neural network are the desired position/orientation and current robot joint configuration while the targets are the required joint angles of the robot relative to those points. The proposed neural network has 13 elements in the input layer, which are the gripper position P and orientation R in Cartesian coordinates and current joint angles Q_c of the robot. The output layer has six elements of joint coordinates, which are the joint angles Q of the robot. Figure 5 shows a block diagram for proposed ANN, the inputs, and outputs of neural network as follows:

$$[Q] = \text{ANN\_Proposed\_Net}(P, R, Q_c),$$
$$Q = [q_1, q_2, q_3, q_4, q_5, q_6],$$
$$P = [x, y, z], \tag{10}$$
$$R = [R_x, R_y, R_z],$$
$$Q_c = [q_{1c}, q_{2c}, q_{3c}, q_{4c}, q_{5c}, q_{6c}].$$

The performance of the neural network was determined based on the mean squared error (MSE) between the neural network's actual output and the desired output. The performance of the proposed ANN is shown in Figure 6. The differences between the network outputs and target are calculated through the mean squared error (MSE); it drops rapidly through the learning process; the MSE of training is decreased until the validation error is stopped at epoch 68 and MSE was $3.3029e^{-8}$ in the best performance. A closed relationship was between output and target samples at correlation 0.99999. The new outputs of the network are checked by the test data, the training samples were 2800, and the validated and test samples were 600 for each one.

5. System Setup

Denso robot VP6242 is a six-axis industrial robot (Quanser Company). Artificial neural network is implemented on this system for predicting joint angles during real-time Cartesian motion. The robot is communicated with MATLAB/Simulink via TCP/IP. An open source program (QUARC) control software is supported; the QUARC software is executed in Simulink for real-time application. A blockset is used to connect Simulink program with Denso driving unit. The joint positions and joint currents (ampere) are reading on PC while the joint velocities/positions are sending to robot. The joint PID parameters or joint feedforward gains are adjustable and user can deal with them. The total arm length is 420 mm, and the payload is 2.5 kg. Figure 7 shows the control parameters between robot and QUARC software SW.

The gripping system is an electrical drive gripper WSG 32 SCHUNK; it is a very precise handling system for medium parts weight up to 0.5 kg, and it has integrate high-sensitivity sensor to detect parts in a gripping force of 5–50 N and opening of up to 65 mm. The gripper is required for pick-and-place part or other handling activities. Figure 8 shows robotic system full setup.

A virtual model is connected with Simulink to visualize the system by using Simulink 3D animation. The virtual model and the system model are combined to create a virtual robotic system environment. This virtual model can record and present the motion path of real robot during experiments. Figure 9 shows the virtual model of robotic system.
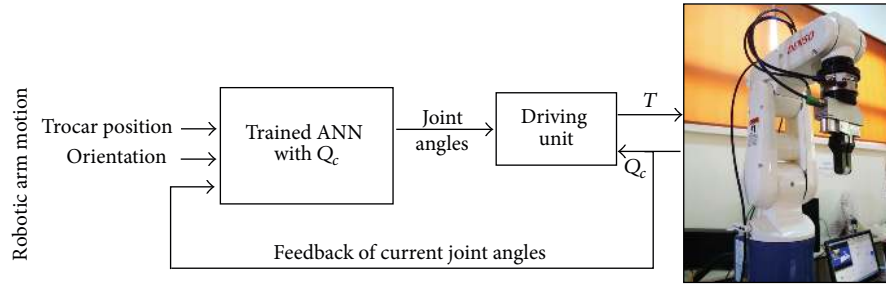
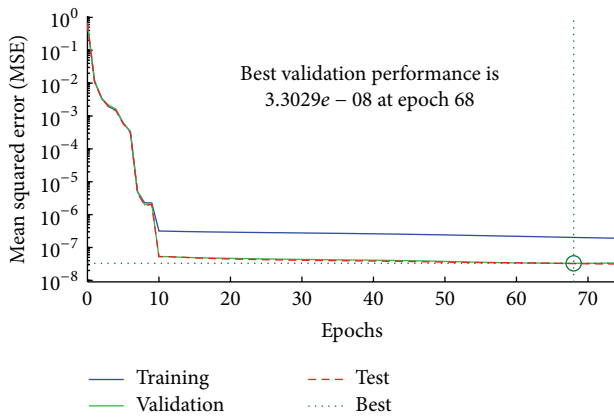FIGURE 5: Block diagram of robot motion control by proposed ANN.

FIGURE 6: The performance of the proposed neural network.

FIGURE 7: The control parameters between robot and QUARC software.
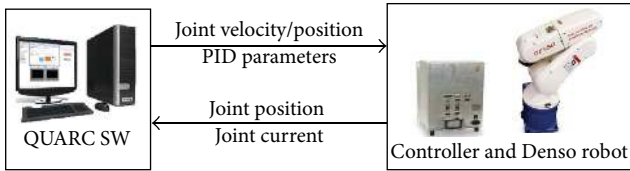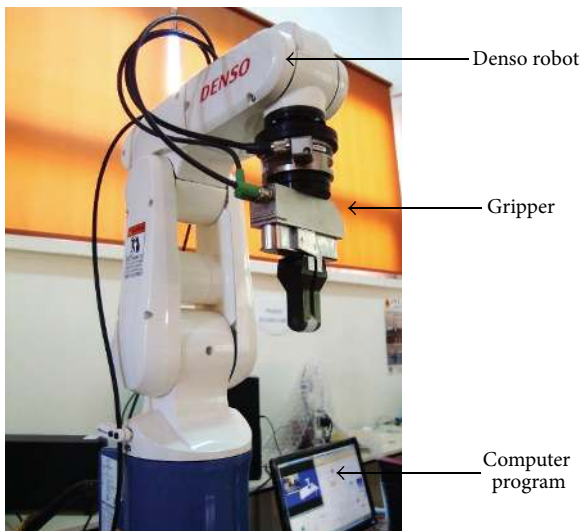

FIGURE 8: Robotic arm system setup.

FIGURE 9: Virtual model of the robotic system.

6. Experiment Results and Discussion

After the training of proposed ANN and traditional ANN is completed, the experiments of movement are carried out by the robotic system. The gripper motion is in a helical path within the workspace area. The input path for gripper tip is generated by $P_x(k) = c_1 \sin(k)$, $P_y(k) = c_2 \cos(k)$, and $P_z(k) = c_3 k$. In this experiment, the robot has to follow a sinusoidal rotation in x-axis and y-axis while the motion in z-axis is a linear path; the constants c_1 and c_2 are the amplitude of radius on the x- and y-axis, respectively; the constant c_3 is the pitch of a helix, and k is the sample time. Figure 10 shows the real robotic system and the virtual model to demonstrate helical motion path by ANN. The inverse kinematics solution for robot motion is achieved by proposed ANN and traditional ANN.

The movements of robot are executed in circular and linear motion. The pitch of path of the end effector is changed by varying the position in z direction relative to workspace coordinates. The rotation of end effector is performed by changing the position to x and y directions. Two paths are generated by using proposed ANN and traditional ANN with desired path. Figure 11 illustrates the paths configuration of the end effector by the robot movement.

The motion of the robot is in a Cartesian space. In Figure 12(a), the motion in x direction is an enlargement of the time period (20 s–25 s). Figure 12(b) is the motion in y direction enlarged for the time period (26 s to 30 s). Figure 12(c) is the motion in z direction enlarged for the time period (14 s to 19 s). Figures mentioned above show the
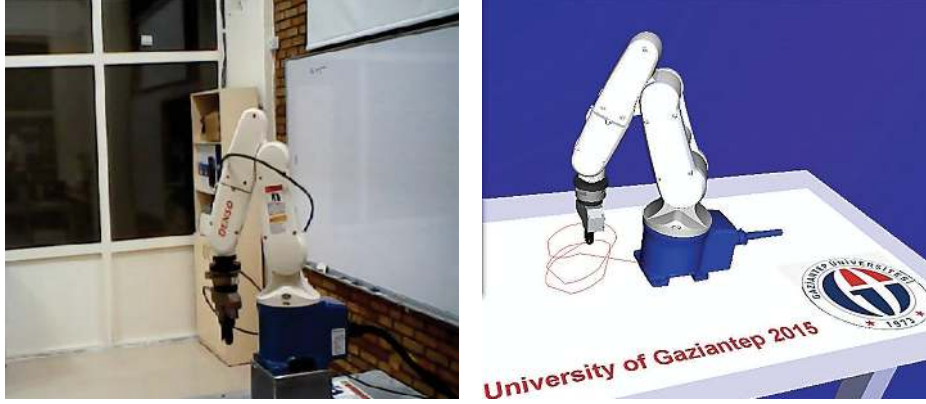
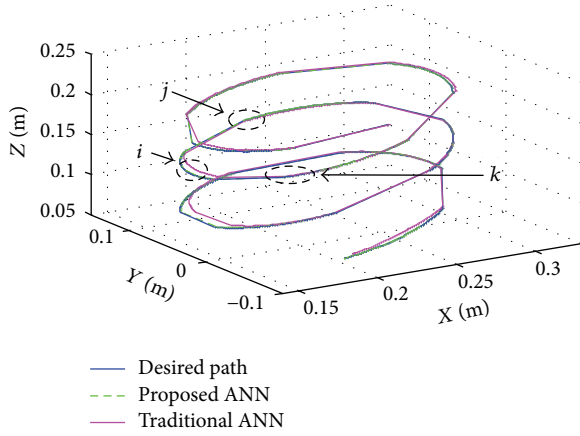FIGURE 10: Experiment on the robotic motion demonstration by ANN (movement of end effector through helical path).

FIGURE 11: Paths configuration generated by robot in two different ANNs. The selected points i, j, and k are enlarged portions in Figure 12.

TABLE 2: Performance of robotic system by ANNs.

Parameters	Proposed ANN	Traditional ANN
P_x error%	0.17	5.78
P_y error%	0.36	7.25
P_z error%	0.12	1.28
MSE	$3.3029e^{-8}$	$1.1892e^{-5}$
Regression	0.99999	0.99758

According to the ANN based solving inverse kinematics results in the literature [2–6, 10, 11], the proposed approach in this study has a minimized error in the inverse kinematics solution. Table 3 shows system performance comparison between this study and other studies from literature. The high accuracy and the low MSE that are obtained in this work can be obviously seen.

The errors are increased in some points and are reduced in other points; the point of high error is because of estimating position by ANN while the point of low error is the point that is near the samples of training set in ANN.

The proposed ANN has given higher accuracy and precision in position than the traditional ANN, and this method is applicable in precise robotic motion.

7. Conclusions

This study introduced a very accurate solution for inverse kinematics by using the artificial neural network to overcome the drawbacks of traditional ANN controller. A new design of artificial neural network ANN has been proposed for the optimal robot motion control in Cartesian coordinates. In order to evaluate the integral performance of the system, the current joint angles information was added to the traditional ANN based inverse kinematics solution. The proposed design showed improvement in performance of end effector in some aspects.

The motion of robot has been executed; it satisfies the constraint of robotic arm motion by the designed structure. Robot kinematics are analyzed, and position/orientation of end effector in different configurations are studied.
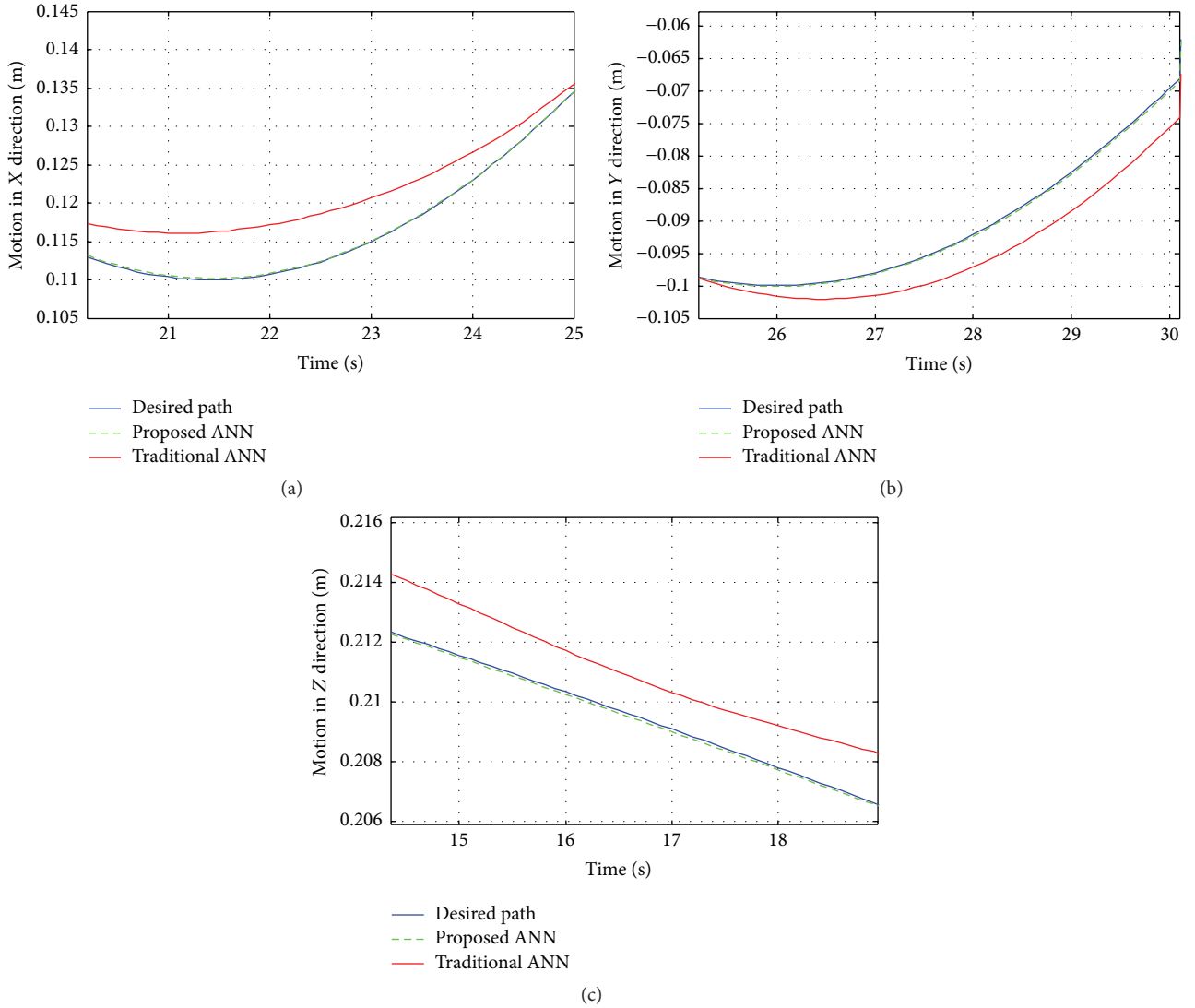
differences in x, y, and z directions for motion resulting from proposed ANN and traditional ANN and the high accuracy and precision of proposed ANN.

The differences between desired path and artificial neural network path are measured by calculating the error in position from proposed ANN and traditional ANN with respect to position from desired path. Figure 13 shows the error in x, y, and z position for proposed ANN, while Figure 14 shows the errors in x, y, and z position for traditional ANN.

The motion curves of proposed ANN are much more precise than the motion curves of traditional ANN in estimating the joint angles of the robot for desired positions. The maximum error in x direction for proposed design of ANN was near 0.22 millimeters and for traditional design of ANN2 was 6.5 millimeters. The maximum error in y direction for proposed design of ANN was less than 0.3 millimeters and for traditional design of ANN2 was near 6 millimeters. The maximum error in z direction for proposed design of ANN was 0.35 millimeters and for traditional design of ANN2 was near 2.5 millimeters. Table 2 shows the error percentages of proposed and traditional ANNs.

Figure 12: Robot motion by two ANNs: (a) the motion in x direction, (b) the motion in y direction, and (c) the motion in z direction.

Table 3: System performance comparison between this study and other studies mentioned in the literature.

Study	System	DOF	Method	MSE	Hidden layers	Errors
Proposed	Denso	6	ANN	$3.3e^{-8}$	10	$x = 0.17\%$ $y = 0.36\%$ $z = 0.12\%$
Luv et al., 2014 [10]	PUMA 560	6	ANN	1.217	30	$x = 6.42\%$ $y = 4.90\%$ $z = 2.92\%$
Hasan et al., 2010 [2]	FANUC M-710i robot	6	ANN	~1	1	$x = 3.34\%$ $y = 6.72\%$ $z = 0.35\%$
Toshani and Farrokhi, 2014 [4]	Simulation PA-10 robot	7	NNs with optimization	1	—	End effector 5 mm
Duka, 2014 [11]	Planar simulation	3	NNT	0.0054	1	—
Köker, 2013 [3]	Stanford	6	ANN	2.38	25	End effector 4.28 mm
Nanda et al., 2012 [5]	Simulation	3	ANN, FLANN	>1	20	—
Daya et al., 2010 [6]	simulation	2	NNT	$5.24e^{-5}$	2	—
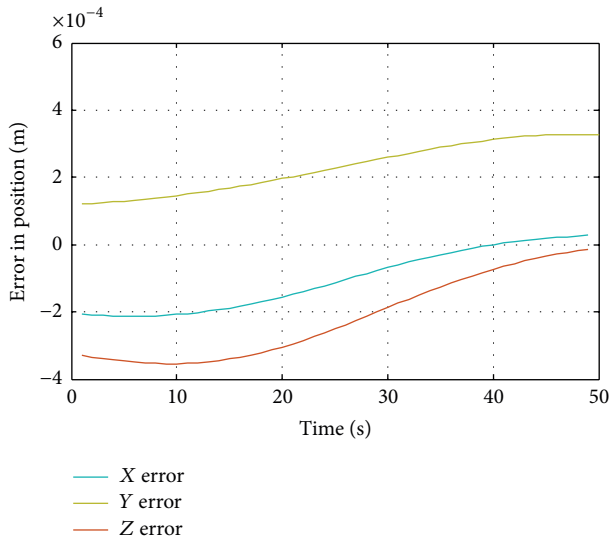
FIGURE 13: Proposed ANN motion error in x, y, and z direction.
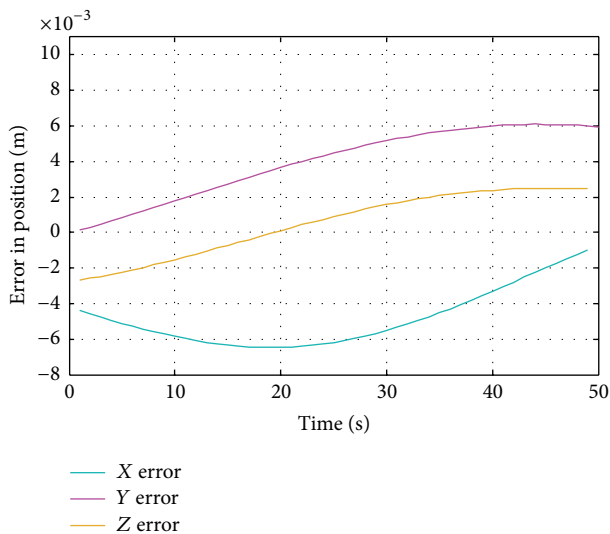
FIGURE 14: Traditional ANN motion error in x, y, and z direction.

A comparison has been carried out between two ANNs. The parameters of motion and the errors were calculated. The results showed that the proposed ANN has superior performance in terms of the joint angles estimation. The design of ANN compared to other techniques is applicable for some of the most difficult and challenging problems of kinematics. These results have proved the effectiveness of the proposed ANN. The inclusion of current configuration of joint angles in ANN increased the accuracy of ANN estimation and succeeded in mapping between input position and joint angles output.

Competing Interests

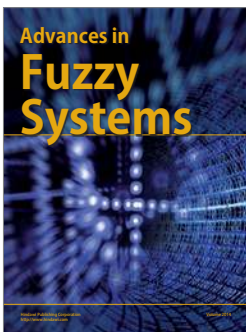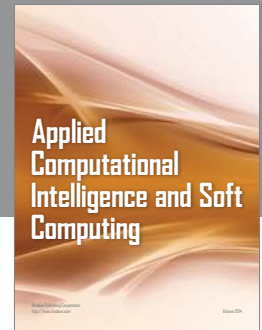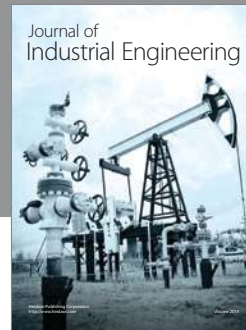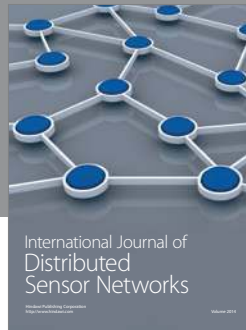The authors declare that there is no conflict of interests regarding the publication of this paper.

References

[1] T. Ogawa and H. Kanada, "Solution for Ill-posed inverse kinematics of robot arm by network inversion," *Journal of Robotics*, vol. 2010, Article ID 870923, 9 pages, 2010.

[2] A. T. Hasan, N. Ismail, A. M. S. Hamouda, I. Aris, M. H. Marhaban, and H. M. A. A. Al-Assadi, "Artificial neural network-based kinematics Jacobian solution for serial manipulator passing through singular configurations," *Advances in Engineering Software*, vol. 41, no. 2, pp. 359–367, 2010.

[3] R. Köker, "A genetic algorithm approach to a neural-network-based inverse kinematics solution of robotic manipulators based on error minimization," *Information Sciences*, vol. 222, pp. 528–543, 2013.

[4] H. Toshani and M. Farrokhi, "Real-time inverse kinematics of redundant manipulators using neural networks and quadratic programming: a Lyapunov-based approach," *Robotics and Autonomous Systems*, vol. 62, no. 6, pp. 766–781, 2014.

[5] S. K. Nanda, S. Panda, P. R. Subudhi, and R. K. Das, "A novel application of artificial neural network for the solution of inverse kinematics controls of robotic manipulators," *International Journal of Intelligent Systems and Applications*, vol. 4, no. 9, pp. 81–91, 2012.

[6] B. Daya, S. Khawandi, and M. Akoum, "Applying neural network architecture for inverse kinematics problem in robotics," *Journal of Software Engineering & Applications*, vol. 3, no. 3, pp. 230–239, 2010.

[7] A. Wu, Z. Shi, Y. Li et al., "Formal kinematic analysis of a general 6R manipulator using the screw theory," *Mathematical Problems in Engineering*, vol. 2015, Article ID 549797, 7 pages, 2015.

[8] Z. Lu, C. Xu, Q. Pan, X. Zhao, and X. Li, "Inverse kinematic analysis and evaluation of a robot for nondestructive testing application," *Journal of Robotics*, vol. 2015, Article ID 596327, 7 pages, 2015.

[9] G. Wu, "Kinematics and dynamics of an asymmetrical parallel robotic wrist," *Journal of Robotics*, vol. 2014, Article ID 296250, 13 pages, 2014.

[10] A. Luv, A. Kush, and J. Ruth, "Use of artificial neural networks for the development of an inverse kinematic solution and visual identification of singularity zone(s)," *Procedia CIRP*, vol. 17, pp. 812–817, 2014.

[11] A. V. Duka, "Neural network based inverse kinematics solution for trajectory tracking of a robotic arm," *Procedia Technology*, vol. 12, pp. 20–27, 2014.

[12] A. Said, E. Rodriguez-Leal, R. Soto, J. L. Gordillo, and L. Garrido, "Decoupled closed-form solution for humanoid lower limb kinematics," *Mathematical Problems in Engineering*, vol. 2015, Article ID 437979, 14 pages, 2015.

[13] D. A. Elvira-Ortiz, R. de Jesus Romero-Troncoso, A. Y. Jaen-Cuellar, L. Morales-Velazquez, and R. A. Osornio-Rios, "Vibration suppression for improving the estimation of kinematic parameters on industrial robots," *Shock and Vibration*, vol. 2016, Article ID 6954012, 15 pages, 2016.

[14] P. Liang, L. Ge, Y. Liu, L. Zhao, R. Li, and K. Wang, "An augmented discrete-time approach for human-robot collaboration," *Discrete Dynamics in Nature and Society*, vol. 2016, Article ID 9126056, 13 pages, 2016.

[15] K. E. Clothier and Y. Shang, "A geometric approach for robotic arm kinematics with hardware design, electrical design, and implementation," *Journal of Robotics*, vol. 2010, Article ID 984823, 10 pages, 2010.

[16] G. Du and P. Zhang, "IMU-based online kinematic calibration of robot manipulator," *The Scientific World Journal*, vol. 2013, Article ID 139738, 10 pages, 2013.

[17] D. Prusak, K. Kobus, and G. Karpiel, "Kinematics analysis of a novel five-degree-of-freedom spatial parallel micromanipulator," *Journal of Robotics*, vol. 2014, Article ID 806294, 13 pages, 2014.

[18] K. Kishore, A. Srinath, G. Jugalanvesh, R. Premsai, and M. suresh, "Kinematic analysis and simulation of 6Dof KukaKr5 robot for welding application," *International Journal of Engineering Research and Applications*, vol. 3, no. 2, pp. 820–827, 2013.

[19] M. Dahari and J.-D. Tan, "Forward and inverse kinematics model for robotic welding process using KR-16KS KUKA robot," in *Proceedings of the 4th International Conference on Modeling, Simulation and Applied Optimization (ICMSAO '11)*, Kuala Lumpur, Malaysia, April 2011.

[20] S. Sharma, G. K. Kraetzschmar, C. Scheurer, and R. Bischoff, "Unified closed form inverse kinematics for the KUKA youBot," in *Proceedings of the 7th German Conference on Robotics*, pp. 1–6, Munich, Germany, May 2012.

[21] Y. Liu, Z. Jiang, H. Liu, and W. Xu, "Geometric parameter identification of a 6-DOF space robot using a laser-ranger," *Journal of Robotics*, vol. 2012, Article ID 587407, 11 pages, 2012.