International Journal of Electrical and Computer Engineering (IJECE)
Vol. 10, No. 2, April 2020, pp. 1376~1386
ISSN: 2088-8708, DOI: 10.11591/ijece.v10i2.pp1376-1386

Inverse kinematic analysis of 4 DOF pick and place arm robot manipulator using fuzzy logic controller

Tresna Dewi[1], Siti Nurmaini[2], Pola Risma[3], Yurni Oktarina[4], Muhammad Roriz[5]
[1,3,4,5]Electrical Engineering Department, Politeknik Negeri Sriwijaya, Indonesia
[2]Intelligent System Research Group, Faculty of Computer Science, Universitas Sriwijaya, Indonesia

Article Info

Article history:

Received Sep 9, 2019
Revised Oct 17, 2019
Accepted Oct 27, 2019

Keywords:

Degree of freedom
Forward kinematicsl inverse kinematics
Fuzzy logic controller

ABSTRACT

The arm robot manipulator is suitable for substituting humans working in tomato plantation to ensure tomatoes are handled efficiently. The best design for this robot is four links with robust flexibility in x, y, and z-coordinates axis. Inverse kinematics and fuzzy logic controller (FLC) application are for precise and smooth motion. Inverse kinematics designs the most efficient position and motion of the arm robot by adjusting mechanical parameters. The FLC utilizes data input from the sensors to set the right position and motion of the end-effector. The predicted parameters are compared with experimental results to show the effectiveness of the proposed design and method. The position errors (in x, y, and z-axis) are 0.1%, 0.1%, and 0.04%. The rotation errors of each robot links (θ_1, θ_2, and θ_3) are 0%, 0.7% and 0.3%. The FLC provides the suitable angle of the servo motor (θ_4) responsible in gripper motion, and the experimental results correspond to FLC's rules-based as the input to the gripper motion system. This setup is essential to avoid excessive force or miss-placed position that can damage tomatoes. The arm robot manipulator discussed in this study is a pick and place robot to move the harvested tomatoes to a packing system.

Corresponding Author:

Siti Nurmaini,
Intelligent System Research Group, Faculty of Computer Science,
Universitas Sriwijaya,
Jalan Raya Prabumulih-Inderalaya, Palembang, Indonesia.
Email: siti_nurmaini@unsri.ac.id

1. INTRODUCTIO

Due to its location on the equator, Indonesia is blessed with tropical climate and abundance of biodiversity to support Indonesia economic growth in the agriculture and plantation sector. One of the essential commodities of Indonesia plantation is tomatoes. A tomato requires special handling during the harvesting time to ensure only the ripe ones picked, and none of them are crushed or damaged. The efficiency and quality improvement are made possible by the application of digital farming employing a robot to raise the quality and hygiene of the harvested fruit [1-9].

The arm robot manipulator is the most suitable type of robots to be applied in plantation and agriculture for harvesting and packaging. The arm can be customized to imitate the human's arm motion from one point to others during harvesting. Robot motions can be designed using an inverse kinematics method to generate the desired trajectory, and the robot follows the generated trajectory. The inverse kinematics output is the ideal parameters and angles of robot links to ensure the smooth motion during harvesting time [10-21]. The suitable end-effector of robot applied as a harvesting robot is a gripper. Moreover, in order to achieve gripper's smooth motion, artificial intelligence (AI) is applied to utilize the input from the attached sensors at a robot's system. The commonly used AI is the fuzzy logic controller (FLC) [22-25] and the neural network (NN) [26-28]. Many kinds of research have applied inverse kinematics to generate a robot trajectory. However, most of the previous research did not apply AI to ensure

the smoothness of the robot [1-23]. Therefore, it is necessary to design a ready to applied arm-robot in handling the harvested tomatoes. This design can be facilitated by carefully planning the mechanical parameters in inverse kinematics design and simulated the resulting design in SciLab. The motion smoothness can be achieved by implementing FLC to the end-effector design.

This paper discussed the design and control of a pick and place arm robot applied in tomato packaging equipped with a gripper at the end-effector utilized to pick and place a target. In this study, the considered target is a tomato to pick from a starting position and place into a belt conveyor. The mechanical parameters of the arm robot manipulator are designed and predicted using the inverse kinematics method. The designed parameters achieved from inverse kinematics are used to generate robot trajectory, and the smoothness of gripper motion is ensured by applying FLC utilizing the input from proximity sensors attached to the starting position of target and robot's end-effector. The first proximity sensor is for detecting the availability of tomato. The second sensor is to sense the distance between the gripper and the tomato. The parameters from inverse kinematics also decide the angle of gripper opening to ensure the right gripping position in holding the tomato; therefore, no damage caused during gripping and placing the tomato. The novelty of this research is by designing a 4 DOF arm robot manipulator using inverse kinematics analysis and combined with an FLC to ensure the smoothness of gripper motion. The input and experimental results are compared to show the effectiveness of the proposed method. The experiment is conducted by employing the robot to pick the harvested tomato from a box to a belt conveyor. The designed robot is relatively cheap and straightforward, only employing two ultrasonic sensors as the proximity sensors.

2. RESEARCH METHOD

This study proposes the combination of inverse kinematics method and FLC to control a 4 DOF arm robot manipulator applied in the digital farming system. The proposed method ensures the right design of the mechanical system and the smoothness of the end-effector motion. The elaboration of the proposed method is shown as a schematic diagram in Figure 1, while robot detail and the experimental setup are shown in Figure 2. The primary input is the proximity sensors data where the first proximity sensor is to detect target availability, and the second proximity sensor to detect the distance between the end-effector and the target. The end-effector in this study is a gripper. The first proximity sensor is attached to the target pool, and the second proximity sensor is attached to the gripper. These sensors positions are shown in Figure 2(a), and Figure 2(b) shows the target in the final position or at the beginning of a belt conveyor. The proximity sensors are the input to FLC in term of rules-based, which is dicussed in FLC design. FLC is employed to decide the decision to move the robot based on the input from proximity sensors attached to the robot system. The rules-based decides the rotation angles of end-effector's motor rotation (θ_4) related to mechanical design in kinematics analysis.

The 4 DOF arm robot manipulator considered in this study is shown in Figures 2 and 3, where each link is powered by a servo motor. Each servo motor rotation creates the angle or orientation of the robot, and the angles are named θ_1, θ_2, and θ_3, respectively, as presented in Figure 3(a) and 3(b). θ_1, θ_2, and θ_3 are solved by kinematics analysis or widely known as inverse kinematics method. The kinematics analysis method consists of forward and inverse kinematics of a robot. This method is conducted to design the motion and pose of a robot based on the ideal mechanical condition of the applied robot such as the length of the links presented in Figure 3(c). The output of inverse kinematics gives the precise parameters to design the robot's mechanics and motions. The physical meaning of inverse kinematics outputs is to regulate how much the servo motors must rotate to ensure the right position of the robot and the gripping angle (θ_4) applied while holding the target.

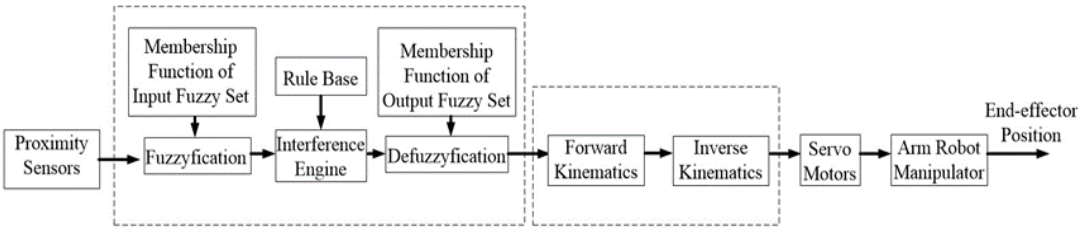
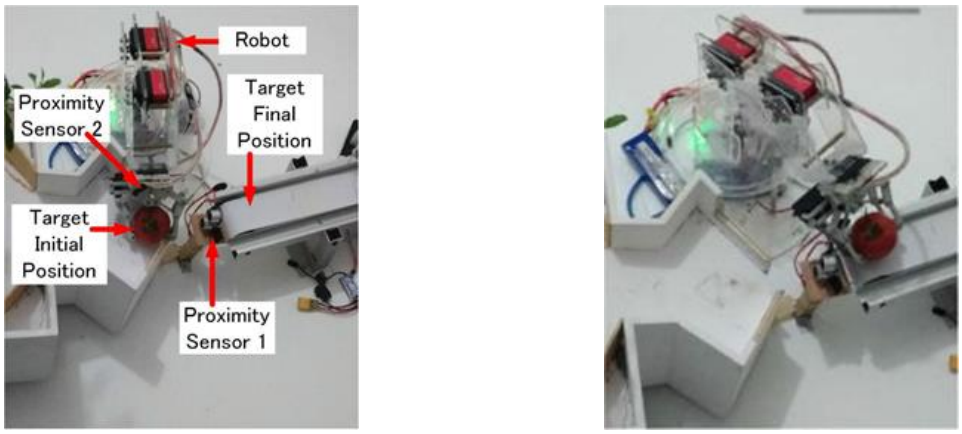
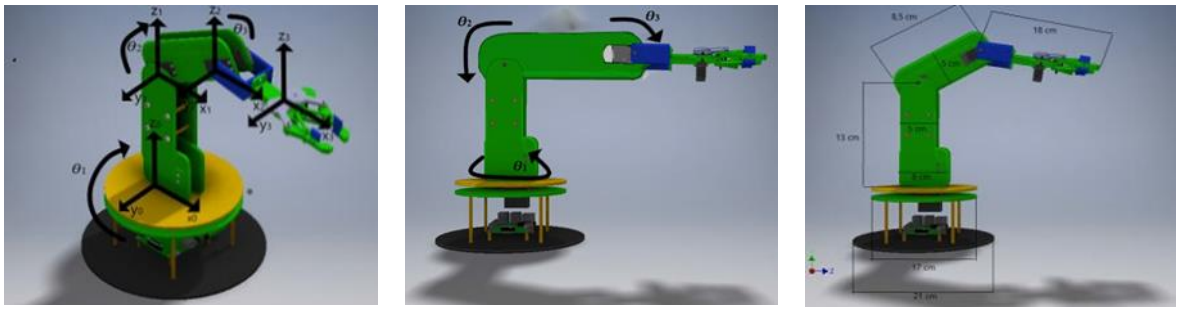


Figure 1. The schematic diagram of the proposed method



(a) (b)

Figure 2. Detail robot and experimental setup,
(a) Robot detail and target positions, (b) Target in Final Position



(a) (b) (c)

Figure 3. Four DOF arm robot applied in this study, (a) Robot in coordinate frames,
(b) Robot in initial position, (c) Robot dimension

2.1. Kinematics analysis method

Kinematics is a method of modeling robot motion without considering the force applied to move the robot. Kinematics analysis is divided into forward and inverse kinematics. By applying kinematics modeling, the robot design can be efficient by considering how the robot moves in x, y, and z-axis. The arm robot considered in this study is shown in Figure 3. θ_1 is the angle between the first joint angle and the base, θ_2 and θ_3 are the second and third angles, and θ_4 is the angle of end-effector or gripper. The initial position of the arm robot manipulator is shown in Figure 3(b).

2.1.1. Forward kinematics modeling

Kinematics analysis of arm robot manipulator starts from forward kinematics. Forward kinematics utilizes mechanical parameters to calculate robot configuration and inverse kinematics to reverse the derivation to acquire the desired configuration. The generic form of forward kinematics of an arm robot manipulator is given by

$$ {}^{0}_{3}T = Forward\ Kinematic\ [P], \quad (1) $$

While the inverse kinematics of an arm robot manipulator is given by

$$ [P] = Forward\ Kinematic\ {}^{0}_{3}T \quad (2) $$

Where [P] is the desired position in x, y, and z-axis. ${}^{0}_{3}T$is the kinematics solution of the robot. Inverse kinematics results are the angle of robot's motion links by modeling robot parameters based on the target

position, the length of the link, the angle between links and those parameters becomes the input to get the transformation matrices. Those matrices are the forward kinematics required to get inverse kinematics. The forward kinematics uses parameters to calculate the robot configuration, and the kinematic inverse reverses this calculation to determine the parameters in achieving the desired configuration.

There are two ways to solve inverse kinematics of a robot that is using the inverse kinematic trigonometry method and Denavit Hartenberg (DH) shown in Table 1. In this study, the mechanical parameters are achieved by deriving DH analysis to get the transformation matrix. The robot parameters from Figure 3(a) are sorted in Table 1, and the transformation matrix is given by

$$ {}^{0}_{1}T = \begin{bmatrix} C\theta_1 & -S\theta_1 & 0 & 0 \\ S\theta_1 C\alpha_1 & C\theta_1 C\alpha_1 & -S\theta_1 & 0 \\ S\theta_1 S\alpha_1 & C\theta_1 S\alpha_1 & C\theta_1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, {}^{2}_{1}T = \begin{bmatrix} C\theta_2 & -S\theta_2 & 0 & 0 \\ S\theta_2 & C\theta_2 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (3) $$

$$ {}^{3}_{2}T = \begin{bmatrix} C\theta_3 & -S\theta_3 & 0 & \alpha_2 \\ S\theta_3 & C\theta_3 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, \text{and } {}^{4}_{3}T = \begin{bmatrix} C\theta_4 & -S\theta_4 & 0 & \alpha_3 \\ S\theta_4 & C\theta_4 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} $$

where Cθn is cos θn and Sθn is sin θn, and

Table 1. Denavit hartenberg parameters

NO	θ_n	d_{n-1}	a_{n-1}	α_{n-1}
1	θ_1	0	0	α_1
2	θ_2	0	a_2	0
3	θ_3	0	a_3	0
4	θ_4	0	a_3	0

2.1.2. Inverse kinematics

Inverse kinematics is the mathematical calculation for finding the parameters in forward kinematics, and by knowing the best value for each parameter in Table 1, the desired position and orientation of the robot is achieved. The derivation of inverse kinematics is a challenging mathematical task, especially when the angles are more than 2. In this study, the inverse kinematics is only to find 1 to 3. The parameter 4 is related to the end-effector, and FLC decides the end-effector motion based on the input from the proximity sensors attached to the robot system.

Inverse kinematics in this paper is calculated by an algebraic method to find the reverse of (3) and achieve θ_1 to θ_3. θ_1 to θ_3 calculation are given by (4), (5), and (6).

$$ C\theta_2 = \frac{px^2+py^2-a_1^2-a_2^2}{2a_1 2_2} \quad (4) $$

therefore,

$$ \theta_2 = atan2\left(\pm\sqrt{1-\left(\frac{px^2+py^2-a_1^2-a_2^2}{2a_1 2_2}\right)^2}, \frac{px^2+py^2-a_1^2-a_2^2}{2a_1 2_2}\right) \quad (5) $$

The angle created by the motion link 1 relative to the base (θ_1) and the third link (θ_3) are

$$ \theta_1 = atan2\left(p_x^2+p_y^2-(a_2 cos\theta_2+a_1)^2 a_2 cos\theta_2+a_1\right) \quad (6) $$
$$ \theta_3 = \pm atan2\left(p_x^2+p_y^2-(a_2 cos\theta_{12}+a_1)^2 a_2 cos\theta_{12}+a_1\right) $$

2.2. Fuzzy logic controller design

FLC was introduced by Lofti Zadeh as a controller for a plant. Fuzzy sets are the logic based on the input coming from sensors the fuzzy logic derived from the way human thinks instead of the exact mathematical modeling. The rules-based for FLC application is the probability of possible conditions that might occur in the system. In this study, the rules are set based on the position of a target relative to proximity sensors detection as the input for the system, and the rules-based govern the robot whether to grab the target, start to move, or stay in idle position. The rules-based set in study is shown in Table 2, where

inputs are the detection data from proximity sensors (sensors' installation positions are shown in Figure 2) and the robot base angle (θ1) shown in Figure 3(a). The more rules set, the smoother gripper motion is.

The membership function chosen in this study is given by

$$\mu = \begin{cases} 0 & if\ x \le \alpha \\ (x-\alpha)/(\beta-\alpha) & if\ \alpha \le x \le \beta \\ 1 & if\ x \ge \beta \end{cases} \quad (7)$$

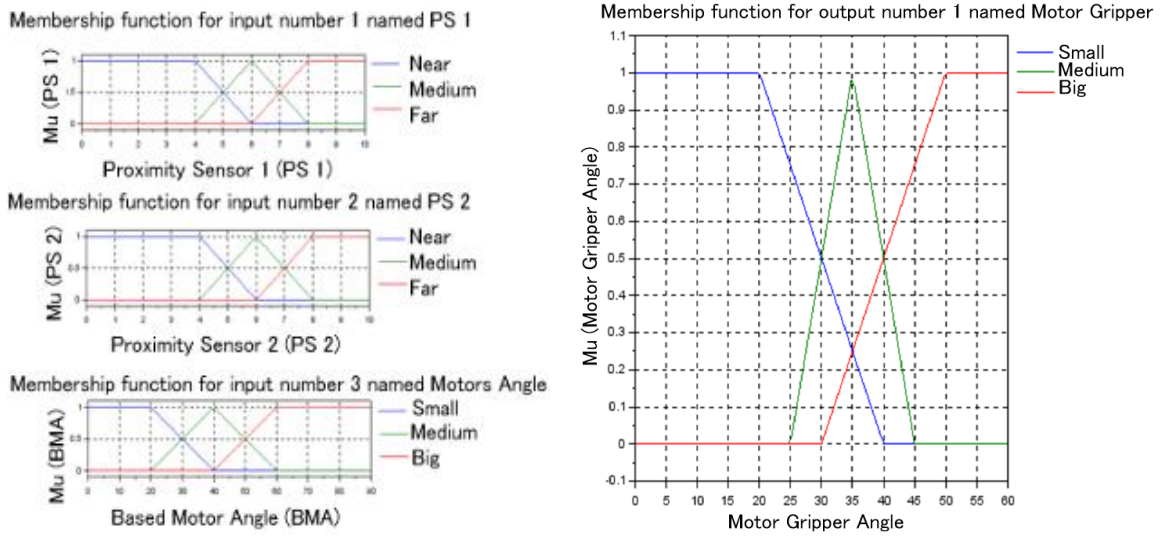
where α and β is the upper and lower boundary indicating "Near" and "Far" distance between the tomato (target) and the gripper. α is between 0 to 5 cm, and β is in the range of 11 to 20 cm.

Based on the place of installation, the proximity sensors rules are divided into rules for proximity sensor 1 and sensor 2, as shown in Figure 2(a). The first proximity sensor is attached to the picking place to indicate the availability of tomato to be grabbed by the robot and placed on belt-conveyor. Proximity sensor 1 rules are in "Near" or "Far"; "Near" is when the distance is ≤ 5 cm, and "Far" is > 5cm. Proximity sensor 1 output data is related in robot motion in picking and placing a tomato. The rules for proximity sensor 2 are set to be "Near," "Medium," and "Far." "Near" is set when the proximity sensor detects the target to be 0 to 5 cm from the robot. "Medium" is set in the range 6 to 10 cm and "Far" is in 11 to 20 cm. The data from proximity sensor 2 is related to the servo motor rotation and gripper angle (θ_4) whether it should be "Small", "Medium", and "Big".

The rules in Table 2 are converted into the membership function as shown in Figure 4. Membership inputs are shown in Figure 4(a) and output (gripper motion) is given in Figure 4(b). FLC outputs are the possibilities of gripper's final motion in grabbing the tomato, and are in term of "Small," "Medium," and "Big." These terms related to the degree (angle) of servo motors rotation and robot motions. "Small" means the servo motor's degree in the range of 20o to 25o, and the robot is in the state of picking and placing the target or manipulated object. "Medium" is when the servo rotation in a range of 25o to 25o, and the robot is in initial motion or starting to pick and place the target. "Big" is when the servo rotation in the range of 37o to 60o and the robot is in standby mode. The elaboration of how to relate the inputs to the output is by fuzzification shown in Figure 5, and how the input in Figure 5(a) can cause the motion in Figure 5(b).

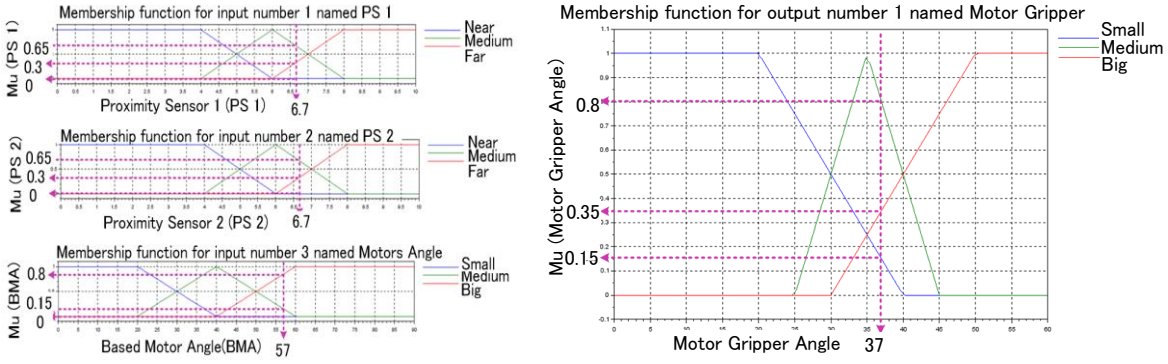
Table 2. Rules based fuzzy logic to controller 4DOF arm robot manipulator

NO	INPUT		OUTPUT	
	Proximity Sensor1	Proximity Sensor2	Base Angle(θ_1)	Gripper (θ_4)
1	Near	Near	Small	Small
2	Near	Near	Medium	Small
3	Near	Near	Far	Small
4	Near	Medium	Small	Small
5	Near	Medium	Medium	Medium
6	Near	Medium	Far	Medium
7	Near	Far	Small	Medium
8	Near	Far	Medium	Medium
9	Near	Far	Far	Medium
10	Medium	Near	Small	Medium
11	Medium	Near	Medium	Medium
12	Medium	Near	Far	Medium
13	Medium	Medium	Small	Medium
14	Medium	Medium	Medium	Medium
15	Medium	Medium	Far	Medium
16	Medium	Far	Small	Medium
17	Medium	Far	Medium	Medium
18	Medium	Far	Far	Big
19	Far	Near	Small	Big
20	Far	Near	Medium	Big
21	Far	Near	Far	Big
22	Far	Medium	Small	Big
23	Far	Medium	Medium	Big
24	Far	Medium	Far	Big
25	Far	Far	Small	Big
26	Far	Far	Medium	Big
27	Far	Far	Far	Big



(a) (b)

Figure 4. Membership function of input and output,
(a) Input membership function, (b) Output membership function



(a) (b)

Figure 5. Input and output fuzzification, (a) Input membership function, (b) Output membership function

3. RESULTS AND ANALYSIS

Data collecting method in this study is divided into two steps, arm robot motion data; taken using inverse kinematics to detect the precise position of tomato in Cartesian coordinates frames, and Gripper data; taken by calculating how much the servo rotation angles produced during picking and placing tomatoes. The arm-robot positions in Cartesian coordinates frames are converted into the angles of every robot joints (θ_1, θ_2, and θ_3) in Figure 3a, and the calculation of θ_1, θ_2, and θ_3 is given in eq. 4 to 6. The initial position of the arm robot manipulator is shown in Figure 3(b). Table 3 presents the robot positions reference in Cartesian coordinates frame (x, y, and z-axis) as the input to kinematics analysis to calculate θ_1, θ_2, and θ_3, and the actual robot positions in x, y, and z-axis By inverse kinematics calculation, arm robot position relatives to the target and the angles of every joint are acquired by knowing the length of links and other mechanical parameters presented in Table 1. The inverse kinematics results are compared to the real data from experiments to show the effectiveness of the inverse kinematics analysis in this study. Tables 4 and 5 show the error comparison between the data from inverse kinematics and the actual data taken from the experiment. The error is calculated by

$$\% \text{ Error } = \frac{\text{Data from Inverse Kinematics}}{\text{Actual Data from Experiment}} \times 100\% \quad (8)$$

Table 3. Input and output positions of arm robot manipulator

No	Input Cartesian (cm)			Output Cartesian (cm)			Orientation Angle (°)		
	x	y	z	x	y	z	θ_1	θ_2	θ_3
1	9	2	0	8.9	2.1	0	0	74.6	53.4
2	9	3	0	9.1	3.2	0	0	86.5	59.9
3	9	4	0	9.1	3.8	0	0	100.4	67
4	9	5	1	9	4.9	1.1	6.34	118.4	76
5	9	6	1	8.9	6.1	1.1	6.34	121.5	80
6	10	4	1	10.2	3.9	1.2	5.71	91.8	59.1
7	10	4	5	10.1	3.9	5.1	26.6	91.8	59.1
8	11	4	5	11	3.9	4.9	24.4	84.9	52,1
9	12	4	5	12.1	4	5.2	22.7	79	45.5
19	10	4	5	9.9	4	5	26.6	91.8	59.1

Table 4. Arm robot error positions in Cartesian coordinates

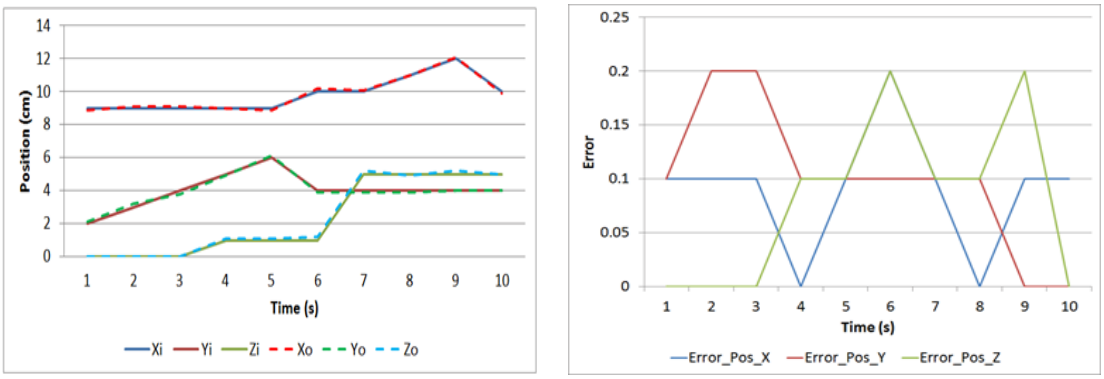
No	Input			Output			Error			Error Percentage		
	x	y	z	x	y	z	x	y	z	x	y	z
1	9	2	0	8.9	2.1	0	0.1	0.1	0	1.1%	5%	0%
2	9	3	0	9.1	3.2	0	0.1	0.2	0	1.1%	6.7 %	0%
3	9	4	0	9.1	3.8	0	0.1	0.2	0	1.1%	5%	0%
4	9	5	1	9	4.9	1.1	0	0.1	0.1	0%	2%	10%
5	9	6	1	8.9	6.1	1.1	0.1	0.1	0.1	1.1%	1.6%	10%
6	10	4	1	10.2	3.9	1.2	0.2	0.1	0.2	2%	2.5%	20%
7	10	4	5	10.1	3.9	5.1	0.1	0.1	0.1	1%	2.5%	2%
8	10	4	5	11	3.9	4.9	0	0.1	0,1	0%	2.5%	2%
9	10	4	5	12.1	4	5.2	0.1	0	0.2	0,8%	0%	4%
10	10	4	5	9.9	4	5	0.1	0	0	1%	0%	0%
Error Average										0.1%	0.1%	0.04%

Table 5. Error percentage between the angles achieved from inverse kinematics and experiment

No	Inverse Kinematics Prediction Angle (^{o})			Output From Experiments Angle (^{o})			Error Percentage Angle (^{o})		
	θ_1	θ_2	θ_3	θ_1	θ_2	θ_3	θ_1	θ_2	θ_3
1	0	74.6	53.4	0	74.6	53.4	0	0	0
2	0	86.5	59.9	0	86.5	59.9	0	0	0
3	0	100.4	67	0	100.4	67	0	0	0
4	6.34	120	76	6.34	118.4	76	0	1.4%	0
5	6. 34	120	80	6.34	121.5	80	0	1.25%	0
6	5.71	90	60	5.71	91.8	59.1	0	2%	1.5%
7	26,6	90	60	26.6	91.8	59.1	0	2%	1.5%
8	24.4	84.9	52.1	24.4	84,9	52.1	0	0	0
9	22.7	79	45.5	22.7	79	45.5	0	0	0
10	26,6	91.8	59.1	26.6	91.8	59.1	0	0	0
Error Average							0%	0.7%	0.3%

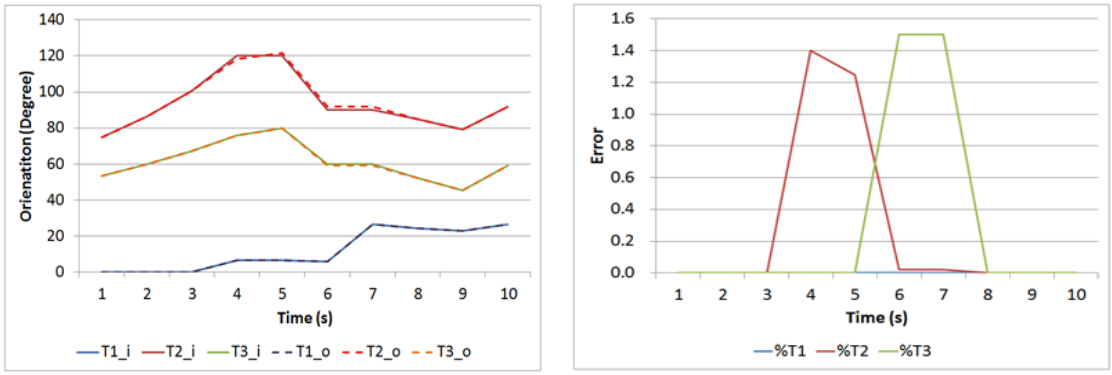
The comparison of robot references and actual positions are shown in Figure 6(a) where X_i, Y_i, and Zi are the positions input to inverse kinematics, and X_0, Y_0, and Z_0 are the actual position. The errors between the references and actual positions are shown in Figure 6(b) where Error_Pos_X, Error_Pos_Y, and Error_Pos_Z are the error position in *x*, *y*, and *z*-axis respectively. Table 4 and Figure 6 indicate that the errors are very small (0.1%, 0.1%, and 0.04% in x, y, and z-axis respectively).

The comparison of robot's orientation between the predicted by inverse kinematics and actual orientation from the experiment is shown in Figure 7(a) where $T1_i$, $T2_i$, and $T3_i$ are θ_1, θ_2, and θ_3 reference inputs respectively, and $T1_0$, $T2_0$, and $T3_0$ are the actual θ_1, θ_2, and θ_3 achieved from the experiment. The error between the predicted and actual orientation is given in Figure 7 where %T1, %T2, and %T3 are the error in θ_1, θ_2, and θ_3, respectively. Table 5 and Figure 7 show that the errors are very small (0%, 0.7%, and 0.3% for θ_1, θ_2, and θ_3, respectively). The errors in robot positions and orientations occur due to the servo gears do not allow the motor to rotate precisely and the rounding calculation that unavoidable to the limited microprocessor memory.



(a) (b)

Figure 6. The comparison of robot's references and actual,
(a) Comparison of robot position references and actual, (b) Comparison error



(a) (b)

Figure 7. The comparison of robot's orientation between the predicted by inverse kinematics and actual orientation achieved by experiment, (a) Comparison of robot orientation references and actual, (b) Comparison error

Ten experiments were conducted to compare the arm robot error position in Cartesian coordinates and angle of orientation achieved from inverse kinematics and experiments. In experiment 1, the Cartesian reference input in x, y, and z-axis are 9, 2, and 0, respectively. This coordinates position is the target to be achieved by arm robot manipulator. The real positions from experiments are 8.9, 2.1, and 0 for x, y, and z-axis and 0°, 74.6°, and 53.4° for θ_1, θ_2, and θ_3, respectively. The error found in this experiment is ≤ 0.1 cm.

In experiment 5, the Cartesian reference input in x, y, and z-axis are 9, 6, and 1, respectively. This coordinates position is the target to be achieved by arm robot manipulator. The real positions from experiments are 8.9, 6.1, and 1.1 for x, y, and z-axis and 6.34°, 121.5°, and 80° for θ_1, θ_2, and θ_3, respectively. The error found in this experiment is 0.1 cm. In experiment 7, the Cartesian reference input in x, y, and z-axis are 10, 4, and 5, respectively. This coordinates position is the target to be achieved by arm robot manipulator. The real positions from experiments are 10.1, 3.9, and 5.1 for x, y, and z-axis and 26.6 °, 91.8 °, and 59.1° for θ_1, θ_2, and θ_3, respectively. The error found in this experiment is 0.1 cm.

In experiment 9, the Cartesian reference input in x, y, and z-axis are 12, 4, and 5, respectively. This coordinates position is the target to be achieved by arm robot manipulator. The real positions from experiments are 12.1, 4, and 5.2 for x, y, and z-axis and 22.6°, 79°, and 45.5° for θ_1, θ_2, and θ_3, respectively. The error found in this experiment is ≤ 0.2 cm. In experiment 10, the Cartesian reference input in x, y, and z-axis are 10, 4, and 5, respectively. This coordinates position is the target to be achieved by arm robot manipulator. The real positions from experiments are 9.9, 4, and 5 for x, y, and z-axis and 26.5°, 91.8°, and 59.1° for θ_1, θ_2, and θ_3, respectively. The error found in this experiment is ≤ 0.1 cm.

As presented in Table 2, FLC is responsible for rotating the motor to move the gripper (end-effector). The artificial intelligence is necessary to create robot motion in picking and placing tomato. The defined smoothness is including controlling the force applied to the robot to ensure the robot do not damage tomatoes as the targets in this study. Table 6 shows the comparison between FLC rules-based in Table 2 and the experimental results data. If the angle of the motor gripper is between 0^o-25^o is the angle, then the robot is gripping the tomato. If the angle of motor gripper is between 25^o-33^o is the angle, then the robot is starting to move or initial motion. If the angle of the motor gripper is between 37^o-60^o is the angle then the robot is in the standby position. 60^o is the position when the gripper is in full open, and 0^o is the angle when the gripper is completely closed.

In sample 1; If Proximity sensor1: Near AND Proximity sensor2: Near AND Motor1 angle: Small then Gripper: Picking and Placing (Small). This condition is where the gripper is 20^o open or in rules-based is Small. In sample 2; If Proximity sensor1: Near AND Proximity sensor2: Far AND Motor1 angle: Medium then Gripper: Initial Motion (Medium). This condition is where the gripper is 27^o open or in rules-based is Medium. In the sample 3; If Proximity sensor1: Medium AND Proximity sensor2: Far AND Motor1 angle: Medium then Gripper: Initial Motion (Medium). This condition is where the gripper is 33^o open or in rules-based is Medium. In sample 4; If Proximity sensor1: Far AND Proximity sensor2: Far AND Motor1 angle: Small then Gripper: Initial Motion (Medium). This condition is where the gripper is 60^o open or in rules-based is Medium. The input of FLC and the data taken from experiment in Table 6 show that the gripper angle (θ_4) or output data is following the rules set in FLC input shown in Table 2. The difference is in data no 10 of Table 6, and this condition might occur due to the size of the tomato. Figure 8 shows the experiment of picking and placing tomatoes, from the tomatoes pool to the belt-conveyor shown in Figure 2. Figure 8(a) shows the robot in initial motion; where it is taking the green tomatoes, Figure 8(b)-(e) show the process of picking and placing tomatoes, and Figure 8(f) shows the robot in the standby position. The experimental results show that the proposed method to design arm robot manipulator is effective proven by the small errors in position and orientation of robot achieved by simulation design and real application. The designed robot is cost-effective and more accessible to apply by any farmer.

Table 6. Comparison between the input FLC and the experiment results of θ_4

o	Input Proximity Sensor 1	Input Proximity Sensor 2	Input Rules Based	End Effector Angle(θ_4)in^o
1	Detected	0-5 cm	Small	20^o-25^o(Small)
2	Detected	0-5 cm	Small	20^o-25^o(Small)
3	Detected	0-5 cm	Small	20^o-25^o(Small)
4	Detected	6-10 cm	Small	20^o-25^o(Small)
5	Detected	6-10 cm	Medium	25^o-33^o(Medium)
6	Detected	6-10 cm	Medium	25^o-33^o(Medium)
7	Detected	11-20 cm	Medium	25^o-33^o(Medium)
8	Detected	11-20 cm	Medium	25^o-33^o(Medium)
9	Detected	11-20 cm	Medium	25^o-33^o(Medium)
10	No detection	0-5 cm	Big	37^o-60^o(Big)
11	No detection	0-5 cm	Big	37^o-60^o(Big)
12	No detection	0-5 cm	Big	37^o-60^o(Big)
13	No detection	6-10 cm	Big	37^o-60^o(Big)
14	No detection	6-10 cm	Big	37^o-60^o(Big)
15	No detection	6-10 cm	Big	37^o-60^o(Big)
16	No detection	11-20 cm	Big	37^o-60^o(Big)
17	No detection	11-20 cm	Big	37^o-60^o(Big)
18	No detection	11-20 cm	Big	37^o-60^o(Big)

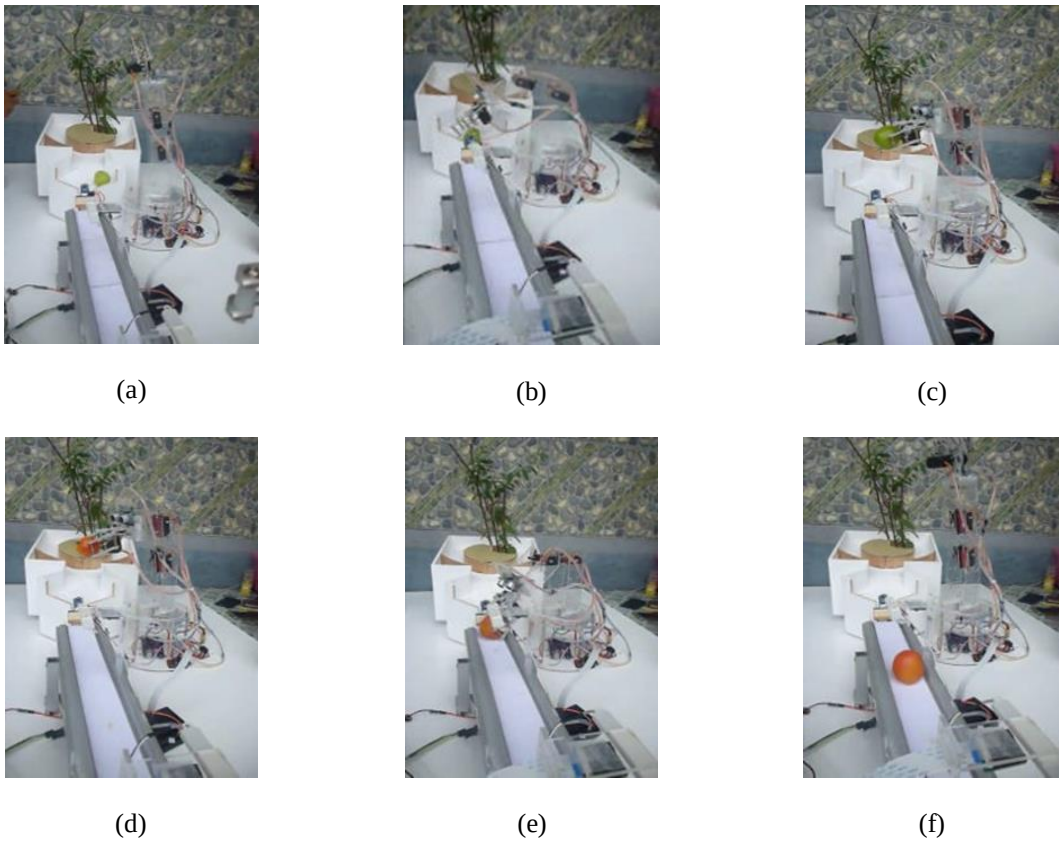


Figure 8. Robot picking and placing green and red tomato from initial motion and standby, (a) Initial motion, (b) Picking and placing, (c) Picking and placing, (d) Picking and placing, (e) Picking and placing, (f) Standby

4. CONCLUSION

The digital farming application allows efficient and effective harvesting assisted by a robot. The current technology makes robot production cost cheaper, and easier for the farmer to apply it. Robot trajectory generation can be achieved by inverse kinematics analysis. Inverse kinematics analysis produces the position and orientation of a robot as the trajectory references. The end-effector in this study is a gripper that suitable to grab the harvested tomatoes (target). Gripper motion is designed by FLC to increase motion smoothness and the amount of force applied to the target to avoid excessive power in grabbing the target. Ten experiments were conducted to show the effectiveness of the proposed method. The experimental results show that the actual robot position achieved from experiments are similar to the reference inputs from inverse kinematics analysis. The position errors are 0.1%, 0.1%, and 0.04% in x, y, and z-axis respectively, and orientation errors are 0%, 0.7%, and 0.3% for θ_1, θ_2, and θ_3 respectively. The errors in robot positions and orientations occur due to the servo gears do not allow the motor to rotate precisely, and the rounding calculation that unavoidable to the limited microprocessor memory. By comparing the FLC rules and experiment data, it shows that the gripper motions also follow the rules from FLC.

REFERENCES

[1] M. Stoelen, K. Krzysztof, V. F. Tejada, N. Heiberg, C. Balaguer, and A. Korsaeth, "Low-Cost Robotics for Horticulture: A Case Study on Automated Sugar Pea Harvesting," *10th European Conference on Precision Agriculture (ECPA)*, 2015. DOI: 10.3920/978-90-8686-814-8 34.

[2] T. T. Nguyen, E. Kayacan, J. De Baedemaeker, W. Saeys, *"Task and Motion Planning for Apple Harvesting Robot," IFAC Proceedings Volumes*, Vol. 46, Issue. 18, pp. 247-252, 2013. https://doi.org/10.3182/20130828-2-SF-3019.00063.

[3] G. Wang, Y. Yu, and Q. Feng, "Design of End-Effector for Tomato Robotic Harvesting," *IFAC-PapersOnLine*, Vo. 49, Issue 16, pp. 190-193, 2016. https://doi.org/10.1016/j.ifacol.2016.10.035

[4] W. Ji et al.,"Automatic Recognition Vision System Guided for Apple Harvesting Robot Computers and Electrical Engineering," *Computers & Electrical Engineering*, Vol. 38, No. 5, pp. 11861195, 2012. https://doi.org/10.1016/j.compeleceng.2011.11.005
[5] C. Xiong, C. Xiaoman, X. Zou X, "A Method for Identification and Matching of the Picking Point for Mature Litchi under Structural Environment," *Journal of Applied Biotechnology & Bioengineering*, Vol. 3, Issue 6, pages. 4, 2017. Retrieved: http://medcraveonline.com.
[6] S.S. Mehta and T. F. Burks, "Vision-Based Control of Robotic Manipulator for Citrus Harvesting," *Computers and Electronics in Agriculture*, Vol. 102, pp.14658, 2014. http://dx.doi.org/10.1016/j.compag.2014.01.003.
[7] J. Li, M. Karkee, Q. Zhang, K. Xiao, and T. Feng, "Characterizing Apple Picking Patterns for Robotic Harvesting", *Computers and Electronics in Agriculture*, Vol. 127, pp.633640, 2016. http://dx.doi.org/10.1016/j.compag.2016.07.024.
[8] P. M. Blok, R. Barth, and W. van den Berg, "Machine Vision for a Selective Broccoli Harvesting Robot," *IFAC-PapersOnLine*, Vol. 49, Issue. 16, pp. 6671, 2016. http://dx.doi.org/10.1016/j.ifacol.2016.10.013.
[9] G. Feng, C. Qixin, and M. Nagata, "Fruit Detachment and Classification Method for Strawberry Harvesting Robot," *International Journal of Advanced Robotic Systems*, Vol. 5, Issue 1, pp. 4148, 2008. https://doi.org/10.5772/5662.
[10] T. Dewi, P. Risma, Y. Oktarina, and M. T. Roseno, "Perbandingan Desain dan Pemodelan Tangan Robot Lima Jari sebagai Sistem Underactuated (In Bahasa)," *Annual Research Seminar 2016*, Vol 2, No. 1, pp. 139-145, 2016.
[11] M. Ceccarelli and E. Ottaviano, "Kinematic Design of Manipulators," *Robot Manipulators, Marco Ceccarelli (Ed*.), InTechopen, pp. 49-72, 2008. ISBN: 978-953-7619-06-0.
[12] H. Lin, Y. Liu, and Y. Lin, "Intuitive Kinematic Control of a Robot Arm via Human Motion," in *Proceeding of 37th National Conference on Theoretical and Applied Mechanics (37th NCTAM 2013) & The 1st International Conference on Mechanics (1st ICM), Procedia Engineering*, vol. 79, pp. 411-416, 2014.
[13] A. B. Rehiara, "Kinematics of Adept Three Robot Arm," *Robot Arms, Prof. Satoru Goto (Ed.)*, pp. 21-38, InTechopen, 2011, ISBN: 978-953-307-160-2.
[14] P. Donelan, "Kinematic Singularities of Robot Manipulators," *Advances in Robot Manipulators, Ernest Hall (Ed.)*, InTechopen, pp. 401-416, 2010. ISBN: 978-953-307-070-4.
[15] W. Song, and G. Hu, "A Fast Inverse Kinematics Algorithm for Joint Animation," in *Proceeding of 2011 International Conference on Advances in Engineering, Procedia Engineering*, vol. 24, pp. 350-354, 2011.
[16] T. Balkan, M. K. Ozgoren, and M. A. S. Arikan, "Structure Based Classification and Kinematic Analysis of Six-Joint Industrial Robotic Manipulators," *Industrial Robotics: Theory, Modelling and Control, Sam Cubero (Ed.)*, InTechOpen, pp. 149-184, 2006. ISBN: 3-86611-285-8
[17] S. Zodey, and S. K.Pradhan, "Matlab Toolbox for Kinematic Analysis and Simulation of Dexterous Robotic Grippers," *Procedia Engineering*, Vol. 97, pp. 1886-1895, 2014. https://doi.org/10.1016/j.proeng.2014.12.342
[18] S. Kucuk and Z. Bingul, "Robot Kinematics: Forward and Inverse Kinematics," *Industrial Robotics: Theory, Modelling, and Control, Sam Cubero (Ed.), InTechopen*, pp. 117-148, ISBN: 3-86611-285-8.
[19] I. A. Sultan "Inverse Position Procedure for Manipulators with Rotary Joints," *Industrial Robotics: Theory, Modelling and Control, Sam Cubero (Ed.), InTechopen*, pp. 185-210, 2006. ISBN: 3-86611-285-8.
[20] 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. doi:10.1155/2010/984823.
[21] A. Mohammed, B. Schmidt, L. Wang, and L. Gao, "Minimizing Energy Consumption for Robot Arm Movement," *Procedia CIRP*, Vol. 25, pp. 400-405, 2014. https://doi.org/10.1016/j.procir.2014.10.055.
[22] A. El-Sherbiny, M. A. Elhosseini, and A. Y.Haikal, "A comparative study of soft computing methods to solve inverse kinematics problem," *Ain Shams Engineering Journal*, Vol. 9, No. 4, 2018, pp. 2535-2548, https://doi.org/10.1016/j.asej.2017.08.001.
[23] M. M. U. Atique, M. R. I. Sarker, and M. A. R. Ahad, "Development of an 8DOF quadruped robot and implementation of Inverse Kinematics using Denavit-Hartenberg convention," *Heliyon*, Vol. 4, No. 12, e01053, 2018, 0 https://doi.org/10.1016/j.heliyon.2018.e01053.
[24] S. Nurmaini, "Intelligent Navigation in Unstructured Environment by using Memory-Based Reasoning in Embedded Mobile Robot," *European Journal of Scientific Research*, Vol. 72, No. 2, pp. 228-244, 2012.
[25] S. Nurmaini, B. Tutuko, K. Dewi, V. Yuliza, and T. Dewi, "Improving posture accuracy of non-holonomic mobile robot system with variable universe of discourse," *TELKOMNIKA Indonesian Journal of Electrical Engineering*, Vol 15, No. 3, pp. 1265-1279, 2017. http://dx.doi.org/10.12928/telkomnika.v15i3.6078.
[26] T. Dewi, P. Risma, and Y. Oktarina, "Fuzzy Logic Simulation as Teaching-Learning Media for Artificial Intelligence Class," *Journal of Automation, Mobile Robotics & Intelligent Systems*, Vol. 12, No. 3, pp. 59-65, 2018. DOI: 10.14313/JAMRIS 4-2013/59.
[27] S. Nurmaini and F. Setianingsih, "Enhancement of The Fuzzy Control Response with Particle Swarm Optimization in Mobile Robot System," 2018 *International Conference on Electrical Engineering and Computer Science (ICECOS)*, Pangkal Pinang, 2018, pp. 73-78.doi: 10.1109/ICECOS.2018.8605221.
[28] A. Duka, "ANFIS based Solution to the Inverse Kinematics of a 3DOF Planar Manipulator," *8th International Conference Interdisciplinary in Engineering, INTER-ENG 2014, Procedia Technology*, Vol. 19, pp. 526-533, 2015. https://doi.org/10.1016/j.protcy.2015.02.075.