Robotic technology towards industry 4.0: Automatic object sorting robot arm using kinect sensor

Esa Apriaskar[1,*], Fahmizal[2], M R Fauzi[2]

[1]Department of Electrical Engineering, Universitas Negeri Semarang, Semarang 50229, Indonesia
[2]Department of Electrical Engineering and Informatics, Vocational College, Universitas Gadjah Mada, Yogyakarta, Indonesia

[*]Corresponding author's e-mail: esa.apriaskar@mail.unnes.ac.id

Abstract. As one of the most commonly employed robots in manufacturing business, robot arm has to develop into a technology capable of successing the industry 4.0 revolution. A robot arm that can interact well with humans to solve a problem becomes one of the encouraging research topics. It requires the ability to behave like humans which causes a robot arm should be utilizing intelligent peripherals. This study aims to create a robot arm design that uses a Kinect sensor as a smart peripheral to approach the human ability to see an object. As a case study, a color-based object sorting simulation was conducted. The automatic movement that the robot used is based on inverse kinematics. The experimental test shows that average percentage errors for end-effector's position in x, y and z coordinates are 5.83%, 5.89% and 8.59%, respectively. This result has led the robot arm equipped with the Kinect sensor managed well to sort and move objects based on the color.

1. Introduction

Industry 4.0 revolution has brought a new spirit in industrial machine technology throughout the world. Robots, which are mechanical devices that can perform physical tasks, both using supervision or autonomous, has proved to help human activities, starting from health service [1], mapping [2], to industrial applications [3]. They have advantages that are not possessed by humans, like doing a repetitive job precisely, and never feel tired. Robots can also be reprogrammed to make it available performing several different tasks. Given many potential benefits that the robots can contribute to humans, it confirms that robotic technology provides an essential role in this industry 4.0 revolution [4], [5].

A robot arm, which is a mechanical system used in manipulating the movement of lifting, moving, and manipulating an object to ease social work [6], can be one of the exciting objects for robotic technology development. It is very popular and most straightforward used in the manufacturing process in the form of an anthropomorphic robot arm because it has the advantage in terms of flexibility in two-dimension work space. Thus, it is suitable for applications in most industrial robots.

Several researchers have contributed to the development of robot arm technology. Most of them focused in control, modelling and movement techniques. For example, Faidallah, et.al [7] developed the robot arm from the control and modeling side by using electromyography and flex sensors. Kubota, et.al [8] placed more emphasis on controlling the robot arm based on multi-joint arm viscoelasticity and operator theory. Saafi et.al [9] focused on movement technique by proposing

forward kinematic model of a spherical parallel manipulator. Heading on supporting the fourth industrial revolution, development work on a robot arm technology for manufacturing industry is also an attractive topic. The idea of a dual-arm robot platform design integrated with mobile robots was put forward in [10]. Guo, et.al [11] tried to make robot arm prototype that had been combined with a mobile robot, while Kim, et.al [12] attempted to demonstrate the use of dual-arm robots for case studies in the IT part packaging industry.

The robot arm design presented in this paper is expected to be one of extended developments in robotic technology for manufacturing process. Cutting-edge manufacturing solutions that help the success of industry 4.0 rely on the development of robotic technology [13]. Robots utilization in industry has changed from merely replacing the role of humans in carrying out a task in the production process to collaborative work between machines and humans to make a manufacturing process run more productive [14]. It requires intelligent peripherals to support the worker in making decisions. This work proposes a robot arm design utilized with Kinect sensor as a smart periphery to help humans in object sorting, which is one of established industrial applications. The robot arm uses inverse kinematics calculation to move the robot.

The remainder of this paper is organized as follows. In Section 2, the mechanical design of the robot arm is described together with the method used in this paper. Section 3 explains the experimental setup for conducting the functional test of the robot arm. The following part aims to present results and discussion that show the merit of the proposed design and method. It also elaborates how the robot arm was examined. Section 5 concludes the result and contribution of the paper.

2. Design and Method

2.1. Mechanical Design

The robot arm proposed in this paper has four joints that represents its degree of freedom (DOF), as shown in figure 1. The joint structure of the robot arm consists of a *base*, *shoulder*, *elbow*, and *gripper*. Each joint is controlled and driven by a servo motor. A servo drives the *base* to rotate horizontally. Two servo motors that are facing each other actuate the *shoulder*. The *elbow* is also driven by a servo motor that has the same rotating axis as the *shoulder*. Because the *elbow* must rotate on the same shaft with the *shoulder*, a gear mechanism was made between the *elbow* and the *shoulder*, as presented in figure 2.

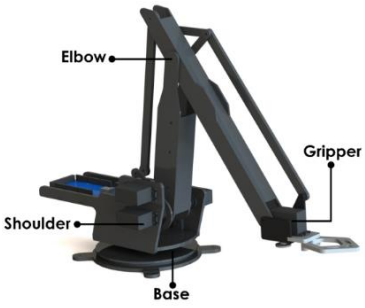


Figure 1. Four joints of the 4-DOF robot arm

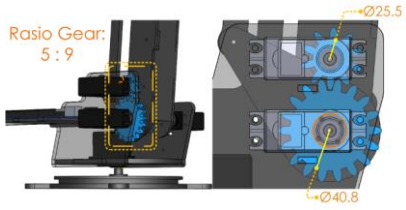


Figure 2. Gear mechanism of the robot arm

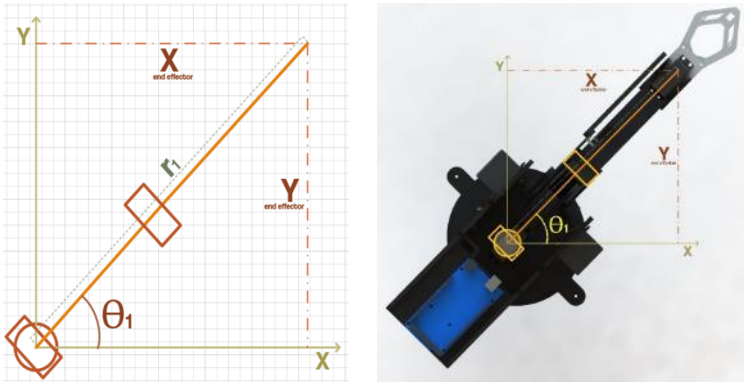
2.2. Inverse Kinematics

Inverse kinematics is a calculation to find the angle variables (*joint*) of a robot in determining the position and orientation of the *end-effector* [6]. The Pythagorean law and trigonometric rules can help to solve inverse kinematic settlement by looking at two sides, particularly the top view and the side view of the robot arm structure, as shown in figures 3 and 4. The top picture is used to find the degree of angle θ_1, which is a vertical rotating axis. The side is used to find the degree of angle θ_2 between

the *shoulder* and *base* joint, and the degree of angle θ_3 between the *elbow* joint. Equation (1) helps to calculate the degree of angle θ_1. X_{ef} and Y_{ef} are the *end-effector* positions on the x and y coordinates, respectively.

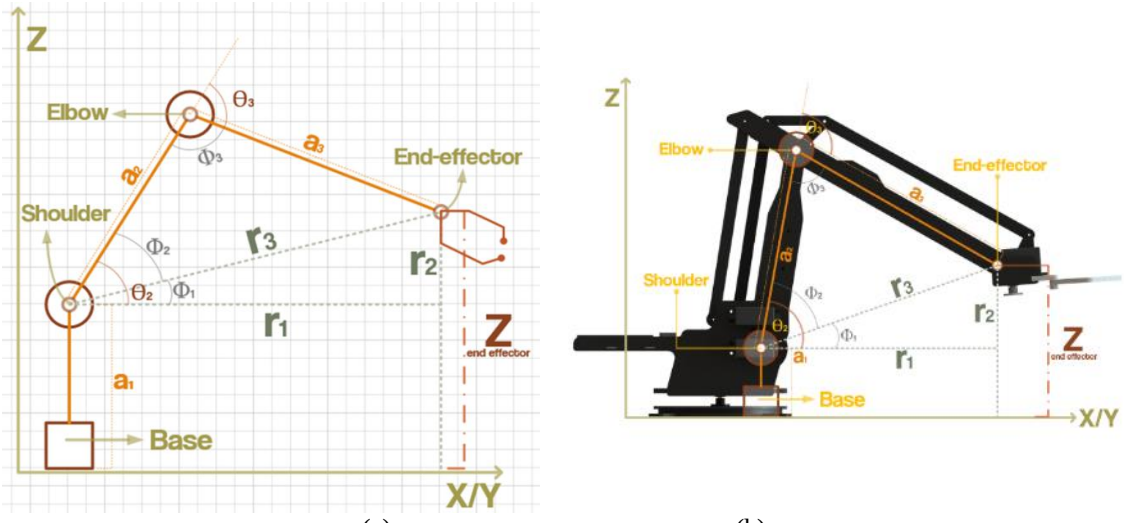
$$\theta_1 = \tan^{-1}\left[Y_{ef} / X_{ef} \right] \tag{1}$$

Furthermore, by looking at the side of the 4-DOF robot arm structure, values of θ_2 and θ_3 can be obtained. From the side, we can see several variables including a_1, a_2, a_3, Z_{ef}, θ_2 and θ_3. a_1 is the link length between the base and the shoulder, while a_2 is the link between the shoulder and the elbow. a_3 denotes the link length between the elbow and the *end-effector*. Z_{ef} represents the *end-effector* coordinates on the Z-axis. The horizontal axis of the shoulder joint and link between the shoulder and the elbow form the angle θ_2, while the link a_3 and extension of the link a_2 constructs angle θ_3.



(a) (b)

Figure 3. Top view of the robot arm shows (a) the joints assemble angle θ_1, and (b) the actual illustration



(a) (b)

Figure 4. Side view of the robot arm shows (a) the joints assemble angle θ_1 and θ_2, and (b) the actual illustration

To obtain the degree of angle θ_2 and θ_3, we need a triangle as highlighted in figure 4(a). The triangle is each initialized with variables r_1, r_2, and r_3 on each side. r_1 denotes horizontal distance of *shoulder* joint to *end-effector*. When viewed from the top side of 4-DOF robot arm structure, a Pythagorean triangle between the X_{ef} and the Y_{ef} forms r_1. r_2 expresses the difference between the Z_{ef}

and the link a_1 (the link between the base and *shoulder* joints), while F is obtained from the Pythagorean triangle between r_2 and r_3. The values of r_1, r_2 and r_3 can be collected from equations (2) - (4). Thus, those parameters lead to obtaining auxiliary angles Φ_1, Φ_2, and Φ_3, as shown in equations (5) - (7). These supporting angles help to determine the value of θ_2 and θ_3 using equations (8) - (9).

$$r_1 = \sqrt{(Xef)^2 + (Yef)^2} \tag{2}$$

$$r_2 = Zef - a_1 \tag{3}$$

$$r_3 = \sqrt{(r_2)^2 + (r_1)^2} \tag{4}$$

$$\Phi_1 = tan^{-1}[r_2 / r_1] \tag{5}$$

$$\Phi_2 = cos^{-1}\left[(a_2)^2 + (r_3)^2 - (a_3)^2 \Big/ 2a_2 r_3 \right] \tag{6}$$

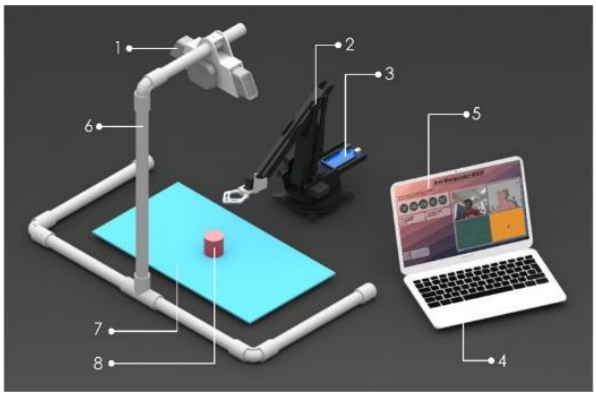
$$\Phi_3 = cos^{-1}\left[(a_2)^2 + (a_3)^2 - (r_3)^2 \Big/ 2a_2 a_3 \right] \tag{7}$$

$$\theta_2 = \Phi_1 + \Phi_2 \tag{8}$$

$$\theta_3 = 180^\circ - \Phi_3 \tag{9}$$

3. Experimental Setup

This work utilizes Kinect camera as an image capturing tool of an object. The information of the picture is in the form of depth and RGB (Red, Green, and Blue) data of the object. The personal computer takes a role in image processing using the OpenCV-based Processing IDE. Arduino Mega 2560 as the main controller of the robot arm receive information from the personal computer before delivering a command to every joint in the sorting object mission. Figure 5 depicts the overall system design of this work.



Description:
1. Kinect Camera
2. 4-DOF Arm Robot
3. Arduino Mega 2560
4. Personal Computer
5. GUI Processing IDE
6. Kinect Camera Buffer
7. Workspace Desk
8. Tested Object

Figure 5. Components for the experimental setup

The desired x, y and z coordinates were processed using inverse kinematics calculation to get the servo angle for each joint. Afterward, the Processing IDE send these values to Arduino IDE through serial communication. Arduino IDE send these values according to the servo pins connected to the

Arduino Mega 2560, then the value of servo angles were sent by Arduino Mega 2560 via a direct connection using a USB cable. Arduino Mega 2560 drove each servo following the magnitude of the earned angle value. The following step is 4-DOF robot arm reaching the x, y and z coordinates of the object, gripped the object using the gripper, and move the object to its default place or the desired place.

4. Result and Discussion

4.1. The workspace of the Robot Arm

Workspace is the total volume that is possible to reach for *end-effector* of the robot arm. The determination of the range of a workspace is regarding x, y, and z coordinates test data. The test was conducted by comparing the reference trajectory and the measured position of the *end-effector* after being moved based on inverse kinematics calculation. Figure 6(a) explains that in x-axis coordinate, the *end-effector* could reach maximum length at 51 cm from the position of robot arm base and had a good response for 15 through 50 cm. The minimum point that the robot could reach is in 8 cm, despite it has considerable relative error. Points from 0 to 7 cm are inaccurate positions for the *end-effector*. It is because the link was too long while the distance of the object was too close to the robot arm, so the *end-effector* could not touch objects at those ranges. The opposite state also happened for the point of more than 51 cm. The length of the link was too short for the object while it was too far away from the robot arm, so the robot could not strike objects that are more than 51 cm. The same pattern also happened for y-axis coordinate, as shown in figure 6(b). Based on the experimental test for z-axis coordinate, the maximum height that could be attained by the robot arm was 26 cm in a diameter of 30 cm, as depicted in figure 6(c). The height between 0 to 10 cm was the most optimal points that robot's *end-effector* could accurately reach out. In addition, these measurement test also informs that average percentage errors for end-effector's position in x, y and z coordinates are 5.83%, 5.89% and 8.59%, respectively. The most acceptable percentage error for x and y coordinates is laying between 15 to 50 cm, while for z coordinate, it is from 0 to 10 cm.

Based on the inverse kinematic test for x, y, and z coordinates, the workspace area of the robot arm can be clearly defined. Figure 6(d), and 7(e) illustrates the workspace of the robot arm from the side view, and top view, respectively. It indicates that the robot arm is not able to pick or put an object outside of the workspace. The closer the position of the object to the edge of the workspace will influence the accuracy of the robot arm.

4.2. Square-shaped Trajectory Tracking

Inverse kinematics testing using trajectory has been explained in the previous section. It actually aimed to determine the accuracy of the kinematics behind the robot arm. Another test was conducted by replacing the robot arm's *end-effector* gripper into an *end-effector* pointer (using a pen) as the tip because it could better visualize the position of the *end-effector* rather than using gripper, as shown in figure 7(a).

The test was started by making a square-shaped path. The paths were using coordinates x, y in (0, 20), (0, 35), (15, 35), and (15, 20). Figure 7(b) shows the result of trajectory testing. The position of the *end-effector* coordinates x, y (0, 20) was indeed accurate since it was the starting point of the *end-effector*. However, for other coordinates, it shows that the result was still less accurate. It might happen because the coordinates were close to the maximum diameter of the workspace of the robot arm. The good thing that this test proffers is that the *end-effector* can return to the beginning position with several looping tests. It ensures that although the robot arm has less accuracy for points near the maximum diameter of the workspace, it still has good repeatability.

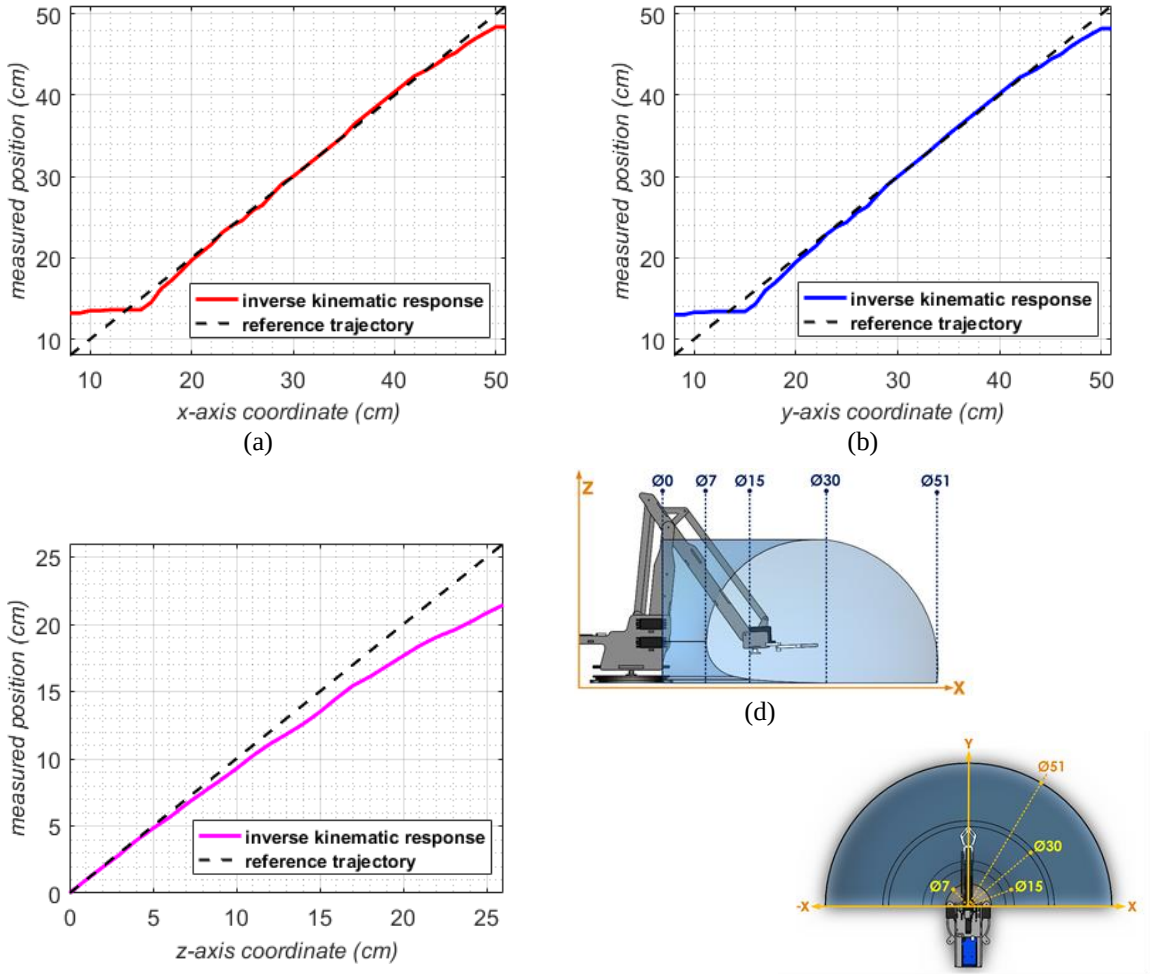
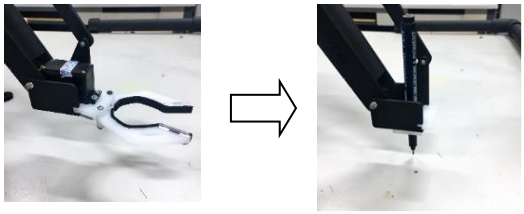


Figure 6. Inverse kinematic test for (a) x-axis, (b) y-axis, and (c) z-axis coordinates generates the workspace area of the robot arm in (d) side view, and (e) top view

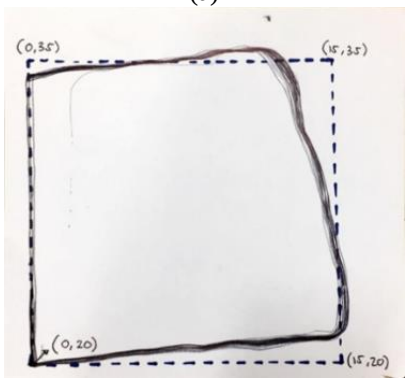
4.3. Color Sorting and Picking Test

In this test, a series of robot arm operation was performed to classify objects according to the color of the object. This operation used all the components previously described in figure 5. First, the robot arm was in the default position. This test managed (0, 15, 0) cm in x, y, and z coordinates, as the default position. That position was the state where the robot arm was standby before getting the command to move. The Kinect sensor above the workspace scanned and defined the colors of the objects on the workspace desk.

On a personal computer, the GUI Processing IDE operated to give the command to the robot arm. Figure 8 shows the display of the GUI Processing IDE. The user could choose the color of the object that they wanted to store by clicking on the color in the GUI. The Color Detection column would display the selected color. Near with the Color Detection column, there is also a column containing the RGB data of the chosen object. There is a coordinate column which includes the coordinates where the selected object is. Then the robot arm started to take the object after the operator pressed the ENTER button. The *end-effector* of the robot arm would go to the coordinates where the object was placed. After the robot arm's *end-effector* reach the intended coordinates, the gripper will grab the object and move the object to the storage area according to its color type. In this system, there are four colors can be selected and sorted, namely blue, red, yellow, and green. Figure 9 is the illustration when the robot arm had successfully sorted the object based on the color.



(a)



(b)

Figure 7. (a) Changing the gripper of the *end-effector* to a pointer, (b) The result of the trajectory tracking using the determined inverse kinematic calculations

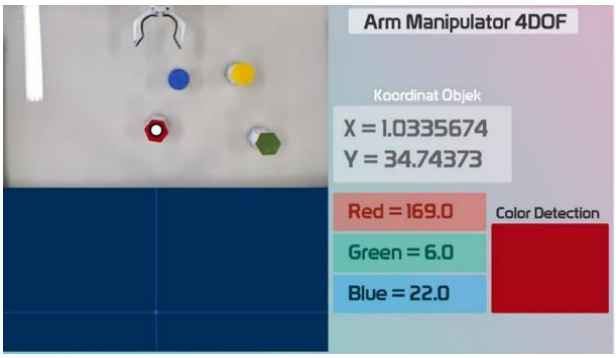


Figure 8. The GUI Processing IDE for color detection

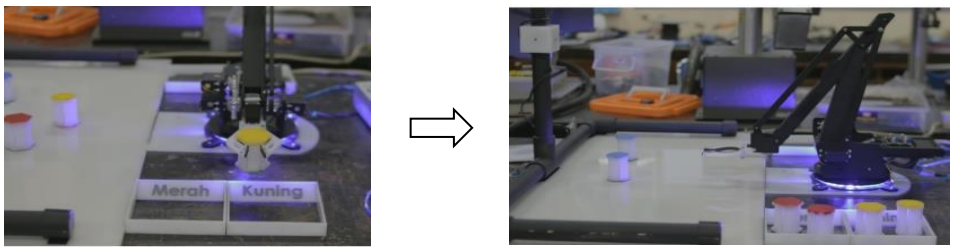


Figure 9. The robot had succeeded to sort objects based on the color

5. Conclusion

Finally, this work contributes to present a design of automatic object sorting robot arm to encourage the fourth industrial revolution. The proposed design has succeeded to take object sorting mission based on color using a Kinect sensor. The concept of inverse kinematic has completed the robot arm's ability to pick up and move the object autonomously. This paper also finds that the workspace of the

robot arm affects the accuracy of the *end-effector* in trajectory tracking. However, this finding does not give any adverse effect on the success of object sorting mission since the robot worked inside the optimal points of the workspace. The study on optimizing the workspace to give the best performance in path tracking, more precisely in object sorting, can be a possible future work of this paper.

References

[1] Ö. Selvi and S. Yavuz, "Dimensional optimisation of a 3-DoF spherical parallel manipulator for rehabilitation by using firefly algorithm," *Int. J. Mech. Robot. Syst.*, vol. 4, no. 2, p. 107, 2018

[2] E. Apriaskar, Y. P. Nugraha, and B. R. Trilaksono, "Simulation of Simultaneous Localization and Mapping Using Hexacopter and RGBD Camera," in *International Conference on Automation, Cognitive Science, Optics, Micro Electro-Mechanical System, and Information Technology*, 2017, pp. 48–53

[3] T. Li *et al.*, "Development of a 5-DOF arc welding robot," *Int. J. Mech. Robot. Syst.*, vol. 3, no. 2/3, p. 129, 2016

[4] S. Grahn, B. Langbeck, K. Johansen, and B. Backman, "Potential Advantages Using Large Anthropomorphic Robots in Human-robot Collaborative, Hand Guided Assembly," in *6th CIRP Conference on Assembly Technologies and Systems (CATS)*, 2016, vol. 44, pp. 281–286

[5] G. Wisskirchen *et al.*, "Artificial Intelligence and Robotics and Their Impact on the Workplace," London, England, 2017

[6] K. M. Lynch and F. C. Park, *Modern Robotics: Mechanics, Planning, and Control*. Cambridge: Cambridge University Press, 2017

[7] E. M. Faidallah, Y. H. Hossameldin, S. M. Abd Rabbo, and Y. A. El-Mashad, "Control and modeling a robot arm via EMG and flex signals," in *15th International Workshop on Research and* Education *in Mechatronics (REM)*, 2014, pp. 1–8

[8] S. Kubota, M. Deng, and Y. Noge, "Robot arm control based on human multi-joint arm viscoelasiticity and operator theory," in *2018 International Conference on Advanced Mechatronic Systems (ICAMechS)*, 2018, pp. 132–136

[9] H. Saafi, M. A. Laribi, S. Zeghloul, and M. Arsicault, "Forward kinematic model of a new spherical parallel manipulator used as a master device," *Int. J. Mech. Robot. Syst.*, vol. 3, no. 2/3, p. 145, 2016

[10] D. Il Park, C. Park, H. Do, T. Choi, and J. Kyung, "Development of dual arm robot platform for automatic assembly," in *2014 14th International Conference on Control, Automation and Systems (ICCAS 2014)*, 2014, no. Iccas, pp. 319–321

[11] H. Guo, K.-L. Su, K.-H. Hsia, and J.-T. Wang, "Development of the mobile robot with a robot arm," in *2016 IEEE International Conference on Industrial Technology (ICIT)*, 2016, vol. 2016-May, pp. 1648–1653

[12] G. Kim, J. Park, T. Choi, H. Do, D. Park, and J. Kyung, "Case studies of a industrial dual-arm robot application," in *2017 14th International Conference on Ubiquitous Robots and Ambient Intelligence (URAI)*, 2017, pp. 301–302

[13] M. Gereald and Z. Peter, "Industrial Robots Meet Industry 4 . 0," *Hadmérnök*, vol. 4, no. December 2017, pp. 230–238, 2017

[14] V. M. Pawar *et al.*, "Manufacturing Robotics: The Next Robotic Industrial Revolution," 2016

Acknowledgments

The authors would like to thank Engineering Faculty of Universitas Negeri Semarang for supporting this work. We also would like to thank Control and Instrumentation laboratory of Department of Electrical and Informatics Engineering, Vocational College, Universitas Gadjah Mada, for bolstering the experiment.