Black Sea Journal of Engineering and Science
doi: 10.34248/bsengineering.1445455

Open Access Journal
e-ISSN: 2619 – 8991

Research Article
Volume 7 - Issue 3: 436-444 / May 2024

ROBOTIC ARM TRAJECTORY TRACKING USING IMAGE PROCESSING AND KINEMATIC EQUATIONS

Yusuf Hamida EL NASER[1]*, Durmuş KARAYEL[1], Mert Süleyman DEMİRSOY[1], Muhammed Salih SARIKAYA[1], Nur Yasin PEKER[1]

[1]Sakarya Applied Science University, Department of Mechatronics Engineering, 54050, Sakarya, Türkiye

Abstract: This study presents a novel approach for scanning and obtaining object contours within a workspace using a camera and subsequently following these 2-D contours to another region with the assistance of an end effector. The process begins with the determination of boundary lines of object images through advanced image processing methodologies. Subsequently, a conversion from camera pixel coordinates to robot metric coordinates is performed to facilitate trajectory planning for the robotic arm. The trajectory of the robot arm, and consequently, the path followed by the end effector, is determined based on these metric coordinates. The article provides a detailed exposition of the methodology, including the mathematical background and experimental study, showcasing the efficacy and accuracy of the proposed approach in real-world scenarios. This research contributes to the advancement of robotic systems capable of precise trajectory tracking and manipulation tasks in diverse applications. This study involves an innovative approach that combines theoretical and practical methods, including object contour detection using image processing, conversion of pixel coordinates obtained from the camera to robot metric coordinates, robot kinematics, and programming steps.

Keywords: Image processing, Robot arm, Inverse kinematics, Trajectory tracking, Smart servos

***Corresponding author:** Sakarya Applied Science University, Department of Mechatronics Engineering, 54050, Sakarya, Türkiye
E mail: yusufelnaser@sakarya.edu.tr (Y. H. EL NASER)

Yusuf Hamida EL NASER	https://orcid.org/0000-0003-4757-6288
Durmuş KARAYEL	https://orcid.org/0000-0001-9326-466X
Mert Süleyman DEMİRSOY	https://orcid.org/0000-0002-7905-2254
Muhammed Salih SARIKAYA	https://orcid.org/0000-0002-2809-9896
Nur Yasin PEKER	https://orcid.org/0000-0003-2468-2627

Received: March 01, 2024
Accepted: March 27, 2024
Published: May 15, 2024

Cite as: El Naser YH, Karayel D, Demirsoy MS, Sarıkaya MS, Peker NY. 2024. Robotic arm trajectory tracking using image processing and kinematic equations. BSJ Eng Sci, 7(3): 436-444.

1. Introduction

Robots can be considered as a complex mechanism consisting of a certain number of limbs, rotating and / or sliding joints, actuators that drive these joints and an end effector attached to the last limb (Sophokleous et al., 2021). In this way, they repeatedly follow the trajectories taught by preprogramming and perform certain processes. Scientific studies in the field of robotics continue their development by integrating with many disciplines such as semiconductors, artificial intelligence, power electronics, computer vision, programming, communication, control, mechatronic systems, material science and such (Romero-Gonzalez et al., 2022). From past to present, robot arms, mobile robots, humanoid robots, spider robots, medical robots, and even robots used for space tasks are being integrated with new technologies every day and are offered for use to make life easier for humanity.

When the studies in the field of robotics are examined in the literature, it can be seen that the studies have increased especially in the last few decades as a natural result of the popularity of the subject. Theoretical studies on robot kinematics and dynamics (El Naser et al., 2020) purposes like pick&place operations (Dixon, 2007), drawing (Cheah et al., 2006), painting and cutting (Grzelczvk et al., 2019), studies for object recognition, sectors such as health & medicine (Dyck et al., 2013), manufacturing (Luo et al., 2019), entertainment shows that the topic is becoming widespread (Liu et al., 2014). In addition to the industrial examples related to use of robots given above, prototypes of the experimental realization of the studies carried out to shed light on the theoretical background of similar objectives also stand out (He et al., 2021).

Since the subject is quite convenient for multidisciplinary studies, it is possible to come across colloborations with artificial intelligence and optimization, control, image processing and many other research areas (Mahmoodpour et al., 2019; Budivanta et al., 2020; Duque et al., 2022). These collaborations have not only inspired scientific articles, but have also allowed the development of original patent ideas.

The aim of this study can be summarized as follows: to scan and obtain the contours of objects on the workspace using a camera, and then to follow these 2-D contours to another region on the workspace with the help of an end effector. The boundary lines of the object image has determined using image processing methodologies and the conversion from camera pixel coordinates to robot metric coordinates has performed. These metric coordinates have been used to determine the trajectory of the robot arm, thereby the path followed by the end effector has been obtained. Detailed methodology, mathematical background, and experimental study has been described in the rest of the article.

2. Materials and Methods

In the study, a 4-axis SCARA type robot was prototyped in the PRRR sequence. Unlike the standard SCARA configuration, the sliding joint was used on the first axis in order to compensate the negative torque caused by the motor, limb and tool weights. Dynamixel AX-12A and MX-28T DC smart servo motors were used as robot actuators and TTL communication of the devices was decisive in the selection of these motors. The images of the objects located in the workspace was taken with the help of the Logitech C270 USB camera and the edge detection was performed by isolating them from the environment by means of image processing techniques. Instead of dealing with camera calibration, simple numerical regression method was used in transition from camera pixel coordinates to robot metric coordinates. Similarly, the obtained points, which will determine the trajectory of the robot arm, were reproduced by numerical interpolation and the robot movement was performed much more smoothly (Fang et al., 2022).

Thus, uniqueness was obtained compared to similar studies in the literature (Uk et al., 2020). The designed robot can follow the coordinates of the object with the help of forward and inverse kinematic calculations and perform trajectory tracking by the end effector. At the same time, by utilizing other end effectors, it can transport the lines of the object to another region on the workspace and draw it in 2-D. The forward and inverse kinematic equations of the robot arm are detailed in the rest of the sections (Chen et al., 2007). While forward kinematics equations are expressed using Denavit - Hartenberg approach, inverse kinematics equations has solved by geometric approach and detailed visually (Jiao et al., 2018; Zhang et al., 2022; Jian-Peng et al., 2023). Programming of all these processes including image processing, coordinate transformations and interpolation operations, kinematic calculations and computer - actuator communications had realized using LABVIEW 2016 software via USB2Dynamixel control card hardware. Figure 1 shows the USB card and USB2Dynamixel control card. Figure 2 shows the flow chart of the system.

Figure 1. USB camera and USB2Dynamixel control card.

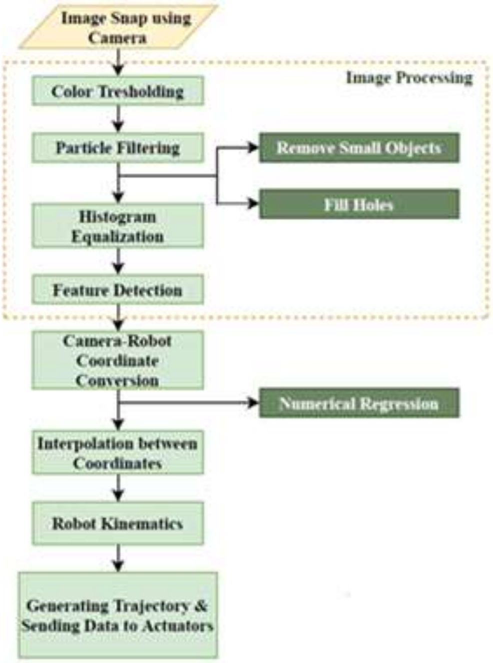


Figure 2. Flowchart of the system.

2.1. Image Processing

A number of image processing steps need to be performed in order to correctly determine the contours of the image taken by camera. These contours will be used to get the trajectory of the end effector driven by the robot arm later on. In this regard, firstly color thresholding procedure was applied to the image taken and the object in the image was tried to be separated from the ground on which it is located (Zhou et al., 2005; Syahrian et al., 2017). Pixel values that are not among the threshold values determined for this are drawn to zero. Thresholding in the HSL (hue-saturation-luminance) color space for the appropriate threshold value was performed according to the following ranges; 0<hue<184, 0<sat<255, 0<lum<29.

Values may vary depending on ambient lightings, the shading, colors and transparency of objects. Since the isolated pixels are still affected by environmental noise, they need to be suppressed. For this purpose, the object in the workspace was completely separated from its background by passing through a series of stages such as particle filtering, hole filling, removal of small objects and histogram equalization. Thanks to this, the image has been made suitable for edge detection. When feature detection algorithms were tested, it was seen that the ideal result for the object in the working space was provided by the Harris algorithm. The determined points were subjected to regression and stored in an array to be converted into metric coordinates. Accordingly, the results obtained are expressed in the following images in Figure 3.

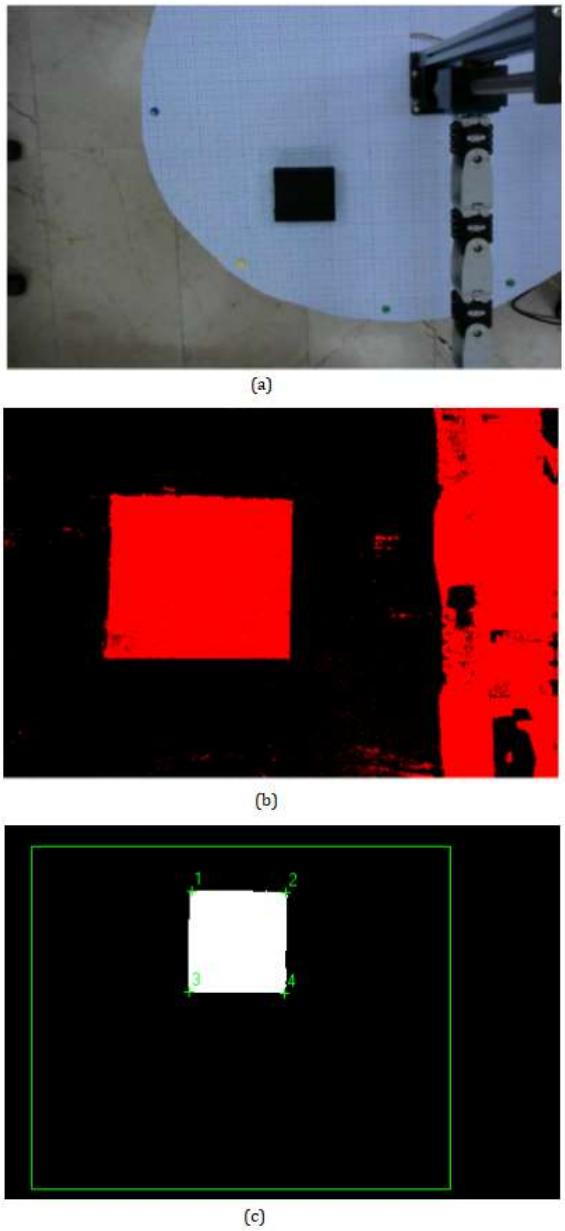


Figure 3. (a) Original image, (b) After color threshold, (c) After edge detection.

2.2. Conversion from Image Pixel to Robot Metric Coordinates

The x-y coordinates obtained from the corner points of the object using edge detection are in pixels and the origin of this coordinate system is the upper left corner of the image. However, the robot arm operates in the metric system and the origin of the coordinate system is located on the first axis of the robot. This should be taken into consideration when converting the found coordinates from pixels to millimeters. In the conversion process, the millimetric coordinates corresponding to the coordinates in pixels found by the feature extraction process were determined by measuring. Accordingly, the pixel coordinates of the points determined as a result of the feature extraction were expressed in pairs $\{x_p, y_p\}$ as the first point $\{651.3, 556.7\}$, second point $\{777.0, 559.0\}$, third point $\{648.2, 680.9\}$ and the fourth point $\{768.8, 683.8\}$. The corresponding millimetric coordinates are shown in pairs {xm, ym} respectively, as the first point $\{36, 146\}$, second point $\{36.5, 217\}$, third point $\{106, 146\}$ and the fourth point $\{107, 216.5\}$. Since there is a 90° difference between the pixel and the milimetric coordinates, x_p values should be converted to y_m values and y_p values to x_m. To find the millimetric x-y coordinates, the following equations 1 and 2 can be written;

$$x_m = my_p + n \tag{1}$$

$$y_m = kx_p + l \tag{2}$$

Two cross corner coordinates are substituted in the equations twice and four equations with four unknowns are solved to find m, n, k and l values the formulation is shown in equations 3a, 3b, 3c,3d and 3e;

$$36 = m.556.7 + n \tag{3a}$$

$$106 = m.680.9 + n \tag{3b}$$

$$146 = k.648.2 + l \tag{3c}$$

$$216.5 = k.768.8 + l \tag{3d}$$

$$\begin{aligned} m = 0.563607\,, n &= -277.7600\,, k \\ &= 0.584577 \; and \; l \\ &= -232.923 \end{aligned} \tag{3e}$$

These values were found using a simple first order regression method. Thus, from pixel to millimeter coordinate transformation operations on both axes were performed with ease. Errors encountered due to the linear approach are discussed in the conclusion section.

2.3. Robot Kinematics

The concept of kinematics can be defined as a discipline that deals with the movements of the systems directly from the geometry, independent of dynamic effects such as force and torque (Tao et al., 2014; Staicu et al., 2018). In mechanisms such as robots, a direct relationship can be established between joint angles and the position of the end effector using kinematic analysis. In this way, trajectory tracking can be performed by modeling the arm movement correctly. Kinematics science is divided into two sub branches as forward kinematics and inverse kinematics (Yanto et al., 2017). Forward kinematics is used to find the end effector position corresponding to certain joint angles (Garriga-Casanovas and Baena, 2019). On the contrary, inverse kinematics is used to determine the joint angles corresponding to the end effector position and orientation (Wang et al., 2015; Nansai et al., 2021). In the study, the forward kinematics of the designed robot were analyzed by Denavit-Hartenberg method and the solutions of the inverse kinematics problem were obtained using geometric approach (Ding and Liu, 2018; Guzman-Gimenez et al., 2020).

2.3.1. Forward kinematics

In Denavit-Hartenberg notation, relationship of two adjacent limbs of the robot with each other and the characteristic of the joint between these two limbs are expressed with the help of a homogeneous transformation matrix shown in Equation 4 given below:

$$ {}^{i-1}_{i}T = \begin{bmatrix} c\theta_i & -s\theta_i & 0 & a_{i-1} \\ s\theta_i c\alpha_{i-1} & c\theta_i c\alpha_{i-1} & -s\alpha_{i-1} & -s\alpha_{i-1}d_i \\ s\theta_i s\alpha_{i-1} & c\theta_i s\alpha_{i-1} & c\alpha_{i-1} & c\alpha_{i-1}d_i \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (4) $$

Note that cθ and sθ represents cosθ and sinθ.

In general, rotation and translation directions of the joints are shown by the z axis, and the directions of the limb perpendicular to movement directions are indicated by an x axis. Then, y axes are determined according to the right hand rule. In Denavit-Hartenberg method, there are 4 joint parameters that constitute the transformation matrix, α_{i-1}, a_{i-1}, d_i and θ_i. The steps to be followed in determining these parameters are as follows:

- α_{i-1} ; distance from z_{i-1} to z_i along x_{i-1},
- d_i ; distance from x_{i-1} to x_i along z_i,
- α_{i-1} ; rotation angle from z_{i-1} ve z_i about x_i,
- θ_i ; rotation angle from x_{i-1} to x_i about z_i.

By multiplying these matrices respectively, the relationship between the first and the last limb of the robot is obtained. The illustration of the system and the D-H parameters table are given in Figure 4 and Table 1. The resulting matrix were written in the equation 5:

$$ {}^{0}_{5}T = \begin{bmatrix} r_{11} & r_{12} & r_{13} & p_x \\ r_{21} & r_{22} & r_{23} & p_y \\ r_{31} & r_{32} & r_{33} & p_z \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (5) $$

where equations 6a - 6l are;

$$ \mathrm{r}_{11} = c(\theta_2 + \theta_3 + \theta_4) \quad (6a) $$

$$ \mathrm{r}_{12} = s(\theta_2 + \theta_3 + \theta_4) \quad (6b) $$

$$ \mathrm{r}_{13} = 0 \quad (6c) $$

$$ \mathrm{r}_{21} = s(\theta_2 + \theta_3 + \theta_4) \quad (6d) $$

$$ \mathrm{r}_{22} = c(\theta_2 + \theta_3 + \theta_4) \quad (6e) $$

$$ \mathrm{r}_{23} = 0 \quad (6f) $$

$$ \mathrm{r}_{31} = 0 \quad (6g) $$

$$ \mathrm{r}_{32} = 0 \quad (6h) $$

$$ \mathrm{r}_{33} = 1 \quad (6i) $$

$$ \mathrm{p_x} = l_1 + l_2 c\theta_2 + l_3 c(\theta_2 + \theta_3) + l_4 c(\theta_2 + \theta_3 + \theta_4) \quad (6j) $$

$$ \mathrm{p_y} = l_2 s\theta_2 + l_3 s(\theta_2 + \theta_3) + l_4 s(\theta_2 + \theta_3 + \theta_4) \quad (6k) $$

$$ \mathrm{p_z} = d_1 \quad (6l) $$

Here, $\mathrm{p_x}$, p_y and p_z denote the position of the end effector in x, y, and z coordinates respectively. The term □1 refers to the displacement of the sliding joint on the 1st axis in the z direction depending on the motor angle θ_1. The linear displacement of the linear guide rail in the first axis is 5 mm per revolution. This formulated in equation 7.

$$ d_1 = \frac{5}{2\pi}\theta_1 \quad (7) $$

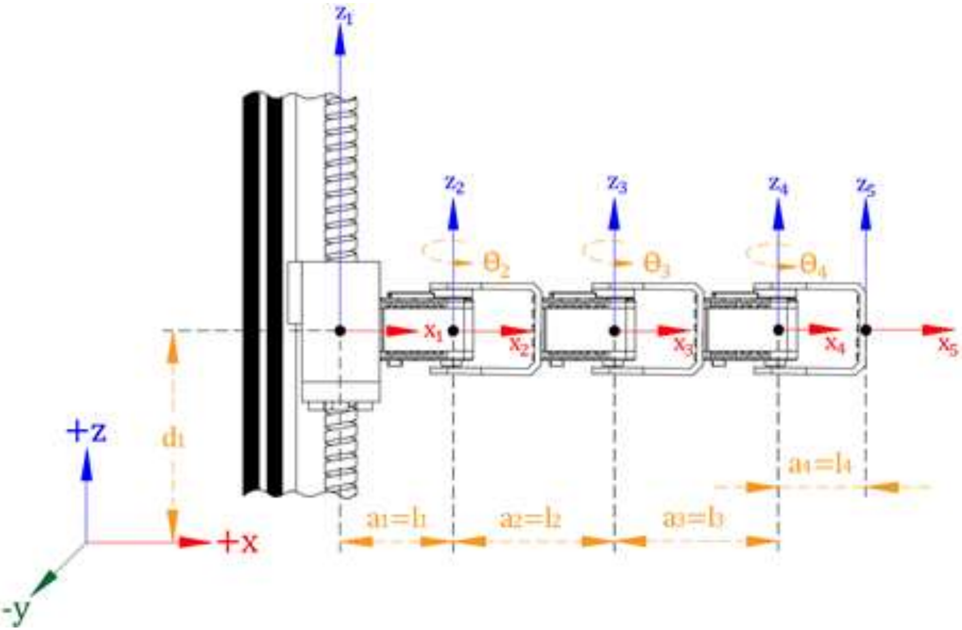


Figure 4. Illustration of D-H parameters of the system.

Table 1. D-H parameters for each axis

	α_{i-1}	a_{i-1}	d_i	θ_i
1st axis	0	0	d_1	0
2nd axis	0	a_1	0	θ_2
3rd axis	0	a_2	0	θ_3
4th axis	0	a_3	0	θ_4

5th axis	0	a_4	0	0

2.3.2. Inverse kinematics

The inverse kinematics problem is used to determine the joint variables d and θ from the position and configuration of the end effector. The joint variable data are processed in real time and sent to the actuators to ensure continuity of motion. To visualize point coordinates, limb lengths and joint angles used in the solution of inverse kinematics problem, the top view of the system is illustrated in Figure 5.

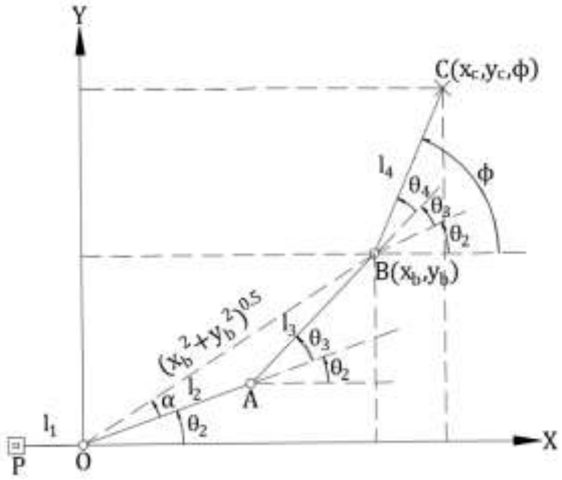


Figure 5. Illustration of inverse kinematics parameters.

The point P here represents linear rail guide on the 1st axis, the line segments |PO|, |OA|, |AB| and |BC| denotes limb lengths bounded by joints, and the point C represents the position of the end effector. For convenience in calculations, the origin of the x-y coordinate system is placed on the 2nd axis. Hence, the joint parameters to be calculated are θ_2, θ_3 and θ_4. The position (x_c, y_c) and orientation (ϕ) of the end effector are known and used in determining these parameters. Using the position and orientation of point C, the coordinates of point B are given below. x_b and y_b are formulated in Equations 8 and 9.

$$x_b = x_c - l_4 cos\phi \tag{8}$$

$$y_b = y_c - l_4 sin\phi \tag{9}$$

It is clear that the shortest distance between point O and B is equal in Equation 10;

$$OB| = \sqrt{x_b^2 + y_b^2} = \sqrt{x_c^2 + y_c^2 + l_4^2 - 2l_4(x_c cos\phi + y_c sin\phi)} \tag{10}$$

The angle $\widehat{OAB}$ in the figure is the suplementary angle of the angle θ_3, so it can be shown as π - θ3. Accordingly, the angle θ_3 can be found using the cosine theorem in the triangle of OAB, shown in Equation 11.

$$\theta_3 = \pi - \cos^{-1}(\frac{l_2^2 + l_3^2 - x_b^2 - y_b^2}{2l_2 l_3}) \tag{11}$$

To find the angle α, the cosine theorem is applied again in the imaginary triangle AOB, shown in Equation 12.

$$x_b^2 + y_b^2 + l_2^2 - 2\sqrt{x_b^2 + y_b^2} l_2 \cos\alpha = l_3^2 \tag{12}$$

and the angle α equals, shown in Equation 13.

$$\alpha = cos^{-1}(\frac{l_2^2 + x_b^2 + y_b^2 - l_3^2}{2l_2\sqrt{x_b^2 + y_b^2}}) \tag{13}$$

Obviously, the angle α+ θ2 can be represented as the coordinates of point B; formulated in Equations 14 and 15.

$$\alpha + \theta_2 = \tan^{-1}(\frac{y_b}{x_b}) \tag{14}$$

$$\theta_2 = \tan^{-1}(\frac{y_b}{x_b}) - \alpha \tag{15}$$

In Figure 5, it can be easily seen that the angle ϕ is equal to the sum of the 2nd, 3rd, and 4th joint angles. That means, shown in Equation 16.

$$\theta_4 = \phi - \theta_2 - \theta_3 \tag{16}$$

and ultimately, the relation between the angle θ_1 and the motion on the z axis were expressed in a linear relation as in Equation 17, and that yields,

$$\theta_1 = \frac{2\pi}{5} d_1 \tag{17}$$

3. Results and Discussion

The following experimental setup has been established to implement the system whose theoretical backgrounds were given in the sections above. Accordingly, the robot arm system, human-PC interface and parts of the program blocks were presented in Figure 6.

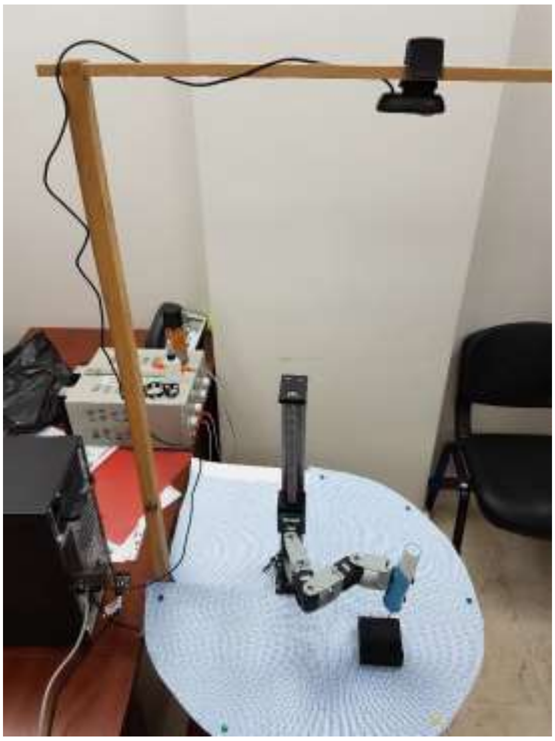


Figure 6. The experimental setup of robot arm system.

The camera was attached on the top of the robot arm to display the object. All actuators used in the system were communicated with each other and the computer via USB2Dynamixel control card. A 0-30V DC power supply were used to energize the motors. Figure 7 and Figure 8 shows the control tools; image, point and angle indicators.

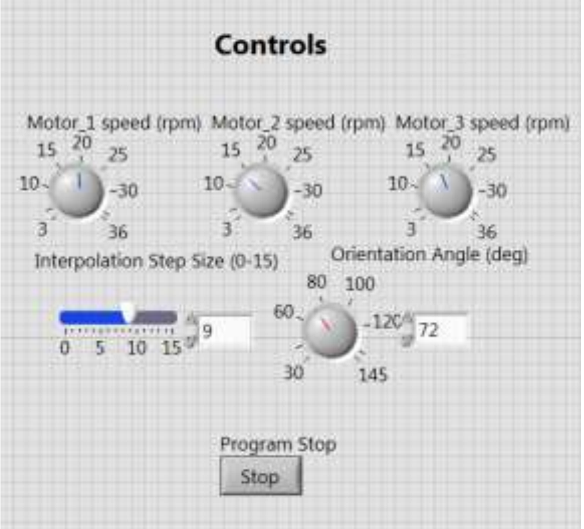


Figure 7. Control tools.

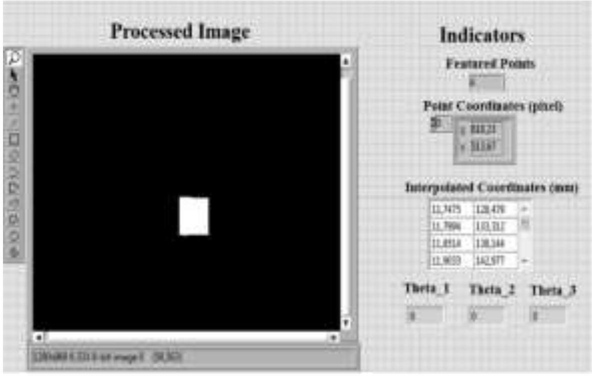


Figure 8. Image, point and angle indicators.

The data I/O on the human-PC interface are shown in the Figure 8. The processed image, feature coordinates, interpolated coordinates and motor angle data were displayed on the screen to inform the user. The user is able to interfere with motor speeds, orientation angle and integration steps in real time.

Sections of the program blocks were given below. In Figure 9, image processing steps were visualized and in Figure 10, interpolation, kinematic calculations and data transmission to actuators were shown.

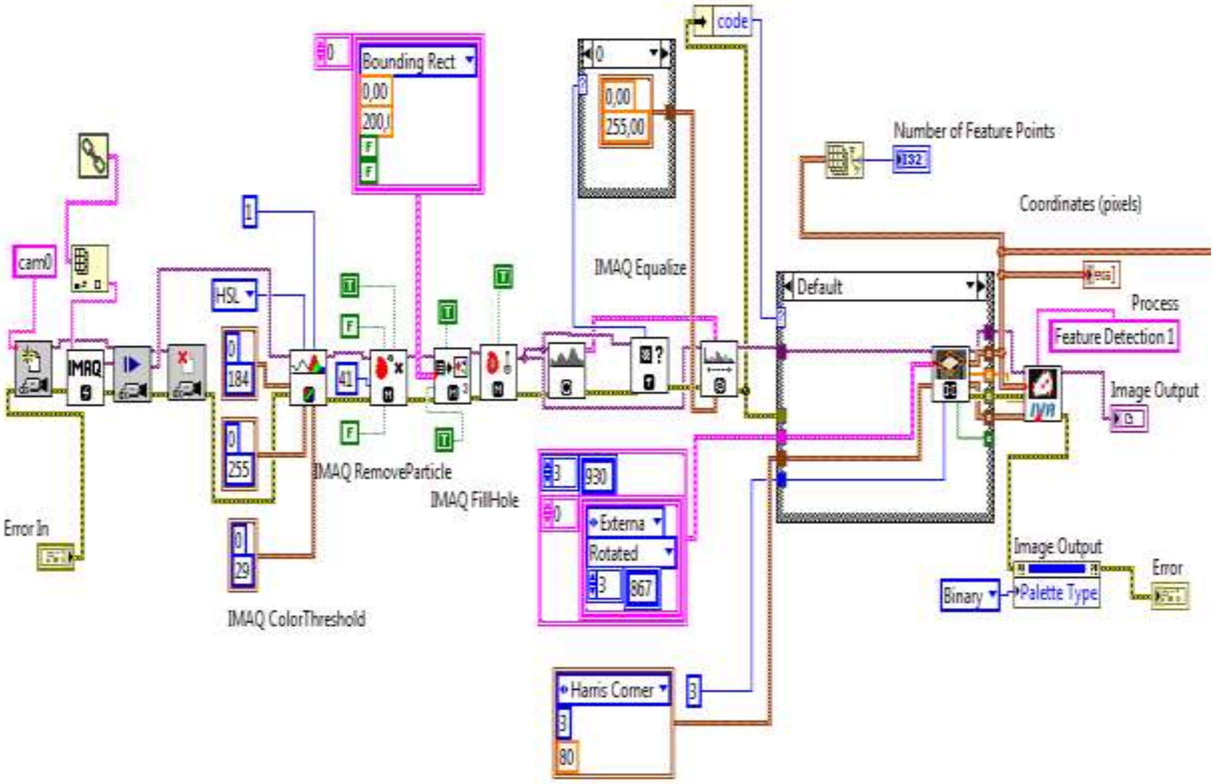
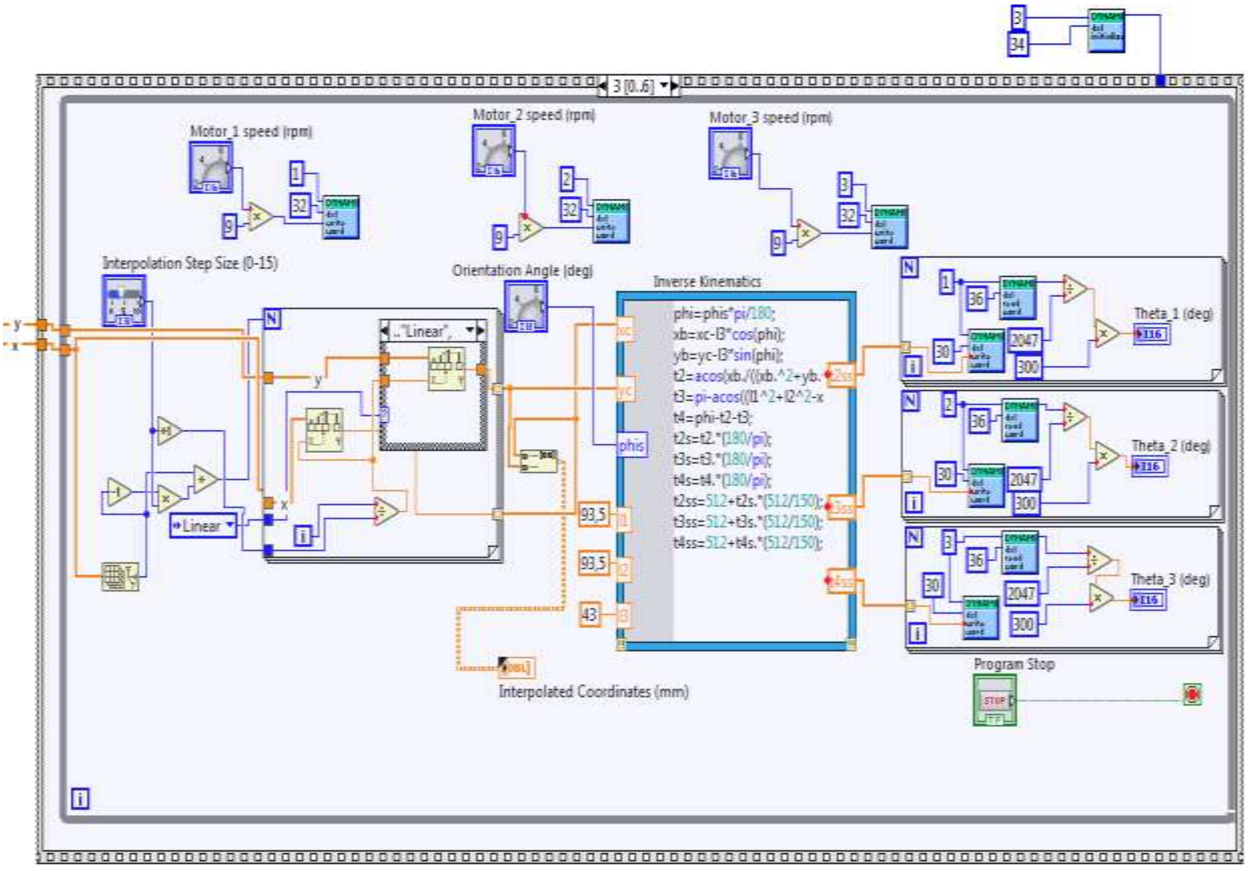


Figure 9. Image processing blocks.

The graph in Figure 11 illustrates the positional error measured in millimeters, between the actual coordinates of the tracked part's geometry and the trajectory coordinates followed by the end effector of the robotic arm.

The illustration above shows the positional errors occurring in the x and y axes along the trajectory of the part geometry during interpolation. These errors primarily result from the resolutions of the camera and servo motors used.



```
phi=phis*pi/180;
xb=xc-l3*cos(phi);
yb=yc-l3*sin(phi);
t2=acos(xb./((xb.^2+yb.
t3=pi-acos((l1^2+l2^2-x
t4=phi-t2-t3;
t2s=t2.*(180/pi);
t3s=t3.*(180/pi);
t4s=t4.*(180/pi);
t2ss=512+t2s.*(512/150);
t3ss=512+t3s.*(512/150);
t4ss=512+t4s.*(512/150);
```

Figure 10. Interpolation, kinematics and data transmission blocks.

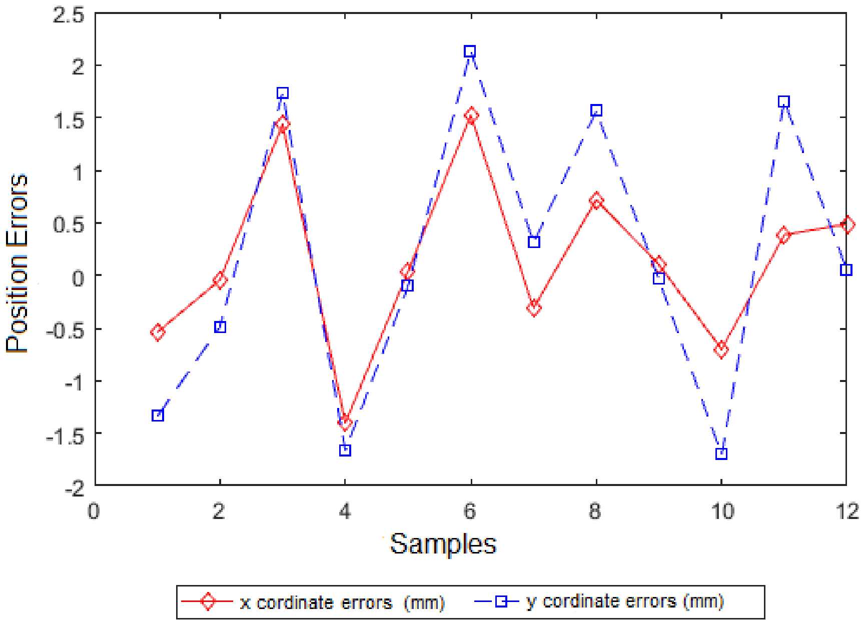


Figure 11. Graphical representation of position errors observed at the interpolation points.

Given that one side of the piece being used is known to be 7 cm, the maximum error rate encountered during advancement along an axis is less than 4% in any cases. As the robotic arm is designed as a SCARA type and exhibits serial manipulator characteristics, positional errors originating from resolution are transferred and magnified from one axis to another.

4. Conclusion

This study presents the design of a robotic arm that utilizes Dynamixel smart servo motors and connecting elements to track the trajectory of an object captured by a camera in a 2D plane. The trajectory tracking process was realized through the kinematic equations of the proposed robot arm. Numeric interpolation technique was employed in the transformation of pixel-based coordinates to millimetric coordinates for the feature points extracted through image processing techniques. Errors that occurred along the tracked trajectory were identified through coordinate feedback from the camera. Position accuracy was demonstrated through graphical representations. The established setup is capable of performing operations for various purposes by integrating different end effectors onto it for possible future studies.

Author Contributions

The percentage of the author(s) contributions is presented below. All authors reviewed and approved the final version of the manuscript.

	Y.H.E.N.	D.K.	M.S.D.	M.S.S.	N.Y.P.
C	35	10	25	20	10
D	100				
S	100				
DCP	50		50		
DAI			100		
L	20	20	20	20	20
W	20	20	20	20	20
CR	20	20	20	20	20
SR	20	20	20	20	20
PM	20	20	20	20	20
FA	20	20	20	20	20

C=Concept, D= design, S= supervision, DCP= data collection and/or processing, DAI= data analysis and/or interpretation, L= literature search, W= writing, CR= critical review, SR= submission and revision, PM= project management, FA= funding acquisition.

Conflict of Interest

The authors declared that there is no conflict of interest.

Ethical Consideration

Ethics committee approval was not required for this study because of there was no study on animals or humans.

References

Budiyanta NE, Sereati CO, Lukas L. 2020. PD controller computer vision and robotics integration based for student's programming comprehension improvement. Telecommun Comput Electron Cont, 18(2): 899-906.

Cheah CC, Liu C, Slotine JJE. 2006. Adaptive tracking control for robots with unknown kinematic and dynamic properties. Int J Robot Res, 25(3): 283-296.

Chen CY, Liao PS, Cheng CC, Jong GF. 2007. Design and implementation of integrated non-uniform rational B-spline and digital differential analyser interpolators for computerized numerical control servocontrollers. Proceedings of the Institution of Mechanical Engineers, Part C: J Mechan Eng Sci, 221(9): 1075-1087.

Ding F, Liu C. 2018. Applying coordinate fixed Denavit–Hartenberg method to solve the workspace of drilling robot arm. Int J Adv Robot Syst, 15(4): 1729881418793283.

Dixon WE. 2007. Adaptive regulation of amplitude limited robot manipulators with uncertain kinematics and dynamics. IEEE Transact Auto Cont, 52(3): 488-493.

Duque-Suárez N, Amaya-Mejía LM, Martinez C, Jaramillo-Ramirez D. 2022. Deep learning for safe human-robot collaboration. In advances in automation and robotics research. Proceedings of the 3rd Latin American Congress on Automation and Robotics, Monterrey, Mexico, May 12-15, pp: 239-251.

Dyck M, Tavakoli M. 2013. Measuring the dynamic impedance of the human arm without a force sensor. IEEE 13th International Conference on Rehabilitation Robotics (ICORR), June 24-26, Seattle, US, pp: 1-8.

El Naser YH, Atalı G, Karayel D, Özkan SS. 2020. Prototyping an industrial robot arm for deburring in machining. Acad Plat J Eng Sci, 8(2): 304-309.

Fang L, Liu G, Li Q, Zhang H. 2022. A high-precision non-uniform rational B-spline interpolator based on S-shaped feedrate scheduling. Int J Adv Manufact Technol, 121(3-4): 2585-2595.

Garriga-Casanovas A, Baena F. 2019. Kinematics of continuum robots with constant curvature bending and extension capabilities. J Mechan Robot, 11(1): 011010.

Grzelczyk D, Szymanowska O, Awrejcewicz J. 2019. Kinematic and dynamic simulation of an octopod robot controlled by different central pattern generators. Proceedings of the Institution of Mechanical Engineers, Part I: J Syst Cont Eng, 233(4): 400-417.

Guzmán-Giménez J, Valera Fernández Á, Mata Amela V, Díaz-Rodríguez MÁ. 2020. Synthesis of the Inverse Kinematic Model of non-redundant open-chain robotic systems using Groebner Basis theory. Appl Sci, 10(8): 2781.

He B, Xu F, Zhang P. 2021. Kinematics model approach to energy efficiency in sustainable manufacturing. Res Square, https://doi.org/10.21203/rs.3.rs-652070/v1.

Jian-Peng S, Jin-Gang J, Wei Q, Zhi-Yuan H, Hong-Yuan M, Shan Z. 2023. Digital interactive design and robot-assisted preparation experiment of tooth veneer preparation: An in vitro proof-of-concept. IEEE Access, 11: 30292-30307.

Jiao SX, Wang H, Xia LL, Zhang S. 2018. Research on trajectory planning of 6-DOF cutting-robot in machining complex surface. MATEC Web of Conf, 220: 06003.

Liu X, Tao R, Tavakoli M. 2014. Adaptive control of uncertain nonlinear teleoperation systems. Mechatronics, 24(1): 66-78.

Luo H, Fu J, Jiao L, Liu G, Yu C, Wu T. 2019. Kinematics and dynamics analysis of a new-type friction stir welding robot and its simulation. Adv Mechan Eng, 11(7): 1687814019866518.

Mahmoodpour M, Lobov A, Hayati S, Pastukhov A. 2019. An affordable deep learning based solution to support pick and place robotic tasks. In Instrumentation Engineering, Electronics and Telecommunications–2019: Proceedings of the V International Forum, November 20-22, Izhevsk, Russian Federation, pp: 66-75.

Nansai S, Ando Y, Itoh H, Kamamichi N. 2021. Design and implementation of a lizard-inspired robot. Appl Sci, 11(17): 7898.

Romero-González C, García-Varea I, Martínez-Gómez J. 2022. Shape binary patterns: an efficient local descriptor and keypoint detector for point clouds. Multimedia Tools Appl, 81(3): 3577-3601.

Sophokleous A, Christodoulou P, Doitsidis L, Chatzichristofis SA. 2021. Computer vision meets educational robotics. Electronics, 10(6): 730.

Staicu S, Shao Z, Zhang Z, Tang X, Wang L. 2018. Kinematic analysis of the X4 translational–rotational parallel robot. Int J Adv Robot Syst, 15(5): 1729881418803849.

Syahrian NM, Risma P, Dewi T. 2017. Vision-based pipe monitoring robot for crack detection using canny edge detection method as an image processing technique. Kinetik: Game Technol Info Syst Comput Network Comput Electron Cont, 2017: 243-250.

Tao Y, Chen F, Xiong H. 2014. Kinematics and workspace of a 4-DOF hybrid palletizing robot. Adv Mechan Eng, 6: 125973.

Uk ME, Sajjad Ali Shah FB, Soyaslan M, Eldogan O. 2020. Modeling, control, and simulation of a SCARA PRR-type robot manipulator. Scientia Iranica, 27(1): 330-340.

Wang T, Wu Y, Liang J, Han C, Chen J, Zhao Q. 2015. Analysis and experimental kinematics of a skid-steering wheeled robot based on a laser scanner sensor. Sensors, 15(5): 9681-9702.

Yanto L, Dewanto RS, Pramadihanto D, Binugroho EH. 2017. Teen-Size humanoid â€ œFLoWâ€ complete analytical kinematics. EMITTER Int J Eng Technol, 5(2): 298-311.

Zhang T, Cao Y, Ma G. 2022. Trajectory planning of 3-CRU parallel robot with linear kinematics equation. Proceedings of the Institution of Mechanical Engineers, Part C: J Mechan Eng Sci, 236(17): 9589-9609.

Zhou Q, Yuan K, Zou W, Lu P, Hu H. 2005. A multi-scale focus pseudo omni-directional robot vision system with intelligent image grabbers. IEEE/ASME International Conference on Advanced Intelligent Mechatronics, July 24-28, Monterey, US, pp: 1563-1568.