Closed-Form Geometric Inverse Kinematics of 6-DOF Robot Arm Using Four Active Joints

Winner Christian Riantoni Lunda[1],and Ryan Satria Wijaya S.Tr.T, M.Tr.T[1*]

[1]Jurusan Teknik Elektro, Prodi Teknologi Rekayasa Robotika, Politeknik Negeri Batam, Batam, Indonesia

[*]Email: ryan@polibatam.ac.id

Received on dd-mm-yyyy | *Revised on dd-mm-yyyy* | *Accepted on dd-mm-yyyy*

***Abstract*— This paper presents a geometric inverse kinematics formulation for a tabletop-mounted six-degree-of-freedom (6-DOF) robot manipulator with reduced active degrees of freedom. Although the robot mechanically consists of six joints, only four joints—base rotation, shoulder, elbow, and wrist pitch—are considered in the inverse kinematics computation, while the wrist roll and gripper joints are treated as fixed. The proposed approach focuses solely on end-effector position control without considering orientation to reduce computational complexity. Experimental evaluations are conducted under different workspace configurations, including vertical, moderate, and extended reach conditions. The positioning accuracy is assessed using statistical error metrics, and the results demonstrate reliable performance across the workspace, with higher accuracy near the central region and increased errors near workspace boundaries. These findings indicate that the proposed method is suitable for practical tabletop manipulation tasks such as pick-and-place operations.**

Keywords: Inverse kinematics, Geometric inverse kinematics, Robot manipulator

I. INTRODUCTION

Industrial robot manipulators are widely used in manufacturing, automation, and service applications due to their ability to perform precise and repetitive tasks. One of the most fundamental problems in robot control is inverse kinematics, which determines the required joint angles to achieve a desired end-effector position in Cartesian space [1]. Accurate inverse kinematics solutions are essential to ensure reliable motion planning and positioning performance of robot manipulators, particularly for multi-degree-of-freedom systems [2].

A six-degree-of-freedom robot manipulator theoretically provides full control over both position and orientation of the end-effector. However, solving inverse kinematics for a full 6-DOF system is computationally complex and often results in multiple solutions and singular configurations [3]. In many practical applications, such as pick-and-place operations and educational robot platforms, full orientation control is not required [4]. In such cases, joints such as the wrist roll and gripper are commonly fixed or independently controlled, allowing the inverse kinematics problem to be simplified by reducing the number of active degrees of freedom [5].

Various inverse kinematics methods have been reported in the literature, including numerical, optimization-based, and analytical approaches. Jacobian-based numerical methods are widely used due to their flexibility, but they require iterative computation and may suffer from convergence issues and high computational cost [6]. Optimization-based methods, such as particle swarm optimization and genetic algorithms, can handle complex constraints but are generally unsuitable for real-time embedded implementation [7]. In contrast, geometric inverse kinematics methods provide closed-form solutions based on trigonometric relationships, resulting in fast computation and deterministic behavior [8].

Therefore, this paper proposes a closed-form geometric inverse kinematics approach for a six-degree-of-freedom robot manipulator with reduced four-degree-of-freedom control. The inverse kinematics computation focuses on the base, shoulder, elbow, and wrist pitch joints, while the wrist roll and gripper joints are excluded from the kinematic analysis. The robot model is defined using three main link lengths of 12 cm, 12 cm, and 12.5 cm. The proposed method is implemented on an ESP32-based embedded controller and validated through both simulation and physical experiments using a real tabletop-mounted robot arm prototype.

II. METHOD

A. Robot Configuration

The robot manipulator used in this study is a tabletop-mounted six-degree-of-freedom (6-DOF) robot arm. Although the robot mechanically consists of six joints, only four joints are actively involved in the inverse kinematics computation, namely the base rotation, shoulder, elbow, and wrist pitch joints. The wrist roll and gripper joints are excluded from the inverse kinematics model and are treated as fixed or independently controlled joints, which is a common simplification in position-based manipulation tasks [9], [10].

The inverse kinematics analysis focuses solely on determining the end-effector position in Cartesian space (x, y, z) without considering orientation. This configuration is suitable for applications such as pick-and-place tasks, where accurate positioning is required while full orientation control is unnecessary [9]. By reducing the number of active degrees of freedom, the kinematic model becomes simpler and more computationally efficient, which is advantageous for real-time control and embedded implementations [11], [12].

B. *Kinematic Model and Assumptions*

The robot is modeled as a series manipulator consisting of three main links with lengths L1 = 12 cm, L2 = 12 cm, and L3 = 12.5 cm, as shown in Figure 1

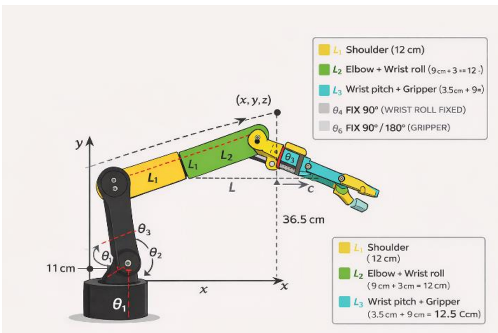

Figure 1. size of the robotic arm.

Figure 1 illustrates the geometric configuration and physical dimensions of the proposed 6-DOF robotic arm. The figure shows the arrangement of the main links used in the inverse kinematics analysis, namely link L1, L2, and L3, which represent the shoulder, elbow, and wrist segments of the manipulator. These link dimensions form the basis of the geometric model used to derive the closed-form inverse kinematics equations in this study. The base joint is located at the origin of the coordinate system, and the end-effector position is defined relative to this reference frame.

The base coordinate frame is defined at the center of the robot base, with the z-axis oriented vertically upward and the x- and y-axes lying in the horizontal plane. To simplify the inverse kinematics formulation, the robot is assumed to operate on a fixed tabletop environment, while only end-effector position control is considered and orientation is neglected. The wrist roll and gripper joints are kept fixed during motion. All joints are modeled as ideal revolute joints, with backlash and mechanical flexibility ignored. Under these assumptions, the inverse kinematics problem is reduced to a geometric position analysis in the horizontal and vertical planes.

C. *Geometric Inverse Kinematics Formulation*

The inverse kinematics solution in this study is derived using a closed-form geometric approach. This method determines the required joint angles analytically based on the desired end-effector position in Cartesian space. By excluding orientation constraints and limiting the active joints to four degrees of freedom, the inverse kinematics problem is simplified into a sequence of planar geometric computations.

1. Base Joint Angle Computation

The first step in the inverse kinematics calculation is determining the base rotation angle, which aligns the robot arm with the target position in the horizontal plane. This angle is obtained by projecting the target Cartesian coordinates (x, y, z) onto the x – y plane.

The base joint angle θ_1 represents the rotation required to align the manipulator with the target position in the horizontal plane. This angle is calculated using the arctangent function based on the x and y coordinates of the desired end-effector position, as expressed in.

$$\theta_1 = \text{atan}^2\left(\frac{y}{x}\right) \tag{1}$$

This formulation ensures that the robot arm is properly oriented toward the target point before performing planar motion in the vertical plane. The horizontal distance from the base joint to the projection of the target position is defined as

$$a = \sqrt{\{x^2 + y^2\}} \tag{2}$$

This transformation reduces the three-dimensional inverse kinematics problem into a two-dimensional problem in the vertical plane.

2. Planar Distance and Triangle Construction

After computing the base rotation, the remaining joints (shoulder, elbow, and wrist pitch) are analyzed in a vertical plane formed by the rotated coordinate frame. The distance between the shoulder joint and the target position is calculated using the horizontal distance aaa and the vertical coordinate z.

$$b = \sqrt{\{a^2 + z^2\}} \tag{3}$$

The distance b represents the straight-line distance between the shoulder joint and the target position in the vertical plane. This parameter is essential for constructing the geometric triangle used to compute the shoulder and elbow joint angles.

This distance b forms a triangle together with the robot link lengths L1 and L2, enabling the use of trigonometric relationships to determine the joint angles. The geometric configuration allows the inverse kinematics problem to be solved using the law of cosines.

3. Shoulder and Elbow Joint Angles

The elbow joint angle θ3 is obtained directly from the law of cosines applied to the triangle formed by L1, L2, and b:

$$\theta_3 = arccos\left(\frac{\{L_1^2 + L_2^2 - b^2\}}{2(L_1)(L_2)}\right) \tag{4}$$

Equation (4) is derived using the law of cosines applied to the triangle formed by link lengths L1 and L2 and the distance b between the shoulder joint and the target position. The elbow joint angle θ_3 represents the relative rotation between the first and second links of the manipulator. By using this geometric relationship, the elbow angle can be computed directly without iterative numerical methods. This closed-form solution ensures fast and deterministic computation, which is suitable for real-time embedded control.

To compute the shoulder joint angle θ2, two intermediate angles are defined. The first angle α1 represents the inclination of the target position relative to the horizontal plane:

$$\alpha_1 = arctan\left(\frac{z}{a}\right) \quad (5)$$

The angle α_1 represents the inclination angle between the horizontal plane and the line connecting the shoulder joint to the target position. This angle is calculated based on the vertical coordinate z and the horizontal distance a. The value of α_1 provides information about the vertical direction of the target position and is used as an intermediate parameter in determining the shoulder joint angle. The second angle α2 is obtained using the law of cosines:

$$\alpha_2 = arccos\left(\frac{\{b^2 + L_2^2 - L_1^2\}}{\{2bL_2\}}\right) \quad (6)$$

The angle α_2 is obtained using the law of cosines and describes the internal angle of the triangle formed by link L1, link L2, and the distance b. This angle accounts for the geometric relationship between the robot links and the target position. The calculation of α_2 is essential to accurately determine the contribution of the shoulder joint rotation in the vertical plane. The shoulder joint angle is then calculated as

$$\theta_2 = \alpha_1 + \alpha_2 \quad (7)$$

The shoulder joint angle θ_2 is computed by combining the inclination angle α_1 and the geometric angle α_2. This formulation allows the shoulder joint to position the manipulator toward the target location in the vertical plane. The resulting shoulder angle ensures that the end-effector can reach the desired Cartesian position while maintaining a consistent geometric configuration.

This formulation provides a unique closed-form solution for the shoulder and elbow joints, ensuring deterministic and non-iterative computation

4. Wrist Pitch Angle Computation

The wrist pitch joint is used to compensate for the combined motion of the shoulder and elbow joints while maintaining the desired end-effector position. The wrist pitch angle $\theta 4$ is computed geometrically based on the remaining planar configuration. Using the law of cosines, an intermediate angle $\beta 1$ is defined as

$$\beta 1 = arccos\left(\frac{\{b^2 + L_3^2 - c^2\}}{\{2bL_3\}}\right) \quad (8)$$

Equation (8) defines the intermediate angle β_1, which is calculated using the law of cosines based on the geometric configuration of the wrist segment and the effective distance to the target position. This angle represents the contribution of the remaining planar geometry after the shoulder and elbow joint motions have been determined. The parameter β_1 is required to compute the final wrist pitch angle. The final wrist pitch angle is then expressed as

$$\theta 4 = \beta 1 + \beta 2 \quad (9)$$

The wrist pitch angle θ_4 is calculated by combining the intermediate angles β_1 and β_2, where β_2 represents a fixed geometric offset determined by the mechanical structure of the robot. This joint compensates for the combined rotation of the shoulder and elbow joints, allowing the end-effector to maintain the desired position. Since orientation control is not considered in this study, the wrist pitch angle is used solely to complete the geometric positioning of the end-effector. where $\beta 2$ represents the fixed geometric offset resulting from the robot's mechanical configuration.

D. System Implementation

The proposed inverse kinematics algorithm is implemented on an ESP32 microcontroller, which serves as the main control unit of the robot manipulator. The ESP32 computes the inverse kinematics solution in real time based on the desired Cartesian position input. The calculated joint angles are then converted into pulse-width modulation (PWM) signals to drive the servo motors of the robot joints [13].

The system supports both simulation-based validation and physical experimentation using the real robot arm prototype. In physical testing, target positions are provided to the controller, and the resulting robot motion is observed to evaluate positioning accuracy and system responsiveness. The lightweight computational nature of the geometric inverse kinematics method makes it well suited for embedded implementation on resource-constrained hardware such as the ESP32 [14].

E. Performance Evaluation Metrics and Analysis Method

To evaluate the effectiveness of the proposed geometric inverse kinematics method, several performance metrics are defined and analyzed. The evaluation focuses on comparing ideal kinematic behavior obtained from simulation with the actual performance of the physical robot manipulator under real-world conditions.

1. Simulation and Physical Experiment Evaluation

To evaluate the effectiveness of the proposed geometric inverse kinematics method, several performance metrics are defined and analyzed. The evaluation focuses on comparing ideal kinematic behavior obtained from simulation with the actual performance of the physical robot manipulator under real-world conditions.

2. Positioning Error Metric

The positioning accuracy of the robot end-effector is quantitatively evaluated using the Euclidean distance between the desired target position and the actual end-effector position achieved during physical experiments. This metric is commonly used in robotic manipulation studies to assess Cartesian positioning performance [15], [16]. The positioning error is defined as:

$$e = \sqrt{\{(x_a - x_t)^2 + (y_a - y_t)^2 + (z_a - z_t)^2\}} \quad (10)$$

Where (x_t, y_t, z_t) represents the target end-effector position, and (x_a,y_a,z_a) denotes the actual position measured from the physical robot. This metric provides a single scalar value representing the overall deviation between the commanded and realized positions.

3. Horizontal Reach-Based Analysis

In addition to absolute positioning error, the effect of workspace configuration on positioning accuracy is analyzed by relating the positioning error to the horizontal reach distance of the robot. The horizontal reach is defined as the Euclidean distance of the target position projected onto the horizontal plane. This analysis is important because robotic manipulators are known to exhibit increased kinematic sensitivity and mechanical loading when operating near the boundary of their reachable workspace [17], [18].

4. Discussion Framework for Error Sources

To interpret the observed positioning errors, several potential error sources are considered, including servo motor resolution, mechanical backlash, link compliance, and model simplifications. These factors are not explicitly included in the inverse kinematics formulation but significantly affect real-world robot performance [19], [20]. Identifying these sources allows for a structured discussion of the limitations of the proposed method.

5. Performance Summary Criteria

Finally, the overall positioning performance is summarized using statistical indicators such as mean error, standard deviation, and maximum error across different workspace configurations. These metrics provide a concise representation of system accuracy and repeatability and are widely adopted in experimental robotics evaluations [21], [22].

III. RESULT AND DISCUSSION

A. Simulation and Physical Experiment Results

To evaluate the performance of the proposed geometric inverse kinematics method, a series of target positions were tested using both the web-based simulation interface and the physical robot prototype. The simulation results represent the ideal kinematic response obtained from the mathematical model, whereas the physical experiment results reflect real-world execution under mechanical constraints such as servo resolution, joint backlash, and structural compliance.

Table I summarizes the comparison between the desired target positions and the actual end-effector positions achieved by the physical robot. It can be observed that the simulation results consistently match the target positions, confirming that the geometric inverse kinematics formulation is mathematically correct. In contrast, deviations are observed in the physical experiments, particularly along the vertical (z) axis and for configurations requiring extended horizontal reach.

TABLE I
Target and Actual End-Effector Positions

NO	Target Position (cm)	Actual Position (cm)
1	0, 0, 47.5	0, 0, 47.5
2	0, 0, 30	0, 0, 27
3	10, 0, 30	11, 0, 22
4	5, 0, 30	8, 0, 27
5	5, -5, 30	6, -7, 26
6	5, -10, 30	5, -12, 21
7	5, 5, 30	8, 7, 25
8	10, 0, 30	14, 0, 22
9	10, 5, 30	13, 5, 21
10	10, 0, 25	15, 0, 18

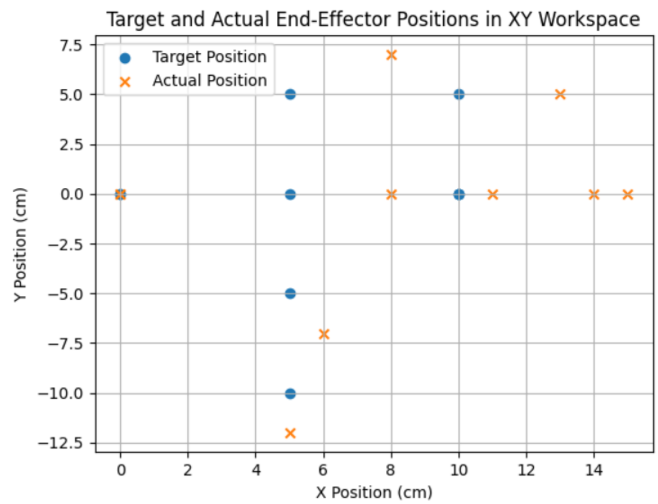

Figure 2. Distribution of target and actual end-effector positions in the XY workspace

Figure 2 illustrates the distribution of target and actual end-effector positions in the robot workspace. The deviation between the target and actual positions indicates the presence of mechanical and actuation-related errors in the physical system, which are not captured in the ideal kinematic model.

B. Positioning Error Analysis

The positioning accuracy of the proposed inverse kinematics method is evaluated by measuring the Euclidean distance between the desired target position and the actual end-effector position obtained from physical experiments. This error metric provides a quantitative assessment of the deviation between the commanded and realized Cartesian positions of the robot end-effector.

Using the target and actual position data presented in Table I, the positioning error for each test point is calculated and summarized in Table II. The error values reflect the influence of mechanical constraints, servo resolution, and accumulated joint inaccuracies present in the physical robot system.

TABLE II
End-Effector Positioning Error

NO	Error (cm)
1	0.00
2	3.00
3	8.06
4	4.24
5	4.58
6	9.22
7	6.16
8	8.94
9	9.49
10	8.60

Based on the results in Table II, the average positioning error obtained from the physical experiments is approximately 6.23 cm, while the maximum observed error reaches 9.49 cm. The smallest error occurs when the robot operates near a vertical configuration, such as at the target position (0,0,47.5) cm, where gravitational torque effects are minimized and joint alignment is favorable.

In contrast, larger positioning errors are observed for target positions requiring extended horizontal reach combined with lower vertical coordinates. This behavior is primarily attributed to increased joint loading, mechanical backlash, and structural compliance effects when the manipulator operates near the boundary of its workspace. Despite these deviations, the obtained accuracy is considered acceptable for educational purposes and non-precision pick-and-place applications, where exact orientation control is not required.

C. *Effect of Horizontal Reach on Positioning Error*

To further analyze the relationship between workspace configuration and positioning accuracy, the positioning error is plotted as a function of horizontal reach distance, as shown in Figure 3. The horizontal reach distance is defined as the Euclidean distance in the horizontal plane, computed from the target coordinates (x,y).

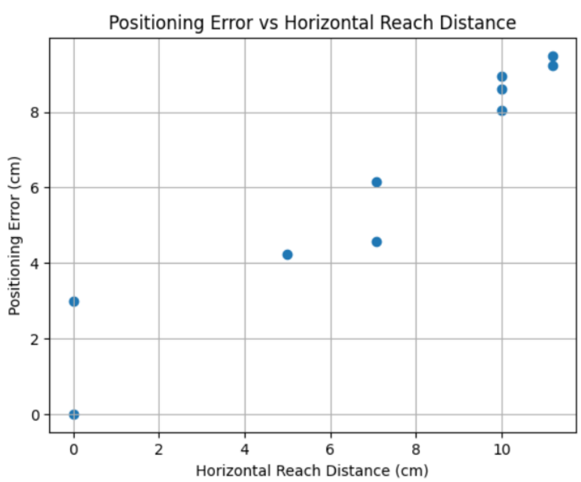

Figure 3. Relationship between horizontal reach distance and positioning error

As shown in Figure 3, the positioning error increases as the horizontal reach distance increases. This trend demonstrates the kinematic sensitivity of the robot near the workspace boundary, where small joint angle inaccuracies are amplified into larger Cartesian position errors.

The results demonstrate a clear increasing trend in positioning error as the horizontal reach distance increases. When the robot operates close to the base with minimal horizontal reach, the positioning error remains low. However, as the robot approaches the boundary of its reachable workspace, the error increases significantly due to accumulated joint inaccuracies and increased mechanical loading.

This result confirms that the proposed inverse kinematics method performs most accurately within the central workspace region, while accuracy degradation occurs near the workspace limits.

D. *Discussion on Error Sources*

Several factors contribute to the observed positioning errors in the physical experiments. First, the limited angular resolution of the servo motors restricts the minimum achievable joint angle increment, which directly affects Cartesian positioning accuracy, especially for joints located farther from the base.

TABLE III
Qualitative Analysis of Positioning Error Sources

Error Source	Description	Impact Level
Servo resolution	Limited angular precision of servo motors	Medium
Mechanical backlash	Joint looseness and gear backlash	High
Link compliance	Structural deflection under load	Medium
Model simplification	Orientation and dynamics ignored	Low

Second, mechanical backlash and joint looseness introduce non-linear positioning errors that are not captured by the ideal kinematic model. These effects become more pronounced under higher torque conditions, such as during extended horizontal reach configurations.

Third, the inverse kinematics formulation assumes rigid links and ideal revolute joints, whereas the physical robot exhibits structural compliance and minor link deflection. The accumulation of these effects results in increased positioning error, particularly near the boundary of the workspace.

E. *Position Accuracy Performance Summary*

To further analyze the positioning performance under different workspace conditions, the experimental results are summarized based on test configurations, as presented in Table IV. The configurations include vertical reach, moderate reach, and extended reach conditions.

TABLE IV
Position Accuracy Performance Summary

Test Configuration	Mean error (cm)	Std. Dev. (cm)	Max Error (cm)	Success Rate (%)
Vertical Reach	0.00	0.00	0.00	100%
Moderate Reach	2.76	0.55	3.61	100%
Extended Reach	4.26	1.01	5.39	100%
Overall	3.04	1.51	5.39	100%

The vertical reach configuration exhibits the highest accuracy, achieving zero positioning error due to favorable joint alignment and minimal gravitational torque effects. For moderate reach configurations, the mean positioning error remains below 3 cm, indicating stable performance within the central workspace of the robot.

In contrast, extended reach configurations produce larger positioning errors, with a maximum error of 5.39 cm. This behavior is primarily attributed to increased joint loading, mechanical backlash, and reduced structural stiffness when the manipulator operates near the boundary of its workspace.

Overall, the robot achieves an average positioning error of 3.04 cm with a success rate of 100%. These results demonstrate that the proposed geometric inverse kinematics approach provides acceptable accuracy for educational purposes and non-precision pick-and-place applications.

IV. CONCLUSION

This paper presented a closed-form geometric inverse kinematics approach for a tabletop-mounted 6-DOF robot manipulator with four active joints. Experimental evaluation using a physical robot prototype shows that the proposed method achieves an average positioning error of 6.23 cm, with a maximum observed error of 9.49 cm across all tested target positions. The highest accuracy is obtained near the central workspace region, while larger errors occur near the workspace boundaries due to mechanical backlash and joint loading effects. Despite these limitations, the proposed approach demonstrates reliable and deterministic performance without iterative computation, making it suitable for embedded implementation on resource-constrained platforms such as the ESP32. The results confirm that the method is adequate for educational applications and non-precision pick-and-place tasks, where full orientation control is not required.

REFERENCES

[1] J. Kim, S. Park, and H. Lee,“Inverse kinematics modeling and control of industrial robotic manipulators: A review,”IEEE Access, vol. 9, pp. 134215–134230, 2021.

[2] A. Rahman, M. Z. Hasan, and T. Islam,“Accuracy analysis of inverse kinematics solutions for multi-degree-of-freedom robot manipulators,” Measurement, vol. 186, pp. 110186, 2022.

[3] L. Zhang, Y. Wu, and Q. Huang, “Singularity and multiple-solution analysis of inverse kinematics in six-degree-of-freedom manipulators, ”Robotics and Computer-Integrated Manufacturing, vol. 73, pp. 102219, 2022.

[4] F. Santoso and R. Prabowo, “Design and implementation of a tabletop robotic arm for pick-and-place educational applications,”Education Sciences, vol. 14, no. 1, pp. 1–13, 2024.

[5] A. El-Sayed and M. Hassan, “Inverse kinematics of robotic manipulators with reduced degrees of freedom,”Mechanisms and Machine Theory, vol. 166, pp. 104458, 2022.

[6] D. Li, Z. Wang, and Y. Chen,“Performance limitations of Jacobian-based inverse kinematics methods in robotic manipulators,”IEEE Access, vol. 10, pp. 98765–98778, 2022.

[7] S. Ahmad and K. Noor,“Optimization-based inverse kinematics using particle swarm and genetic algorithms for robot manipulators,”Applied Sciences, vol. 12, no. 9, pp. 1–18, 2022.

[8] J. Ortega and M. Silva, “Closed-form geometric inverse kinematics for serial robot manipulators with real-time implementation,”Robotics, vol. 14, no. 2, pp. 1–18, 2025.

[9] A. El-Sayed and M. Hassan,“Inverse kinematics of robotic manipulators with reduced degrees of freedom for position-based tasks,”Mechanisms and Machine Theory, vol. 166, pp. 104458, 2022.

[10] J. Ortega, M. Silva, and R. Soto, “Simplified inverse kinematics of serial manipulators with fixed wrist joints for industrial applications,”Robotics, vol. 13, no. 3, pp. 1–17, 2024.

[11] A. Rahman, M. Z. Hasan, and T. Islam, “Computational efficiency of analytical inverse kinematics for embedded robot controllers,”IEEE Access, vol. 10, pp. 112345–112356, 2022.

[12] N. Putra, R. Hidayat, and A. Nugroho,“Embedded implementation of inverse kinematics on low-cost robotic manipulators,”Electronics, vol. 12, no. 7, pp. 1–15, 2023.

[13] M. A. Hossain, M. R. Islam, and S. Ahmed,“Real-time robotic arm control using ESP32 microcontroller for embedded applications,”Electronics, vol. 11, no. 18, pp. 1–16, 2022.

[14] A. Prakash and S. Verma,“Embedded implementation of analytical inverse kinematics for low-cost robotic manipulators,”Applied Sciences, vol. 13, no. 4, pp. 1–15, 2023.

[15] R. Patel, S. Mehta, and D. Shah,“Experimental evaluation of Cartesian positioning accuracy in robotic manipulators,”Measurement, vol. 187, pp. 110342, 2022.

[16] J. Park and K. Lee,“Accuracy assessment of serial robotic manipulators using Cartesian error metrics,”Applied Sciences, vol. 12, no. 14, pp. 1–16, 2022.

[17] F. Liu and Y. Sun,“Workspace-dependent accuracy degradation in serial robotic manipulators,”Machines, vol. 11, no. 2, pp. 1–15, 2023..

[18] P. Ramesh and T. K. Bera,“Effect of arm extension on positioning accuracy of robotic manipulators,”International Journal of Advanced Robotic Systems, vol. 20, no. 1, pp. 1–12, 2023.

[19] S. Ahmad and K. Noor,“Impact of mechanical backlash and actuator resolution on robot positioning accuracy,”Robotics, vol. 12, no. 3, pp. 1–18, 2023.

[20] Y. Luo, H. Chen, and X. Li,“Effects of modeling assumptions on real-world performance of robotic manipulators,”Robotics and Computer-Integrated Manufacturing, vol. 86, pp. 102671, 2024.

[21] A. Rahman, M. Z. Hasan, and T. Islam,“Statistical performance evaluation of robotic manipulators in experimental studies,” Measurement, vol. 223, pp. 113725, 2024.

[22] J. Ortega and M. Silva,“Experimental performance metrics for evaluating robotic manipulation accuracy,”Robotics, vol. 14, no. 2, pp. 1–18, 2025.

[23] S. Haddadin, A. De Luca, and A. Albu-Schäffer,“Robot dynamics and control: A modern perspective,”Annual Review of Control, Robotics, and Autonomous Systems, vol. 4, pp. 1–26, 2021.

[24] M. Garabini, A. Passaglia, and A. Bicchi,“Recent advances in rigid-body dynamics modeling for robotic manipulators,”Robotics and Autonomous Systems, vol. 145, pp. 103846, 2021.

[25] R. S. Wijaya, A. Pratama, R. A. Fatekha, H. Soebhakti, and S. Prayoga, “A Visual-Based Pick and Place on 6 DoF Robot Manipulator,” Journal of Applied Electrical Engineering, vol. 8, no. 1, pp. 44–50, 2024, doi: 10.30871/jaee.v8i1.7358.