Nexo Revista Científica
Vol. 36, No. 04, pp. 482-494/Septiembre 2023

ISSN-E 1995-9516
Universidad Nacional de Ingeniería

http://revistas.uni.edu.ni/index.php/Nexo
https://doi.org/10.5377/nexo.v36i04.16752

A low-cost monocular vision system for robot calibration

Sistema de vision monocular de bajo costo para la calibración de robots

Jesús Abraham Rojas Úrzulo[1], José-Joel González-Barbosa[1,*], Xochitl Yamile Sandoval Castro[1], Maximiano Francisco Ruiz Torres[1], Erick-Alejandro González-Barbosa[2]

[1] Instituto Politécnico Nacional CICATA unidad Querétaro, México.
[2] Tecnológico Nacional de México/ITS de Irapuato, Irapuato, México.
[*]jgonzalezba@ipn.mx

(recibido/received: 25-abril-2023; aceptado/accepted: 14-agosto-2023)

ABSTRACT

The inverse kinematic model uses robot design parameters: ideal link lengths and mounting angles. In practice, these values hardly coincide with the design values due to manufacturing and assembly processes or continuous use of the robot. In order to reduce this geometric error, it is necessary to calibrate the robot to update the geometric model and reduce the resulting error of the robot end-effector. In this work, we propose a methodology based on a vision system to calibrate the robot's geometric parameters and minimize the error between the robot's end-effector theoretical and real trajectory. This way, variations are introduced to the geometric parameters that generate errors between the robot's desired position and the position developed. The results show up a reduction of the average position error of 54.6%.

Keywords: Robot calibration, trajectory tracking, D-H model compensation

RESUMEN

El modelo cinemático inverso utiliza parámetros de diseño del robot: longitudes de eslabones y ángulos de montaje ideales. En la práctica, estos valores difícilmente coinciden con los valores de diseño debido a los procesos de fabricación y montaje o al uso continuo del robot. De esta forma, se introducen variaciones en los parámetros geométricos que generan un error entre la posición deseada del robot y la posición desarrollada. Para reducir este error geométrico, es necesario realizar una calibración del robot para actualizar el modelo geométrico y reducir el error resultante del efector final del robot. En este trabajo se propone una metodología basada en un sistema de visión para calibrar los parámetros geométricos del robot y minimizar el error entre la trayectoria teórica y la real del efector final del robot. Los resultados muestran una reducción del error medio de posición del 54,6%.

Palabras claves: Calibración del robot, seguimiento de trayectoria, modelo de compensación D-H.

1. INTRODUCCIÓN

Automation has increased the number of robots to perform many industrial tasks worldwide. The International Federation of Robotics (IFR) published records mentioning that in 2020, 3.01 million robots were in operation worldwide. This number will be increased by more than 517 thousand units per year (IFR, 2022). However, in robots, positioning errors may occur due to the deviation of the nominal or design parameters compared to the real parameters. This can happen due to manufacturing and assembly errors (Costa et al., 2020; Li et al., 2019; Roth et al., 1987), friction (Roth et al., 1987), material used for its construction (Zha et al., 2020; Roth et al., 1987), temperature (Santolaria et al., 2009; Roth et al., 1987), shear forces (Olsson et al., 2010). These differences mainly affect the geometric parameters of the direct and inverse kinematic models, introducing errors to link lengths (Δl_i) and/or angular values of the joints ($\Delta \theta_i$). Within the literature, various ways exist to determine these variations to compensate and reduce the error between the desired position and the real position of the robot end-effector.

In Zu et al., (2021), the authors establish the kinematic model of a serial 6-DOF robot. This work compensates the geometric parameters and links deformation due to the load in the robot end-effector. Position average error was reduced from 7.25 mm to 1.29 mm, representing an increase in position accuracy of 82.84%. The work presented in Diaz-Cano et al., (2021) uses a hand-eye methodology for robot calibration, using a structured light 3D camera (Kinect 360) that obtains information from the real world and a six-axis industrial robotic arm. In Stepanova et al., (2022), there is a system of two serial manipulators with a sphere at the robot end-effectors of each one. Torque sensors in their motors determine the instant in which both spheres have contact. Ideally, this should occur at specific defined angular values. However, these are produced experimentally at other values, the existing difference being the data used to determine the parameters within the D-H model. In Wang et al., (2023), the authors present a classic quaternion as a good representation of rotation that exerts its advantage for the separable methods of the hand-eye calibration problems.

Following these methodologies, this work uses a vision system to calculate the compensation of the geometric parameters as in Liu et al., (2020). The authors use an optical tracking system in a serial robot of the brand Universal Robot. The optical system can obtain the coordinates of the robot end-effector and its orientation in real time. By calibrating the geometric parameters, the error of a sequence of points was reduced from 3 mm to less than 0.2 mm in average. In Wang et al., (2021), the accuracy of a robot KUKA KR500-3 is increased by 80.84% concerning the position and 57.29% concerning the orientation of the robot end-effector employing a binocular vision system. Finally, Yan et al., (2021) uses a pattern sphere and a monocular vision system mounted on the robot end-effector. The camera takes images of the sphere from different positions and orientations. Using the D-H model, the geometric center of the pattern is determined, and the difference between the known value and the calculated one is used for robot calibration. The work presented in Costa et al., (2020) proposes a strategy to evaluate the geometric errors of the rotor system of a hydrostatic rotary table. The authors use an ultra-precision CMM (Coordinate Measuring Machine). The result shows that the perpendicularity error between the two thrust plates and the rotor is 14 µm and 21 µm, respectively. The parallelism error between the two thrust plates is 28 µm with a measurement uncertainty of 0.5 µm. Furthermore, the experimental result on the optimization method to guide the rotor system assembly indicates that the parallelism error was reduced to 12.1 µm, noting an improvement of 57%.

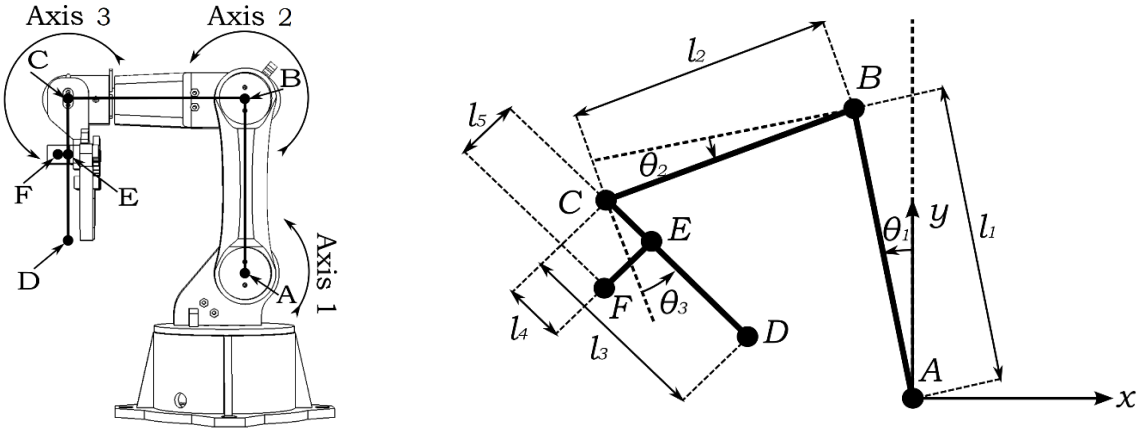
There are different procedures for robot calibration. Some works use laser tracking systems, single or dual camera vision systems, torque sensors in the manipulator's joints, and advanced optical systems capable of determining the position and orientation of the robot end-effector in real-time. However, all these tools tend to have a high cost, mainly used for calibrating industrial robots and seeing their application limited in the design and development of non-industrial prototypes. In Mechatronics, there are many implementations of two and three-DOF planar robots with open and four and five-bar linkage with closed kinematic chains. The main characteristic of all of them is that they restrict their movement to one plane. This document proposes

a tracking system capable of evaluating the performance of a robot during trajectory and their calibration using a commercial webcam. An algorithm detects the targets placed on the robot. We computed the robot end-effector coordinates *(x,y)* using the direct kinematic and pinhole camera models. The algorithm computes the coordinates of a trajectory made by the robot. The algorithm uses this information to obtain the calibration parameters that allow updating the direct and inverse kinematic model. Finally, we reduced the error considerably after the procedure.

Section 2 describes the robotic system used in the document. The authors establish the direct and inverse kinematic model of the 3-DOF planar serial manipulator, where the lengths of the links were obtained by measurements made to the CAD design. In section 3, the characteristics of the vision system are mentioned. The transformations are necessary to convert the coordinates *(u,v)* from a point on the image to a point *(x,y)* regarding the robot system, as well as the camera projection and distortion model used in the calibration of the toolbox by Matlab. The target detection algorithm uses a sequence of filters to analyze the image and determine the geometric center of the circles located on the axes of rotation of the servo motors. The target detection algorithm allows us to extract the center of the targets placed on the robot joints. In section 4, the modified direct and inverse kinematic model is described. This section describes a compensation model for the length links and the joint angles. We define the path error function and an iterative algorithm for determining the parameters that minimize the error function. Finally, section 5 presents the results obtained. Starting with the camera calibration parameters, the results of the optimization algorithm for the three calibrations performed, as well as the change in the trajectories developed by the manipulator after the update of the inverse kinematic model by analyzing how the average error in position behaves in the points of the trajectory.

2. MODEL OF THE ROBOTIC SYSTEM

Figure 1.(a) shows a serial manipulator with 5-DOF. However, in the implementation made in this article, only the three active joints with parallel axes of rotation are enabled. This simplification makes it possible to measure the position *(x,y)* of the points of interest *A*, *B*, *C*, and *F* using a single camera. This way, the robot can be modelled as a 3-DOF planar manipulator.



(a) 5-DOF serial manipulator with only 3-DOF enabled.

(b) Kinematic model of a 3-DOF planar manipulator.

Figure 1: Kinematic model and CAD model of the 3-DOF serial planar robot.

2.1 Direct and inverse kinematic models of the 3-DOF planar manipulator

During the operation of a robotic system, its end-effector must move to the required position. In order to carry out a task, it is necessary to establish a system of equations that relate the degrees of freedom of the

manipulator (position and orientation of the robot end-effector) as a function of the joint control variables. A recurring solution is to use the algorithm of Denavit-Hartenberg (D-H). This algorithm shifts the reference frame from the robot's base to the robot end-effector using rigid rotation and translation transformations on the axes *x* and *z*. To bring the reference system from point $i-1$ to point i, we calculate ${}^{i-1}A_i = T(z, \theta_i)T(z, d_i)T(x, a_i)T(x, \alpha_i)$, defined by (1).

$$ {}^{i-1}A_i = \begin{pmatrix} cos\theta_i & -cos\,\alpha_i\, sin\theta_i & sin\,\alpha_i\, sin\theta_i & a_i\, cos\,\theta_i \\ sin\theta_i & cos\,\alpha_i\, cos\theta_i & -sin\,\alpha_i\, cos\theta_i & a_i\, sin\,\theta_i \\ 0 & sin\,\alpha_i & cos\,\alpha_1 & d_i \\ 0 & 0 & 0 & 1 \end{pmatrix} \quad (1) $$

Where $T(z, \theta_i)$ is a rotation of θ_i degrees around the axis *z*, $T(z, d_i)$ is a translation of d_i units through the axis *z*, $T(x, a_i)$ is a translation of a_i units through the axis *x*, and $T(x, \alpha_i)$ is a rotation of α_i degrees around the axis *x*. The kinematic model of a 3-DOF planar robot using the D-H algorithm has been described numerous times in existing literature (Banga et al., 2009; Duka et al., 2014; ZHEKOV et al., 2020; Gonzalez-Barbosa et al., 2022; Elsisi et al., 2021). Table 1 shows nominal robot geometrical parameters, where an offset has been added in angles of the active joints. In this way, the zero position coincides with the one shown in Figure 1.(a). In addition, point *C*, *E*, and *F* are rigidly joined, that is, the angle formed between the vectors **CE** and **EF** is constant, as well as their magnitude. The length of the links shown in Figure 1.(b) was obtained by making measurements in the CAD design.

Table 1. D-H parameters of the 3-DOF planar manipulator.

Point *i*	$a_i\,(mm)$	$\alpha_i\,(rad)$	$d_i\,(mm)$	$\theta_i\,(rad)$
B	$l_1 = 120.00$	*0*	0	$\theta_1 + \frac{\pi}{2}$
C	$l_2 = 121.70$	*0*	0	$\theta_2 + \frac{\pi}{2}$
E	$l_4 = 38.00$	*0*	0	$\theta_3 + \frac{\pi}{2}$
F	$l_5 = 6.73$	*0*	0	$-\frac{\pi}{2}$

Using Equation 1 and data presented in Table 1. Equations (2), (3), and (4) were obtained. These equations represent coordinates *(x,y)* of the point *F*, and the angle of rotation around *z*-axis of the robot end-effector.

$$ F_x = -l_1 s(\theta_1) - l_2 c(\theta_{12}) + l_4 s(\theta_{123}) - l_5 c(\theta_{123}) \quad (2) $$

$$ F_y = l_1 c(\theta_1) - l_2 s(\theta_{12}) - l_4 c(\theta_{123}) - l_5 s(\theta_{123}) \quad (3) $$

$$ \theta = \theta_{123} + \pi \quad (4) $$

Where $\theta_{12}=\theta_1+\theta_2$, $\theta_{123}=\theta_1+\theta_2+\theta_3$, $s(\theta)=sin(\theta)$, y $c(\theta)=cos(\theta)$. Using equations (2), (3), and (4), inverse kinematics of the robot is determined, $\theta_i(F_x,F_y,\theta_F),\ i\in\{1,2,3\}$. Given the coordinates *(x,y)* and orientation θ_F that are required in the robot end-effector we can calculate articular displacements necessary to carry out it. Implementing (1) again and data from Table 1 equations (5) and (6) are established. These equations represent the coordinates *(x,y)* of point *C*.

$$ C_x = F_x - l_4 s(\theta_F - \pi) + l_5 c(\theta_F - \pi) \quad (5) $$

$$ C_y = F_y + l_4 c(\theta_F - \pi) + l_5 s(\theta_F - \pi) \quad (6) $$

With auxiliary equations (5) and (6) equations (7), (8), and (9) are established.

$$\theta_2 = -arcsin\left(\frac{C_x^2+C_y^2-l_1^2-l_2^2}{2l_1l_2}\right) \tag{7}$$

$$\theta_1 = -arctan\left(\frac{C_y l_2 c(\theta_2)-C_x(l_2 s(\theta_2)-l_1)}{C_x l_2 c(\theta_2)+C_y(l_2 s(\theta_2)-l_1)}\right) \tag{8}$$

$$\theta_3 = \theta_F - \theta_1 - \theta_2 - \pi \tag{9}$$

3. MODEL OF THE VISION SYSTEM

In Durović et al. (2017), the authors use a RGB-D vision system external to the robot end-effector that allows observing at the same time position and orientation of the SCARA robot end-effector and the object to be taken. The authors propose a two-step calibration system and a low-cost vision system-based planning method. The positioning of the tool is achieved using mark tracking and depth information that is provided by RGB-D camera without encoders or other sensors.

This work proposes to measure coordinates *(x,y)* of the point *F*, as shown in Figure 2. Using the toolbox by Matlab camera calibration, we compute the intrinsic and extrinsic parameters of rotation and translation of the camera that allow us to project a point in the camera reference system to the reference system of the calibration pattern defined by (10).

$$\boldsymbol{P}_{cal} = {}^{cal}\boldsymbol{R}_{cam}\boldsymbol{P}_{cam} + {}^{cal}\boldsymbol{t}_{cam} \tag{10}$$

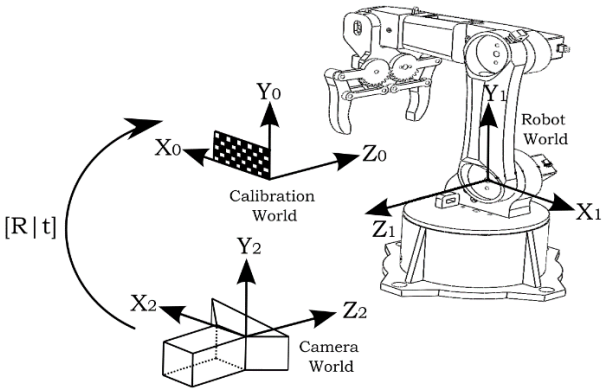


Figure 2. Installation of the robot, camera, and calibration standard.

Where $\boldsymbol{P}_{cal}$ is a point concerning to calibration pattern's coordinate system, $\boldsymbol{P}_{cam}$ is a point concerning to camera's coordinate system, and ${}^{cal}\boldsymbol{R}_{cam}$ and ${}^{cal}\boldsymbol{t}_{cam}$ are the rotation matrix and the translation vector respectively, between both reference systems.

However, we need to obtain coordinates *(x,y)* in mm of the point *F* regarding point *A*. The vector $\boldsymbol{AF}_{rob} = \boldsymbol{F}_{rob} - \boldsymbol{A}_{rob}$ is calculated, where the subscript rob indicates that they are referenced to robot's coordinate system. Nevertheless, both points measured by the camera are referenced to the calibration pattern obtained by equation (11).

$$\boldsymbol{AF}_{rob} = \boldsymbol{F}_{rob} - \boldsymbol{A}_{rob} = {}^{rob}\boldsymbol{R}_{cal}\,(\boldsymbol{F}_{cal} - \boldsymbol{A}_{cal}) \tag{11}$$

In equation (11), it can be observed that the vector of translation between the calibration pattern's coordinate system and robot's coordinate system ${}^{rob}\boldsymbol{t}_{cal}$ disappears due to the subtraction. ${}^{rob}\boldsymbol{R}_{cal}$ represents the rotation matrix between both reference systems. The matrix transformation does not have rotation around *x* and *z* axes, it only has a rotation of π radians around *y* axis. This rotation on *xz*-plane is represented by the equation (12).

$$\begin{pmatrix} -1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & -1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} x \\ y \\ z \\ 1 \end{pmatrix} = \begin{pmatrix} -x \\ y \\ -z \\ 1 \end{pmatrix} \tag{12}$$

3.1 Camera model

The pinhole model has been implemented repeatedly within literature (Yan et al., 2021; Arredondo-Soto et al., 2021). This model is used in Matlab toolbox camera calibration defined by (13).

$$w\begin{pmatrix} u & v & 1 \end{pmatrix} = \begin{pmatrix} X & Y & Z & 1 \end{pmatrix} \begin{pmatrix} \boldsymbol{R} \\ \boldsymbol{K} \end{pmatrix} \boldsymbol{K} = \begin{pmatrix} X & Y & Z & 1 \end{pmatrix} \begin{pmatrix} r_{11} & r_{12} & r_{13} \\ r_{21} & r_{22} & r_{23} \\ r_{31} & r_{32} & r_{33} \\ t_x & t_y & t_z \end{pmatrix} \begin{pmatrix} f_x & 0 & 0 \\ s & f_y & 0 \\ c_x & c_y & 1 \end{pmatrix} \tag{13}$$

Where extrinsic parameters are represented by $\boldsymbol{R}$, and $\boldsymbol{T}$, the rotation matrix, and the translation vector, respectively. Matrix $\boldsymbol{K}$ represents intrinsic parameters. Vector $(u \quad v \quad 1)$ corresponds to homogeneous coordinates of a point in the digital image, $(X \quad Y \quad Z \quad 1)$are the homogeneous coordinates of a 3D point in world reference system, and *w* is a scale factor.

To simplify computing of the measurement algorithm, product of extrinsic and intrinsic parameters matrices can be performed to obtain the matrix $\boldsymbol{M}$ defined by (14).

$$\boldsymbol{M} = \begin{pmatrix} m_{11} & m_{12} & m_{13} \\ m_{21} & m_{22} & m_{23} \\ m_{31} & m_{32} & m_{33} \\ m_{41} & m_{42} & m_{43} \end{pmatrix} = \begin{pmatrix} r_{11} & r_{12} & r_{13} \\ r_{21} & r_{22} & r_{23} \\ r_{31} & r_{32} & r_{33} \\ t_x & t_y & t_z \end{pmatrix} \begin{pmatrix} f_x & 0 & 0 \\ s & f_y & 0 \\ c_x & c_y & 1 \end{pmatrix} = \begin{pmatrix} \boldsymbol{R} \\ \boldsymbol{K} \end{pmatrix} \boldsymbol{K} \tag{14}$$

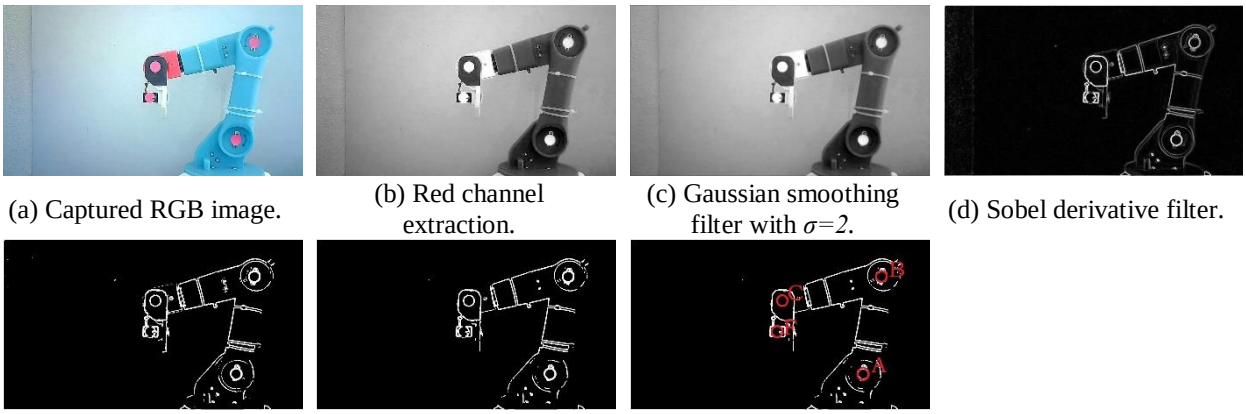
Finally. Calibration results also determines camera's radial and tangential distortion parameters necessary to eliminate distortion due to the manufacture of camera's lens. These values are obtained in a vector $(k_1 \quad k_2 \quad k_3 \quad p_1 \quad p_2)$, where k_1, k_2, and k_3 are associated with radial distortion and p_1 and p_2 with tangential distortion. Radial and tangential distortion are calculated using (15).

$$\begin{pmatrix} x_q \\ y_q \end{pmatrix} = (1 + k_1 r^2 + k_2 r^4 + k_3 r^6) \begin{pmatrix} u \\ v \end{pmatrix} + \begin{pmatrix} 2p_1 uv + p_2(r^2 + 2u^2) \\ p_1(r^2 + 2u^2) + 2p_2 uv \end{pmatrix} \tag{15}$$

Where (x_q, y_q) are coordinates of a point in the distorted image, (u,v) are coordinates of a point in the corrected image, and $r^2 = u^2 + v^2$.

3.2 Measurement algorithm

To calculate the coordinates (x,y) of the point *F*. We have to determine the coordinates (u,v) of the point within the image taken by the camera and use (13) that relates both reference systems. The algorithm shown in Figure 3 was implemented, it shows the steps carried out to identify the points *A*, *B*, *C*, and *F* in the digital image. Sampling of fifty images of the manipulator was carried out in each developed trajectory.



(a) Captured RGB image. (b) Red channel extraction. (c) Gaussian smoothing filter with $\sigma=2$. (d) Sobel derivative filter.

(e) Binarization with $\mu=0.5$. (f) Morphological opening operation. (g) Detection by circular Hough transform.

Figure 3. Steps to follow up on key points.

A sequence of filters was implemented to extract the outline of the circles. Once the capture is made, the red channel is taken, which presents the greatest contrast between the patterns to be followed and the piece where they are located. A Gaussian smoothing filter is applied to remove the noise from the capture, followed by a Sobel derivative filter in u and v axes independently and then calculate the gradient using both results. Once the previous steps have been carried out, the contour of the circles is extracted. Therefore, the binarization of the image is carried out, followed by the morphological opening operation to eliminate a few isolated points. Finally, with the image shown in Figure 3.(f), the circular Hough transform is executed to determine the position of the center of the circles, which are the coordinates in the image of the points *A*, *B*, *C*, and *F*.

With the coordinates (u,v) of the points *A* and *F*, values of matrix $\boldsymbol{M}$ are calculated by (14) where the distance $z_0=-80.87$ mm, which represents the existing distance in *z*-axis between the calibration plane and the plane in which the planar manipulator moves. We have all the data that we need to use (16) to determine the coordinates (x,y) of points *A* and *F*.

$$\begin{pmatrix} x \\ y \end{pmatrix} = \begin{pmatrix} m_{11}-m_{13}u & m_{21}-m_{23}u \\ m_{12}-m_{13}v & m_{22}-m_{23}v \end{pmatrix}^{-1} \begin{pmatrix} (m_{43}u-m_{41})+(m_{33}u-m_{31})z_0 \\ (m_{43}v-m_{42})+(m_{33}v-m_{32})z_0 \end{pmatrix} \tag{16}$$

4. ROBOT CALIBRATION ALGORITHM

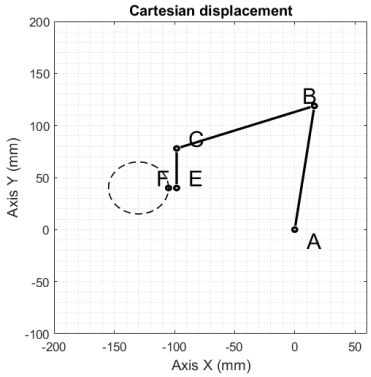
Robot calibration is performed three times using the following trajectories:

- In the first trajectory, the point *F* must be displaced in a circle centered at the point $x=-130$ mm, $y=40$ mm with a radius of 15 mm, with an orientation of the end effector of $\theta_F=\pi$.

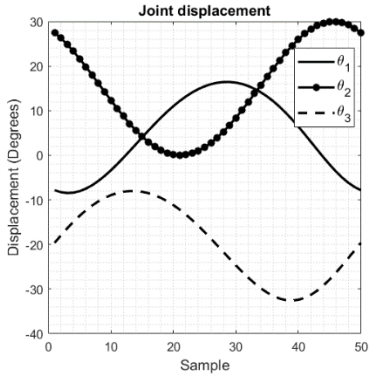
- In the second trajectory, a circle is made centered at the same point with the same orientation of the robot end-effector but this time with a radius of 25 mm.

Path 1 is used for the first calibration, path 2 is used for the second calibration, and paths 1 and 2 are used simultaneously for the third calibration. A third trajectory is used as validation. In this, the trajectory makes a straight line that goes from the point $x_1=-145$ mm, $y_1=25$ mm to the point $x_2=-115$ mm, $y_2=45$ mm. Applying (7), (8), and (9). We computed the joint trajectories corresponding to each cartesian trajectory. Figure 4 shows the joint trajectories necessary for the first cartesian trajectory. Since the sampling was of

fifty points for each trajectory, we have fifty pairs of coordinates *(x,y)*, in addition to joint values ($\theta_1,\theta_2,\theta_3$). Arduino Uno board writes these values, and once the displacement is completed, the image is captured. This image shows that the coordinates *(u,v)* of points *A* and *F* are calculated using the process described in Figure 3. Radial and tangential distortion of the image is eliminated using the parameters (k_1,k_2,k_3,p_1,p_2) obtained in the camera calibration, and then we can use the values of the matrix $\boldsymbol{M}$ in (16).



(a) Cartesian trajectory of the point *F*, circumference centered in *x=-130* mm, *y=40* mm, with a radius of 15 mm.



(b) Joints trajectory trajectories to perform the cartesian trajectory.

Figure 4. Cartesian and joints trajectories of 3-DOF planar serial robot.

Ideally, dimensions l_1, l_2, l_4, and l_5 coincide with the measurements obtained in the CAD model. However, due to 3D printing and assembly, there is a difference between its real and theoretical value Δl_i. Similarly, $\theta_1,\theta_2,\theta_3$ represent the angular values of the DC servo motors mounted in rotational joints. The zero pose should be shown in Figure 1, but at the time of installation, an error occurs generating values $\Delta\theta_i$. Introducing these variations to the ideal model presented in (2) and (3). We obtain equations (17) and (18).

$$F'_x = -l'_1 s(\theta'_1) - l'_2 c(\theta'_{12}) + l'_4 s(\theta'_{123}) - l'_5 c(\theta'_{123}) \tag{17}$$

$$F'_y = l'_1 c(\theta'_1) - l'_2 s(\theta'_{12}) - l'_4 c(\theta'_{123}) - l'_5 s(\theta'_{123}) \tag{18}$$

Where $l'_i = l_i + \Delta l_i$, $i \in \{1,2,3\}$, $\theta'_1 = \theta_1 + \Delta\theta_1$, $\theta'_{12} = \theta_{12} + \Delta\theta_1 + \Delta\theta_2$, $\theta'_{123} = \theta_{123} + \Delta\theta_1 + \Delta\theta_2 + \Delta\theta_3$. Finally, the error is defined by (19), where the values of l_1, l_2, l_4, and l_5 are those presented in Table 1, and values for θ_1, θ_2, and θ_3 are those calculated using the inverse kinematics equations (7), (8) and (9).

$$e_n = \sqrt{(x_n - F'_x)^2 + (y_n - F'_y)^2}, \quad n \in \{1,2,\ldots,N\} \tag{19}$$

Where (x_n,y_n) are the measured coordinates of the point *F* in the sample number *n* when the robot's end-effector is executing the trajectory. Since we have $\boldsymbol{N}$ samples, the vector E defined by (20) is obtained.

$$\boldsymbol{E} = (e_1, e_2, e_3, \ldots, e_N)^T \tag{20}$$

Equation (21) represents the average error of the *N* samples.

$$error = \frac{1}{N}\sum_{n=1}^{N} e_n \tag{21}$$

In our case *N=50*, in first and second calibration, while *N=100* at the third calibration. The Jaccobian of error function (function to be optimized) is defined by equation (22).

$$\boldsymbol{J} = \left(\frac{\partial \boldsymbol{E}}{\partial \Delta l_1} \quad \frac{\partial \boldsymbol{E}}{\partial \Delta l_2} \quad \frac{\partial \boldsymbol{E}}{\partial \Delta l_4} \quad \frac{\partial \boldsymbol{E}}{\partial \Delta l_5} \quad \frac{\partial \boldsymbol{E}}{\partial \Delta \theta_1} \quad \frac{\partial \boldsymbol{E}}{\partial \Delta \theta_2} \quad \frac{\partial \boldsymbol{E}}{\partial \Delta \theta_3} \right) \tag{22}$$

Using an iterative algorithm, values $\boldsymbol{X} = (\Delta l_1, \Delta l_2, \Delta l_4, \Delta l_5, \Delta \theta_1, \Delta \theta_2, \Delta \theta_3)$ that minimizes function (21) are calculated by the update equation (23) and the starting vector $\boldsymbol{X}_0$.

$$\boldsymbol{X}_{k+1} = \boldsymbol{X}_k - \alpha \boldsymbol{E}_k^T \boldsymbol{J}_k \tag{23}$$

Where α is the size of the step for the minimum error search. Algorithm 1 is proposed for the calibration of the robot**.**

Algorithm 1. Robot calibration

Procedure: Calibration
Inverse kinematics$(l_1,l_2,l_4,l_5,F_x,F_y,\theta_F) \rightarrow (\theta_1,\theta_2,\theta_3)$, *see section* 2.1
Camera calibration $\rightarrow \boldsymbol{M}, k_1, k_2, k_3, p_1, p_2$, see section 3.1
Trajectory measurement $\rightarrow \{(x_1,y_1),(x_2,y_2),(x_3,y_3),\ldots,(x_n,y_n)\}$, see section 3.2
Error minimization:
$\boldsymbol{X}_0 \leftarrow (5,5,1,1,5,5,5)$ % Initialization
$k \leftarrow 0$
While error ≥ errorMax **do:**
    $F'_x \leftarrow (l_1,l_2,l_4,l_5,\theta_1,\theta_2,\theta_3,\boldsymbol{X}_k)$, *equation (17)*
    $F'_y \leftarrow (l_1,l_2,l_4,l_5,\theta_1,\theta_2,\theta_3,\boldsymbol{X}_k)$, *equation (18)*
    $\boldsymbol{E}_k^T \leftarrow (x_1,y_1,F'_x,F'_y)$, *equation (20)*
    $error \leftarrow \boldsymbol{E}_k^T$, *equation (21)*
    $\boldsymbol{J}_k \leftarrow (l_1,l_2,l_4,l_5,\theta_1,\theta_2,\theta_3,\boldsymbol{X}_k,\boldsymbol{E}_k^T)$, *equation (22)*
    $\boldsymbol{X}_{k+1} \leftarrow \boldsymbol{X}_k,\alpha,\boldsymbol{E}_k^T,\boldsymbol{J}_k$, *equation (23)*
    $k \leftarrow k+1$
Robot calibration $\rightarrow (\Delta l_1, \Delta l_2, \Delta l_4, \Delta l_5, \Delta \theta_1, \Delta \theta_2, \Delta \theta_3)$
Inverse kinematics with D-H compensation $(l_1,l_2,l_4,l_5,F_x,F_y,\theta_F) \rightarrow (\theta_1,\theta_2,\theta_3)$
Trajectory measurement with D-H compensation $\rightarrow \{(x_1,y_1),(x_2,y_2),(x_3,y_3),\ldots,(x_n,y_n)\}$

5. Experimental Results

5.1 Camera calibration

For the camera calibration of ACTECK WM20 Webcam, we use a chess pattern of 7-row and 10-column, where each square is 23 mm in length. Twenty captures of the pattern calibration were taken to perform the calibration using Matlab. The extrinsic and intrinsic parameters values obtained from the camera calibration and the radial and tangential distortion values are presented in Table 2.

5.1.1 Vision system evaluation

To determine the accuracy of the vision system, we made measurements of patterns of the brand Mitutoyo of 40 mm, 30 mm, 10 mm, and 5 mm. A process similar to that described in Figure 3, was carried out to measure the known length. Twenty measurements were made of each of the patterns, changing the position between each capture. The presented results of this evaluation are presented in Table 3. As in Icasio-Hernandéz et al. (2019), the uncertainty *U* of the parameters is calculated as indicated by (24)

Table 2. Extrinsic, intrinsic and distortion parameters of the camera.

Parameter	Value	Parameter	Value	Parameter	Value	Parameter	Value
f_x	1438.6338	r_{11}	-0.9988	r_{12}	-0.0047	r_{13}	0.0496
f_y	1439.1291	r_{21}	0.0041	r_{22}	-0.9999	r_{23}	-0.0129
c_x	654.1560	r_{31}	0.0497	r_{32}	-0.0127	r_{33}	0.9987
c_y	369.6557	t_x	118.6608	t_y	41.3400	t_z	565.4370
s	0.0000	k_1	0.1630	k_2	0.0428	p_1	0.0064
p_2	0.0016	k_3	-2.1819				

Table 3. Results of the vision system evaluation.

Length of the pattern (mm)	Average measurement $(mm)^2$	Mean square error (mm)	Uncertainty
40	40.2301	0.0688	0.1294
30	30.2279	0.0651	0.1175
10	10.2029	0.0492	0.0921
5	5.2320	0.0595	0.0776

$$U = \sigma_x = \sqrt{\frac{\sum_{j=1}^{n} (q_j - \bar{q})^2}{n-1}} \tag{24}$$

Where n corresponds to the number of times the parameter or measurement is calculated, q_j represents the j-ith calculated value, and $\bar{q}$ is the average of the n measurements.

5.2 Robot calibration

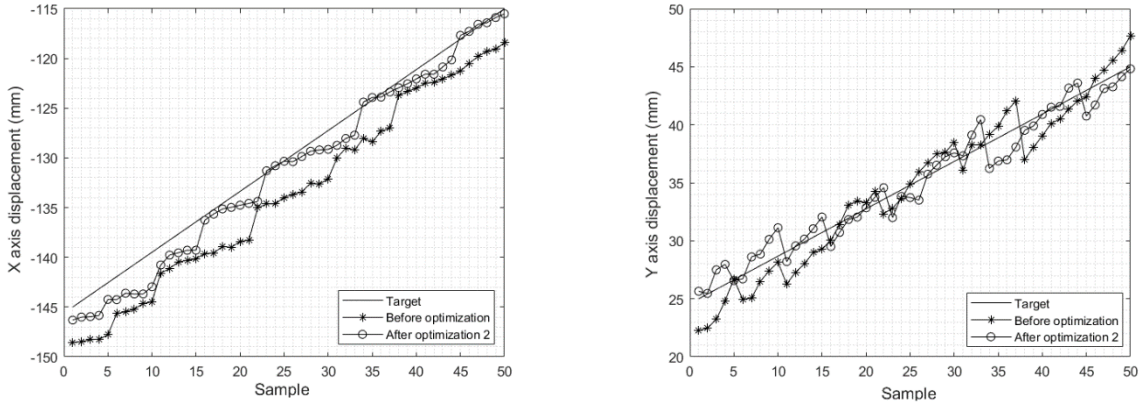
The values of the constants Δl_1, Δl_2, Δl_4, Δl_5, $\Delta\theta_1$, $\Delta\theta_2$, and $\Delta\theta_3$ obtained from the optimization carried out using Algorithm 1 in the three calibrations carried out are presented in Table 4.

Table 4. Robot calibration parameters performed using Algorithm 1 for three different calibration paths.

Parameter	First Calibration	Second Calibration	Third Calibration	Uncertainty
Δl_1 *(mm)*	6.6740	6.7015	6.6341	0.0339
Δl_2 *(mm)*	4.5566	4.5704	4.5967	0.0204
Δl_4 *(mm)*	-0.6543	-0.7074	-0.6110	0.0483
Δl_5 *(mm)*	0.9701	1.0208	1.0120	0.0271
$\Delta\theta_1$ *(°)*	1.0875	1.1354	1.0304	0.0526
$\Delta\theta_2$ *(°)*	1.0521	1.0037	1.0231	0.0244
$\Delta\theta_3$ *(°)*	4.7973	4.7563	4.7755	0.0205

Second calibration results and the modified inverse kinematics represented by (17) and (18) are used to calculate the angular values of the planar robot joints. As mentioned in section 4, the third path is used to analyze the result of the robot calibration. Fifty samples of the trajectory were made again with the compensation of the geometric parameters, using the procedure of section 3.2. Figure 5 shows the displacements along the axes x and y, respectively, before and after the modification to the inverse kinematic model of the manipulator.

Figure 6 shows the error defined by (19) for the validation trajectory generated before and after the modification to the kinematic model (obtained with data of second calibration) of the manipulator. The average error in the trajectory position was reduced from 4.0064 mm to 1.8186 mm, which means a reduction of 54.6076%.



(a) Path along *x*-axis. (b) Path along *y*-axis.

Figure 5. Cartesian trajectory of point *F* in *x*-axis and *y*-axis.

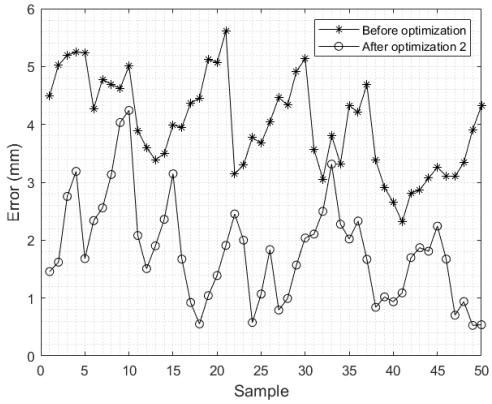


Figure 6. Error calculated using equation (19) for a linear trajectory.

6. Conclusions and future work

This document shows that by measuring the trajectory of the end effector at a certain number of points and evaluating the error of the robot end-effector with respect to the ideal inverse kinematics, an algorithm can be established to determine the variations Δl_i and $\Delta \theta_i$ necessary to update the geometrical parameters of the inverse kinematic model. The measurements were made with a vision system, which has an error of less than 0.25 mm. The vision system error was calculated using 40 mm, 30 mm, 10 mm and 5 mm standards gauge. Therefore, the measurements made are sufficiently accurate for our implementation.

Calibration was performed three times using two circles of 25 mm and 15 mm of radius. The first calibration using the points belonging to the largest radius circle, the second calibration using the points of the smallest radius circle. In contrast, the third calibration uses both paths. The results of the three calibrations were very similar, generating small uncertainties in the geometric parameters. The inverse kinematic model was updated using the geometric dimensions obtained from the calibration, and it was found that the error decreased in all the calibrations. The straight-line path tracking was presented on the *xy*-plane before parameter update and after robot calibration. The second calibration was the one that presented the greatest reduction in the mean error (54.6076 %). It is shown that the coordinates *x* of the robot's end-effector are better adjusted to the desired trajectory, keeping the coordinates *y* around the trajectory. Therefore, it was shown that using a single camera, the trajectory tracking performance of a planar robot can be evaluated, this being a handy tool for prototyping. However, we can use the same process and add a second camera to track trajectories in three-dimensional space.

Using single camera allows measurements to be made in the plane where the robot is traveling. However, to measure a trajectory in three-dimensional space (with the robot's five degrees of freedom enabled), it is necessary to add a second camera. In this way, the spatial coordinates of the robot end-effector can be found. The markers used in this implementation are circles located on the axes of rotation of the robot. This procedure can be generalized by using a sphere at the robot end-effector. Since both cameras would capture a circle, the sphere's center in 3D space coinciding with the centers of each circle in the captured images. However, our robot uses frequency DC servomotors, which have a high mechanical play and low resolution regarding the control of the angular position, making it necessary to use a robot with better actuators to experiment.

Finally, only the optimization of the error function was carried out for the variations in the length of links and mounting angles, that is, geometric parameters of the D-H model. However, the error in the position is also caused by non-geometric errors, such as end-effector charge, deformation of links due to the load, hysteresis, mechanical play, and dead zone of the actuators. They are a future work area the compensation of these phenomena during the execution of trajectories.

7. REFERENCES

Arredondo-Soto, M., García-Murillo, M. A., Cervantes-Sánchez, J. J., Torres, F. J., Moreno-Avalos, H. A., 2021. Identification of geometric parameters of a parallel robot by using a camera calibration technique. J Mech Sci Technol 35, 729–737. DOI: 10.1007/s12206-021-0133-z

Banga, V. K., Kumar, R., Singh, Y., 12 2009. Fuzzy-genetic optimal control for four degree of freedom robotic arm movement. World Academy of Science, Engineering and Technology 60. DOI: 10.5281/zenodo.1057761

Costa, J., Antonio Urzedo Machado, T., Carneiro, M., 2020. Implementation and validation of thor 3d printed open source robotic arm. IEEE Latin America Transactions 18 (05), 907–913. DOI: 10.1109/TLA.2020.9082919

Duka, A.-V., 2014. Neural network based inverse kinematics solution for trajectory tracking of a robotic arm. Procedia Technology 12, 20–27, the 7th International Conference Interdisciplinarity in Engineering, INTER-ENG 2013, 10-11 October 2013, Petru Maior University of Tirgu Mures, Romania. DOI: 10.1016/j.protcy.2013.12.451

Durović, P., Grbić, R., Cupec, R., 2017. Visual servoing for low-cost scara robots using an RGB-D camera as the only sensor. Automatika 58 (4), 495–505. DOI: 10.1080/00051144.2018.1461771

Elsisi, M., Mahmoud, K., Lehtonen, M., Darwish, M. M. F., 01 2021. An improved neural network algorithm to efficiently track various trajectories of robot manipulator arms. IEEE Access PP, 1–1. DOI: 10.1109/ACCESS.2021.3051807

González-Barbosa, E.-A., González-Palacios, M. A., Aguilera-Cortés, L. A., González-Barbosa, J.-J., Serrano-Rubio, J. P., Colin Robles, J. Á., 2022. Energy consumption analysis for an adaptive prototype of 3r industrial robot. Robotica 40 (11), 4143–4168. DOI: 10.1017/S0263574722000832

Icasio-Hernández, O., Hurtado-Ramos, J., Gonzalez-Barbosa, J., 2019. Calibration of endoscopic systems coupled to a camera and a structured light source. MAPAN 34, 143–157. DOI: 10.1007/s12647-018-0288-y

International Federation of Robotics (IFR), 2022. Ifr presents world robotics report 2022. https://www.automation.com/en-us/articles/ october-2022/ifr-presents-world-robotics-report-2022 [Accessed: (May eleventh, 2023)].

Li, F., Zeng, Q., Ehmann, K. F., Cao, J., Li, T., 2019. A calibration method for overconstrained spatial translational parallel manipulators. Robotics and Computer-Integrated Manufacturing 57, 241–254. DOI: 10.1016/j.rcim.2018.12.002

Liu, Y., Li, Y., Zhuang, Z., Song, T., 2020. Improvement of robot accuracy with an optical tracking system. Sensors 20 (21). URL: https://www.mdpi.com/1424-8220/20/21/6341 DOI: 10.3390/s20216341

Olsson, T., Haage, M., Kihlman, H., Johansson, R., Nilsson, K., Robertsson, A., Björkman, M., Isaksson, R., Ossbahr, G., Brogårdh, T., feb 2010. Cost-efficient drilling using industrial robots with high-bandwidth force feedback. Robot. Comput. Integr. Manuf. 26 (1), 24–38. DOI: 10.1016/j.rcim.2009.01.002

Roth, Z., Mooring, B., Ravani, B., 1987. An overview of robot calibration. IEEE Journal on Robotics and Automation 3 (5), 377–385. DOI: 10.1109/JRA.1987.1087124

Santolaria, J., Yagüe, J.-A., Jiménez, R., Aguilar, J.-J., 2009. Calibration-based thermal error model for articulated arm coordinate measuring machines. Precision Engineering 33 (4), 476–485. DOI: 10.1016/j.precisioneng.2009.01.002

Stepanova, K., Rozlivek, J., Puciow, F., Krsek, P., Pajdla, T., Hoffmann, M., 2022. Automatic self-contained calibration of an industrial dual-arm ro- bot with cameras using self-contact, planar constraints, and self-observation. Robotics and Computer-Integrated Manufacturing 73, 102250. DOI: 10.1016/j.rcim.2021.102250

Wang, W., Tian, W., Liao, W., Li, B., 2021. Pose accuracy compensation of mobile industry robot with binocular vision measurement and deep belief network. Optik 238, 166716. DOI: 10.1016/j.ijleo.2021.166716

Wang, X., Huang, J., Song, H., 2023. Robot-world and hand-eye calibration based on quaternion: A new method and an extension of classic methods, with their comparisons. Mechanism and Machine Theory 179, 105127. DOI: 10.1016/j.mechmachtheory.2022.105127

Yan, Z., Gao, G., Na, J., Liu, F., 2021. On the measuring system for the position of industrial robots' end-effectors based on monocular vision. In: 2021 IEEE 10th Data Driven Control and Learning Systems Conference (DDCLS). pp. 372–377. DOI: 10.1109/DDCLS52934.2021.9455565

Zha, J., Zhang, H., Chen, Y., 2020. A strategy to evaluate and minimize parallelism errors of a rotor system in a precision rotary table. Int J Adv Manuf Technol 106, 3641–3648. DOI: 10.1007/s00170-019-04828-2

ZHEKOV, Z., 2020. Inverse kinematics neural approximation and neural control of two link planar robot. In: 2020 International Conference Automatics and Informatics (ICAI). pp. 1–5. DOI: 10.1109/ICAI50593.2020.9311376

Zu, H., Chen, X., Chen, Z., Wang, Z., Zhang, X., 2021. Positioning accuracy improvement method of industrial robot based on laser tracking measurement. Measurement: Sensors 18, 100235. DOI: https://doi.org/10.1016/j.measen.2021.100235