

Desarrollo de un brazo robótico manipulador de cuatro grados de libertad con reconocimiento de objetos por visión artificial

Iván Darío Mesa-Parra¹

Santiago Vallejo-Silva²

Dagoberto Mayorca-Torres³

Richard Geovanni Moran-Perafán⁴

Resumen

Los robots son herramientas de alta tecnología que han tomado cada vez mayor importancia en entornos industriales y académicos, ya que generalmente facilitan tareas tediosas o peligrosas para los seres humanos. Particularmente, en el ámbito académico existen varios tipos de robots que facilitan el aprendizaje de los principios de la robótica y programación; sin embargo, muchos de ellos son muy costosos o de funciones limitadas. Por tanto, se presenta la integración de un brazo robótico a un sistema de reconocimiento de objetos con visión artificial. El brazo robótico se basa en un diseño *open source* del modelo Moveo de BCN3D, cuyos componentes fueron construidos y ensamblados por ingenieros mecánicos de la Universidad Mariana. El brazo consta de cuatro grados de libertad impulsados por motores paso a paso y controlados por un sistema embebido, el cual fue programado en *Python*, se usó las librerías *pymata*, *Open CV*, *Numpy*, *Tensorflow* y *Tkinter*. A este brazo se le incorporó un sistema de reconocimiento de imágenes, integrado por una cámara y un sistema de iluminación que utiliza un algoritmo de clasificación de imágenes. Una red neuronal artificial de tipo convolucional es utilizada para el reconocimiento de objetos como figuras geométricas. Con base en 50 ensayos, se estimó

¹Estudiante del Programa de Ingeniería Mecatrónica, Universidad Mariana. Correo electrónico: ivandar97@hotmail.com

²Estudiante del Programa de Ingeniería Mecatrónica, Universidad Mariana. Correo electrónico: santiagosvs00@gmail.com

³Magíster en Ingeniería. Profesor del Programa de Ingeniería Mecatrónica, Universidad Mariana. Correo electrónico: dmayorca@umariana.edu.co

⁴Doctor en Ingeniería. Profesor del Programa de Ingeniería Mecatrónica, Universidad Mariana. Correo electrónico: rmoran@umariana.edu.co

que la precisión de reconocimiento fue del 95 %. Adicionalmente, se determinó un error cuadrático medio (RMSE) de ± 2 mm en el posicionamiento de la pinza en pruebas de selección y clasificación de objetos de diferentes tamaños y formas.

Palabras clave: robot manipulador; visión artificial; robot Moveo BC3D; robótica educativa.

Development of a manipulative robotic arm with four degrees of freedom with artificial vision object recognition

Abstract

Robots are high-tech tools that have become increasingly important in industrial and academic settings, as they generally facilitate tedious or dangerous tasks for humans. Particularly, in the academic field, there are several types of robots that facilitate the learning of the principles of robotics and programming; however, many of them are very expensive or limited in their functions. Therefore, the integration of a robotic arm to an object recognition system with artificial vision is presented, based on an open-source design of the BCN3D Moveo model, whose components were built and assembled by mechatronic engineers from the Mariana University, which consists of four degrees of freedom driven by stepper motors and controlled by an embedded system, programmed in Python. Pymata, Open CV, Numpy, Tensorflow and Tkinter libraries were used. An image recognition system was incorporated, consisting of a camera and a lighting system that uses an image classification algorithm. A convolutional artificial neural network is used for the recognition of objects such as geometric figures. Based on 50 trials, the recognition accuracy was estimated to be 95%. Additionally, a mean square error of ± 2 mm was determined in the positioning of the clamp in tests of selection and classification of objects of different sizes and shapes.

Keywords: manipulator robot; artificial vision; Moveo BC3D robot; educational robotics.

Desenvolvimento de um braço robótico manipulativo com quatro graus de liberdade com reconhecimento de objetos por visão artificial

Resumo

Os robôs são ferramentas de alta tecnologia que se tornaram cada vez mais importantes em ambientes industriais e acadêmicos, pois geralmente facilitam tarefas tediosas ou perigosas para os humanos. Particularmente, no meio acadêmico, existem vários tipos de robôs que facilitam o aprendizado dos princípios da robótica e da programação; entretanto, muitos deles são muito caros ou limitados em suas funções. Para tanto, é apresentada a integração de um braço robótico a um sistema de reconhecimento de objetos com visão artificial, a partir de um projeto open-source do modelo BCN3D Moveo, cujos componentes foram construídos e montados por engenheiros mecatrônicos da Universidade Mariana, composta por quatro graus de liberdade acionados por motores de passo e controlados por um sistema embarcado, programado em Python. Foram utilizadas as bibliotecas Pymata, Open CV, Numpy, Tensorflow e Tkinter. Foi incorporado um sistema de reconhecimento de imagens, composto por uma câmera e um sistema de iluminação que utiliza um algoritmo de classificação de imagens. Uma rede neural artificial convolucional é usada para o reconhecimento de objetos como figuras geométricas. Com base em 50 testes, a precisão do reconhecimento foi estimada em 95%. Além disso, foi determinado um erro quadrático médio de ± 2 mm no posicionamento da pinça em testes de seleção e classificação de objetos de diferentes tamanhos e formas.

Palavras-chave: robô manipulador; visão artificial; Robô Moveo BC3D; robótica educacional.

INTRODUCCIÓN

La robótica es una ciencia multidisciplinaria cuyo objetivo es desarrollar máquinas autónomas que emulan parcialmente el comportamiento de un ser humano o animal de acuerdo a la programación empleada (Revista de Robots, 2020). Los brazos robóticos son las máquinas más populares en el área de la robótica. Este tipo de robots se utilizan para llevar a cabo tareas repetitivas y peligrosas, pero también se pueden usar para fines académicos como la realización de prácticas de control y electrónica (Flexbot, s.f.). Estos sistemas electromecánicos se componen de eslabones y articulaciones que permiten realizar movimientos en un espacio limitado. La movilidad de un brazo robótico se determina según la cantidad de articulaciones independientes, con las cuales se define el número de grados de libertad, valor que representa la capacidad de un robot de alcanzar una posición y orientación determinada en el espacio (Barrientos et al., 2007).

Diseñar un brazo robótico implica definir la geometría de los eslabones y la posición de las articulaciones, así como los mecanismos de transmisión de movimiento y los elementos de control. Particularmente, en el controlador deben programarse los modelos de cinemática directa e inversa, que permiten alcanzar una posición y orientación de acuerdo a la trayectoria o posición definida por el usuario (RoboDK, s.f.). Existen diferentes métodos para definir la cinemática de un robot, el más usado es el algoritmo de Denavit Hartenberg, que hace uso de matrices homogéneas y convenciones para definir la orientación de los ejes en cada articulación (Barrientos et al., 2007). Un sistema robótico puede trabajar junto con técnicas de visión artificial, las cuales mejoran la autonomía del robot para realizar tareas repetitivas, como la selección y clasificación de objetos. Además, las técnicas de visión artificial permiten llevar a cabo tareas de localización de objetos por color y forma (Culjak et al., 2012), algunos robots modernos incorporan aprendizaje automático, lo cual permite realizar predicciones y tomar decisiones (El País, 2017).

El uso de brazos robóticos en la industria se extiende en diferentes sectores. En el área de la manufactura es usado para la manipulación de residuos tóxicos, y según la utilización del robot, se define su capacidad de carga y los materiales de construcción.

Este proyecto se enfoca en la construcción de un robot con fines académicos, para apoyar los procesos de aprendizaje de estudiantes de la facultad de ingeniería de la Universidad Mariana. El diseño se enfoca en el uso de componentes y materiales disponibles en Colombia y una interfaz desarrollada con software de código abierto.

Métodos

Construcción del brazo robótico

Se seleccionó un brazo robótico *open source* para implementarlo en el proyecto, para lo cual se consideraron algunos requerimientos como: la estética, el tamaño, tipos de motor, materiales, disponibilidad de códigos y planos de construcción. Después de considerar varias alternativas, el brazo seleccionado fue MOVEO de BCN3D, el cual cumple con todos los requerimientos mencionados y cuenta con la facilidad de adecuarse a entornos como laboratorios y mesas de trabajo (ver Figura 1). En el diseño original, el brazo tiene 5 grados de libertad, pero para las finalidades del proyecto se redujo a 4 grados de libertad, que se consideraron suficientes para alcanzar la posición de un objeto en tres dimensiones con una orientación vertical de la pinza.

Figura 1

Brazo robótico BCN3D MOVEO

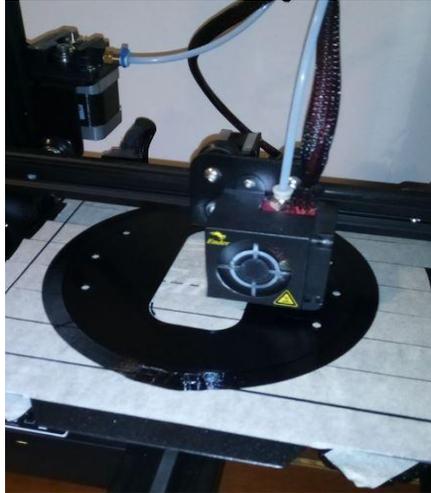


Fuente: BCN3D, s.f.

Los planos de partes y ensamble de los componentes del brazo se encuentran disponibles en un repositorio de archivos (BCN3D, s.f.), en el cual se listan también elementos mecánicos comerciales como tornillos, rodamientos, bandas y poleas; así como los componentes electrónicos, entre los cuales se encuentra la fuente de alimentación, motores paso a paso y placas electrónicas. La mayoría de los componentes no comerciales que constituyen hardware del brazo 3D fueron impresos, se usó PLA con una densidad del 60 %. El equipo usado para la impresión es Ender 3 Pro (Ingenio Triana, s.f.). En la Figura 2 se observa el proceso de impresión 3D de un eslabón del brazo.

Figura 2

Impresión 3D de primer eslabón



Una vez elaboradas las piezas y adquiridos los componentes comerciales, se realizó el ensamblaje del sistema mecánico y electrónico. La verificación del ensamble se realizó a través de pruebas de movimiento usando los motores paso a paso. En la Figura 3 se observa el ensamblaje del brazo robótico.

Figura 3

Brazo robótico ya ensamblado



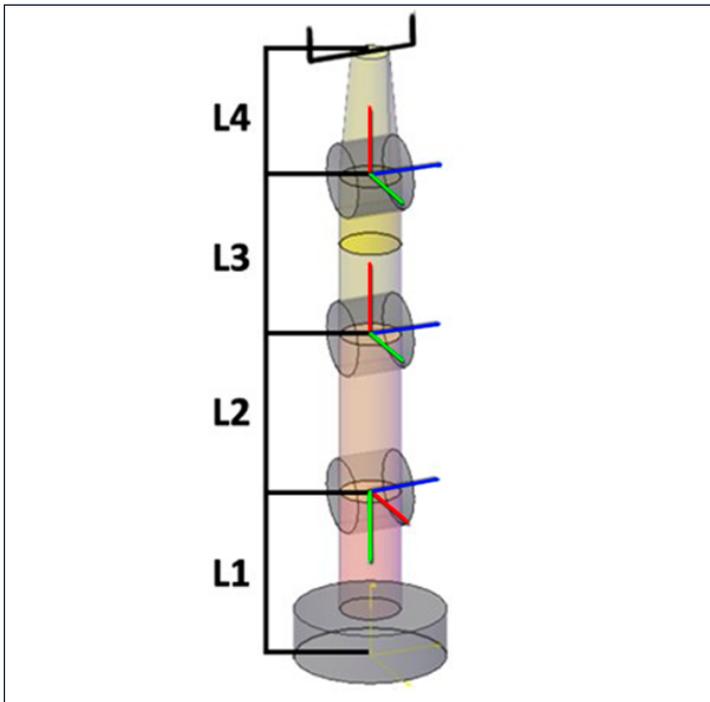
Finalmente, el brazo robótico se instaló sobre una mesa rectangular de madera y acrílico, con unas dimensiones de 98 cm * 58 cm; tamaño suficiente para aprovechar todo el espacio de trabajo del robot.

Análisis de Cinemática directa

En esta etapa se procede a realizar el análisis cinemático del robot, aplicando el algoritmo de Denavit-Hartenberg para la ubicación del extremo del robot. En la Figura 4 se detalla la ubicación de cada eje de coordenada, correspondiente a cada articulación del brazo robótico.

Figura 4

Orientaciones de los ejes locales en cada articulación del brazo robótico MOVEO. Los ejes azules, verdes y rojos representan los ejes locales respectivamente



Los parámetros de Denavit Hartenberg que se determinaron para el brazo MOVEO se presentan en la Tabla 1.

Tabla 1*Parámetros de Denavit Hartenberg*

Articulación		a_i	d_i	α
I	q_1	L_1 (231.5 mm)	0	$-\pi/2$
II	$q_2 - \pi/2$	0	L_2 (221.1 mm)	0
III	q_3	0	L_3 (222.5 mm)	0
IV	q_4	0	L_4 (168 mm)	0

Donde: q_1, q_2, q_3 y q_4 son los ángulos de la base, hombro, codo y muñeca del brazo robot, respectivamente. Así mismo, L_1, L_2, L_3 y L_4 corresponden a las dimensiones de los cuatro eslabones del robot (ver Figura 4). A partir de los parámetros presentados en la Tabla 1, es posible deducir la matriz de transformación de cada articulación i , al remplazar cada parámetro de la matriz (θ, d, a, α) , según corresponda a la columna de la Tabla 1.

$$A_i = \begin{bmatrix} c\theta_i & -\alpha_i * s\theta_i & s\alpha_i * c\theta_i & a_i * c\theta_i \\ s\theta_i & \alpha_i * s\theta_i & -s\alpha_i * c\theta_i & a_i * s\theta_i \\ 0 & s\alpha_i & c\alpha_i & d_i \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (1)$$

$$A_1 = \begin{bmatrix} \cos(q_1) & 0 & -\sin(q_1) & 0 \\ \sin(q_1) & 0 & \cos(q_1) & 0 \\ 0 & -1 & 0 & L_1 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (2)$$

$$A_2 = \begin{bmatrix} \cos(q_2 - \frac{\pi}{2}) & -\sin(q_2 - \frac{\pi}{2}) & 0 & \cos(q_2 - \frac{\pi}{2}) * L_2 \\ \sin(q_2 - \frac{\pi}{2}) & \cos(q_2 - \frac{\pi}{2}) & 0 & \sin(q_2 - \frac{\pi}{2}) * L_2 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (3)$$

$$A_3 = \begin{bmatrix} \cos(q_3) & -\sin(q_3) & 0 & \cos(q_3) * L_3 \\ \sin(q_3) & \cos(q_3) & 0 & \sin(q_3) * L_3 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (4)$$

$$A_4 = \begin{bmatrix} \cos(q_4) & -\sin(q_4) & 0 & \cos(q_4) * L_4 \\ \sin(q_4) & \cos(q_4) & 0 & \sin(q_4) * L_4 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (5)$$

Al tener todas las matrices de transformación A_i , se encontró la matriz de transformación final del sistema T , como la multiplicación de las matrices presentadas en las ecuaciones (2), (3), (4) y (5), en el orden presentado en la ecuación (6).

$$T = A_1 * A_2 * A_3 * A_4 \tag{6}$$

$$T = \begin{bmatrix} nx & ox & ax & px \\ ny & oy & ay & py \\ nz & oz & az & pz \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{7}$$

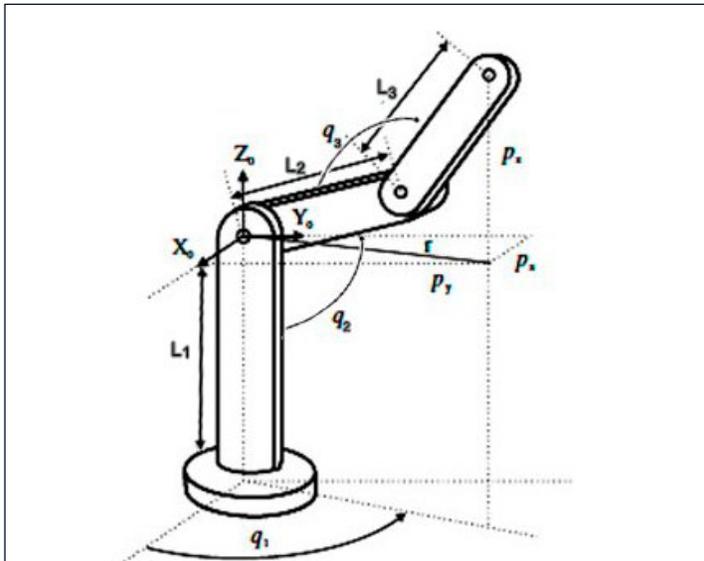
La matriz (7) representa la transformación global del sistema, en ella la columna 4 está formada por las componentes p_x , p_y y p_z que representan los componentes cartesianos de traslación, desde el sistema coordenado de la base hasta el sistema de la pinza. La cinemática inversa resuelve la configuración que debe adoptar el robot para llegar a una posición y orientación del extremo conocidas.

Cinemática inversa de ubicación

Se utilizó desacoplamiento cinemático para resolver, primero, la cinemática de ubicación, correspondiente a los tres primeros grados de libertad. La cinemática de orientación se presenta posteriormente para obtener la orientación de la muñeca. El brazo Moveo es un brazo angular o antropomorfo y las ecuaciones de cinemática de ubicación han sido desarrolladas por varios autores (Barrientos et al., 2007). La Figura 5 presenta un esquema simplificado de la geometría de este tipo de brazo y los principales parámetros geométricos que definen la cinemática.

Figura 5

Parámetros que describen la cinemática de un robot articular



Fuente: Barrientos et al., 2007.

Las ecuaciones que definen los ángulos de las articulaciones q_1 , q_2 y q_3 , en función de las longitudes de los eslabones L_1 , L_2 y L_3 y las componentes cartesianas de traslación p_x , p_y y p_z se presentan en las ecuaciones 8, 9 y 10.

$$q_1 = \arctg\left(\frac{p_y}{p_x}\right) \tag{8}$$

$$q_2 = \arctg\left(\frac{p_z}{\pm\sqrt{p_x^2+p_y^2}}\right) - \arctg\left(\frac{L_3*\sen q_3}{L_2+L_3*\cos q_3}\right) \tag{9}$$

$$q_3 = \arctg\left(\frac{\pm\sqrt{1-\cos^2 q_3}}{\cos q_3}\right) \tag{10}$$

Cinemática inversa de orientación

Para definir el ángulo de la articulación q_4 se consideró la orientación de la garra, para que permanezca vertical respecto al plano de la mesa, por lo tanto, se hizo el análisis geométrico y se determinó la ecuación (11).

$$q_4 = 180 - q_3 - q_2 \tag{11}$$

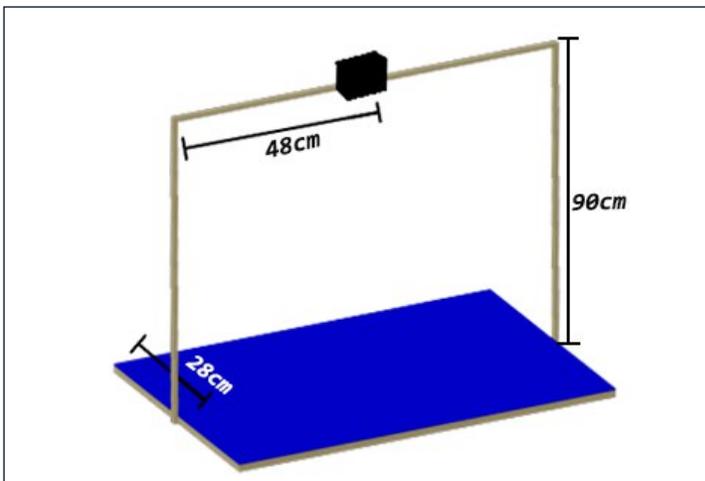
De esta manera se definen los ángulos de cada articulación q_1 , q_2 , q_3 y q_4 en función de la posición del objeto de interés

Implementación del sistema óptico

Se implementó un sistema óptico en la parte superior de la mesa, posición estratégica que permite la captura de todo el espacio de trabajo del robot (ver Figura 6).

Figura 6

Ubicación del sistema



Debido a que las imágenes capturadas presentan una distorsión por la curvatura del lente de la cámara, fue necesario realizar un proceso de calibración, el cual se realizó con base en el método Zhangs (Unipython, 2018) que utiliza un tablero de ajedrez y la librería *Open CV* para reducir esta distorsión y así obtener una imagen calibrada.

Verificación de funcionamiento

Experimento 1 – Prueba de ubicación manual

Para verificar el correcto funcionamiento de los motores y la programación del controlador, se programaron posiciones dentro del espacio de trabajo del robot, las cuales se contrastaron con respecto a mediciones experimentales realizadas en la ubicación de la garra. Las mediciones se realizaron utilizando una regla con una resolución de ± 1 mm. Se compararon los resultados en cada eje de movimiento y se utilizó la fórmula del error cuadrático medio para obtener el error en la ubicación del brazo robótico, a continuación, se muestra la fórmula.

$$ECM = \frac{1}{n} \sum_{i=1}^n (y_i - \bar{y}_i)^2 \quad (12)$$

Donde ECM es el error cuadrático medio, n es el número de pruebas realizadas y y_i es la diferencial de cada prueba.

Experimento 2 – Prueba de ubicación con visión artificial

Se ubicaron diferentes objetos sobre el espacio de trabajo y se usaron las coordenadas dadas por el sistema de visión artificial, para ubicar el brazo robótico sobre cada uno de los objetos presentes en el espacio de trabajo. Se realizaron 30 ensayos y se registraron en una tabla los resultados, 1 o 0, según, si la garra del brazo se ubicó correctamente o no sobre los objetos, respectivamente.

Experimento 3- Las pruebas de la garra

Al igual que en el experimento 2, se ubicaron diferentes objetos sobre el espacio de trabajo. Se obtuvieron las coordenadas de los objetos y su clasificación por tipo, utilizando el algoritmo de clasificación de imágenes, esto permite que internamente se realice una secuencia de posiciones, la cual contiene todas las coordenadas en las que el brazo robótico debe ubicarse para clasificar todos los objetos; se inició las secuencias de clasificación y se ingresó en una tabla cuántas reubicaciones exitosas se presentaron. Estas pruebas se realizaron 20 veces con el objetivo de identificar si el sistema de clasificación de imágenes funciona correctamente y, además, saber si la garra del brazo robótico es capaz de levantar los objetos.

Resultados

El resumen de los resultados de la primera prueba se muestra en la Tabla 2. Con base en la fórmula del error cuadrático medio se determinó un error en el posicionamiento de la pinza de $\pm 3.44\text{mm}$ y $\pm 3.93\text{mm}$ en los ejes X y Y, respectivamente.

Tabla 2

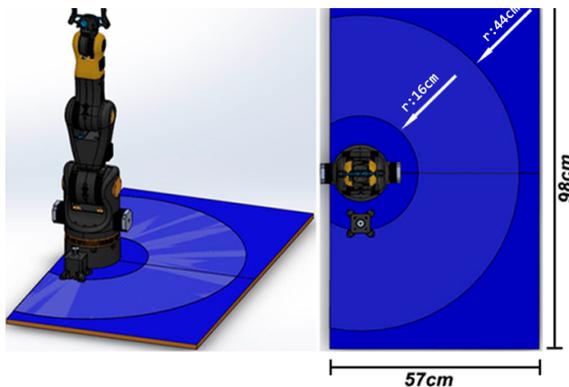
Pruebas de exactitud de posicionamiento

Prueba	Valores programados (mm)		Valores medidos (mm)	
	X	Y	X	Y
1	250	0	242	0
2	250	130	248	125
3	250	260	254	254
4	250	310	253	306
5	340	0	335	0
6	340	50	335	45
7	340	150	340	145
8	340	250	341	249
9	400	0	395	0
10	400	50	397	50

Utilizando una herramienta de medición se identificó la distancia mínima y máxima por la cual el brazo robótico es capaz de desplazarse sin sobrepasar los límites físicos, lo cual constituye el espacio de trabajo, y que dibujado sobre el plano de la mesa se muestra en la Figura 7.

Figura 7

Área de trabajo

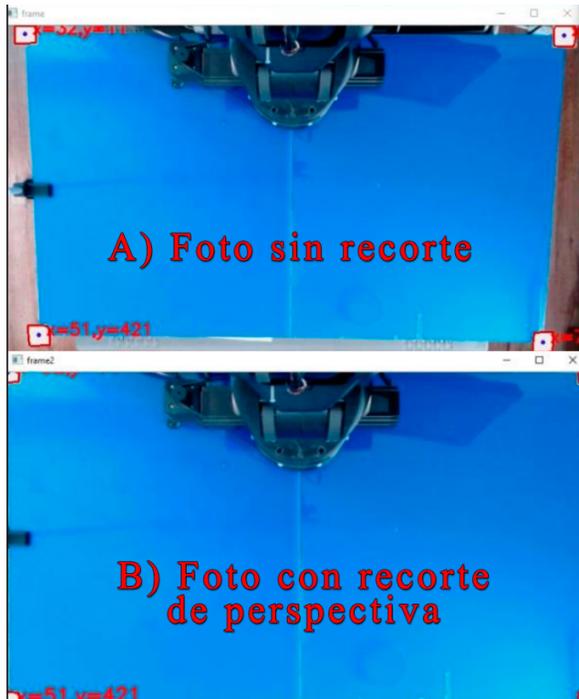


Implementación de técnicas de Machine Learning

Una vez implementado el sistema óptico y con la posibilidad de realizar capturas, las imágenes fueron procesadas para detectar las cuatro esquinas de la mesa, donde se ubicaron cuatro pequeños cuadrados cuyo color blanco sirvió para diferenciar estas posiciones. Teniendo las cuatro esquinas identificadas se realizó un recorte de perspectiva para así obtener una nueva imagen completa del espacio de trabajo. La detección de esquinas y el recorte del espacio de trabajo se puede observar en la Figura 8.

Figura 8

A) Foto sin recorte, B) Foto con recorte de perspectiva



Se utilizó un fondo azul en la mesa de trabajo para poder detectar cualquier otro objeto sobre ese espacio. Se realizó una binarización de la imagen para poder detectar contornos y centros de masa de los objetos, obteniendo así la localización de estos en la imagen. Para obtener las medidas correctas se realizó la conversión de pixeles a milímetros, en la Figura 9 se puede observar los objetos localizados en la imagen con sus correspondientes coordenadas X y Y.

Figura 9

Ejemplo de coordenadas obtenidas



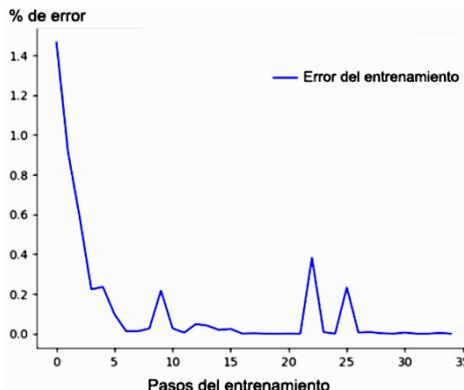
Con la ubicación de los objetos sobre la mesa de trabajo fue necesario realizar un modelo de clasificación de imágenes para reconocer la forma del objeto. Se definió una base de datos con fotos de los objetos que se esperaban encontrar. Para este proyecto se realizó una base de datos compuesta de 50 fotografías por cada objeto y un total de 4 objetos, lo que resultó en un total de 200 fotos. Los objetos tomados al azar para formar la base de datos fueron los siguientes: un yoyo blanco con un diámetro de 6 centímetros, un yoyo morado con un diámetro de 5 centímetros, una esfera de icopor roja con un diámetro de 4.5 centímetros y un driver de motor paso a paso de 5 cm por 7cm, todos estos objetos tienen una altura aproximada de 3 centímetros

Se realizó el entrenamiento del modelo de clasificación de imágenes utilizando un algoritmo de *Python*, que permite crear un sistema que contiene toda la información de la base de datos; de esta manera fue capaz de reconocer los objetos de una forma autónoma. El resultado del entrenamiento mostró una precisión en el reconocimiento del 95 %, porcentaje que fue corroborado en futuras pruebas.

El proceso de entrenamiento del modelo de clasificación de imágenes se observa en la Figura 10, la cual muestra la disminución del error en la clasificación, valor que debe ser cercano a cero. En el caso de este proyecto se obtuvo un error del 0.07 %.

Figura 10

Disminución del porcentaje de error durante el entrenamiento

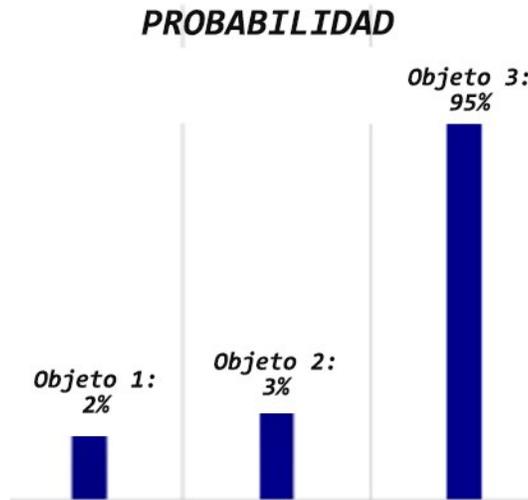


En las pruebas del modelo se utilizó una imagen como entrada y se obtuvo una lista de probabilidades; la clase que obtenga la probabilidad más alta es a la que el objeto pertenece.

En la Figura 11 se observa el resultado de una prueba en la que se utilizó una imagen de un yoyo como entrada. El modelo arrojó una probabilidad del 95 % a la clase número 3, que sí corresponde al objeto de la imagen.

Figura 11

Objeto detectado por el modelo de clasificación de imágenes



Validación funcionamiento del brazo robótico mediante Machine Learning

Una vez listo el sistema de localización y de identificación, se procedió a unir el sistema robótico con el sistema de visión artificial y *machine learning*, el cual fue probado para verificar su funcionamiento. En la primera prueba se identificó la exactitud del brazo robótico, se ubicaron objetos sobre la mesa de trabajo; usando el sistema de visión artificial se obtuvieron las coordenadas del objeto, en las cuales el brazo robótico tuvo que ubicarse haciendo uso de la cinemática inversa presentada previamente. Algunas de estas coordenadas y su resultado se muestran en la Tabla 3, donde se presentan las coordenadas (X, Y) del centroide de cada objeto, las medidas fueron en mm.

Tabla 3

Pruebas de coordenadas

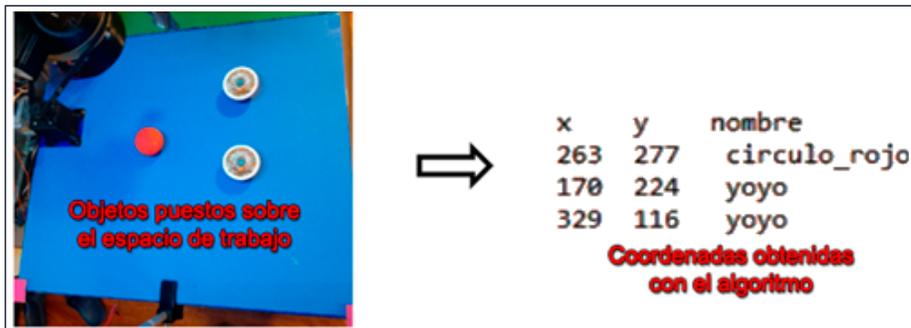
Prueba	X (mm)	Y (mm)	Llegó
1	120	302	Sí
2	286	236	Sí
3	314	53	Sí
4	88	315	Sí
5	255	287	No
6	147	300	Sí
7	268	273	Sí
8	44	261	Sí
9	334	204	Sí

Se recopilaron los resultados de la prueba y se procedió a realizar una pequeña calibración de aproximadamente 1 grado en la articulación 1, ya que esto causaba que en algunas pruebas el brazo robótico no se ubicara correctamente.

Con la correcta ubicación del brazo robótico, utilizando el sistema de visión artificial, se implementó el sistema de *machine learning*, para poder realizar la clasificación de objetos. A cada objeto se le asignó una coordenada en la que será depositado, así se obtuvo una serie de coordenadas para que el brazo robótico clasifique cada objeto en donde corresponde.

Figura 12

Ejemplo de coordenadas obtenida a partir del algoritmo



El componente en Z se fijó para todos los objetos en 25 mm, desde la superficie de trabajo hasta la punta de la garra del brazo robótico. Con la posibilidad de generar

secuencias, se realizaron las pruebas de exactitud de todo el sistema, en donde se ubicaron objetos sobre la mesa, se ejecutó la función de obtención de coordenadas y la de clasificación de imágenes para obtener todos los movimientos que el brazo robótico debe hacer; se inició cada una de estas secuencias y se ingresó en una tabla la cantidad de ubicaciones exitosas por prueba, los resultados se pueden ver en la Tabla 4.

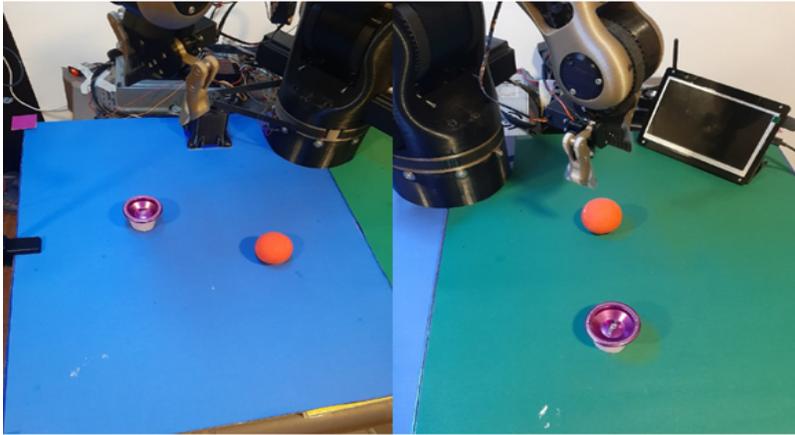
Tabla 4
Pruebas finales de reubicación de objetos

Prueba	Reubicación de objeto exitosa		Total de reubicaciones exitosas
	Objeto 1	Objeto 2	
1	1	1	2
2	1	1	2
3	1	1	2
4	1	1	2
5	1	1	2
6	1	1	2
7	0	1	1
8	1	1	2
9	1	1	2
10	1	1	2

Se pudo observar en las pruebas que el error que tiene el brazo robótico al momento de ubicarse no interfiere en la gran mayoría de los casos, ya que la garra puede sujetar a los objetos para ser reubicados. En algunos casos los objetos no pueden ser agarrados debido a que sufren una pequeña traslación causada por las vibraciones del brazo robótico al moverse. Dichas vibraciones son causadas por los motores paso a paso con los que cuenta el brazo robótico, una solución posible sería reducir la velocidad con la cual funciona todo el sistema o diseñar una mesa de trabajo más robusta que ayude a reducir estas vibraciones. En la Figura 13 se muestran dos objetos en la posición inicial, en la parte izquierda, y en la posición después de haber sido reubicados por el brazo robótico, en la parte derecha.

Figura 13

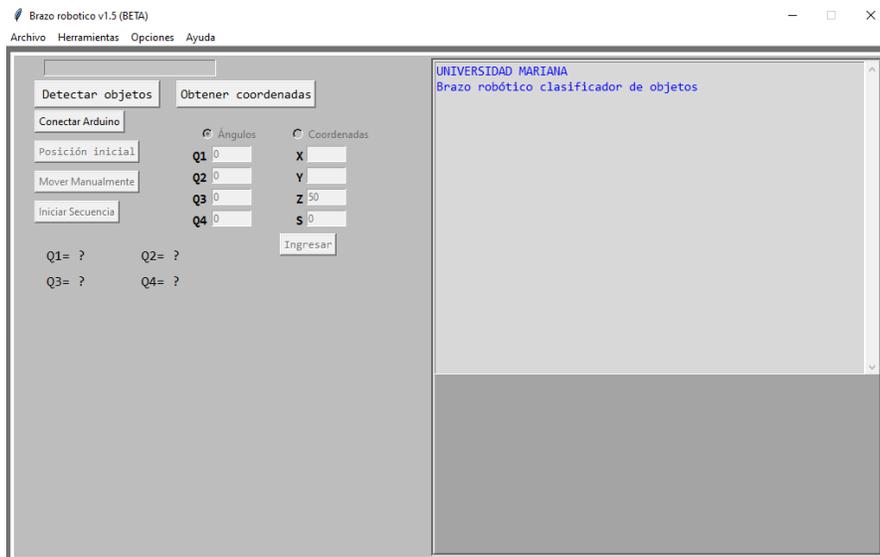
Objetos en posición inicial y objetos reubicados



En la Figura 14 se observa la interfaz de control del brazo robótico y el sistema de adquisición de datos.

Figura 14

Interfaz Gráfica



Conclusiones

Es posible ubicar la herramienta de agarre del brazo robótico en un punto en el espacio con el uso de coordenadas cartesianas, gracias al análisis de la cinemática inversa.

El método de clasificación de imágenes no presenta problemas cuando se quiere detectar a qué clase pertenece un objeto puesto sobre el espacio de trabajo.

Un modelo de detección de objetos, como el que se pensó usar en este proyecto, no es capaz de detectar objetos que no ocupen al menos un 8 % de la imagen, aproximadamente; para solucionarlo es posible utilizar un modelo con más capacidades y ejecutarlo en una computadora.

El uso de acrílico como cubierta de la mesa de trabajo no es la mejor opción, ya que refleja la luz y afecta directamente a la imagen, lo que provoca que los objetos no puedan ser identificados fácilmente.

El error presentado en la ubicación del brazo robótico utilizando una entrada manual es de aproximadamente 3 mm, pero este error puede aumentar aproximadamente a 4.5 milímetros cuando se hace uso del sistema de visión artificial, debido a factores como la luz ambiental.

Los movimientos del brazo robótico se limitan al plano XY, ya que con los dispositivos utilizados no es posible identificar la profundidad a la que se encuentran los objetos, se da por hecho que cada objeto se encuentra puesto sobre el espacio de trabajo.

Referencias

Barrientos, A., Peñin, L., Balaguer, C. y Aracil, R. (2007). *Fundamentos de robótica* (2.^a ed.). Mc Graw Hill.

BCN3D. (s.f.). BCN3D MOVEO – Un brazo robótico de código abierto impreso en 3D [Blog]. <https://www.bcn3d.com/es/bcn3d-moveo-un-brazo-robotico-de-codigo-abierto-impreso-en-3d/>

Culjak, I., Abram, D., Pribanic, T., Dzapo, H., & Cifrek, M. (2012). *A brief introduction to OpenCV*. IEEE.

Flexbot. (s.f.). ¿Cuáles son las aplicaciones de la robótica? [Blog]. <https://www.flexbot.es/aplicaciones-de-la-robotica/>

Ingenio Triana. (s.f.). *Brazo Robot* [Blog]. http://ingenio-triana.blogspot.com/p/blog-page_20.html

¿Qué es y para qué puede servir el 'machine learning'? (2017, 19 de octubre). *El País*. https://retina.elpais.com/retina/2017/10/19/innovacion/1508392516_816211.html

¿Qué es la robótica? (2020, 5 de diciembre). *Revista de Robots*. <https://revistaderobots.com/robots-y-robotica/que-es-la-robotica/>

RoboDK. (s.f.). Programación de Robots. <https://robodk.com/doc/es/Robot-Programs.html>

Unipython. (2018). CALIBRACIÓN DE LA CÁMARA OPENCV [Blog]. <https://unipython.com/calibracion-la-camara-opencv/>