

Tomo 06

Ingenierías

Prototipo de Brazo Robótico de Bajo Costo Operado mediante Interfaz Gráfica de Usuario

Juan Abisai Briano Rosales¹, Dr. Luis Carlos Méndez González²,
Ing. Fátima Sánchez Molina³, Dr. Isidro Jesús González Hernández⁴.

Resumen— Se diseñó un prototipo de brazo robótico capaz de realizar una manipulación de objetos moviéndolos de un punto a otro, primero detectando el color del objeto si se cumple el color, el prototipo toma la pieza y la deja en una posición mediante la programación de una placa Arduino. Se acciona mediante una interfaz gráfica de usuario utilizando el lenguaje de Python, se puede visualizar 2 cámaras en la interfaz conectadas al pc para monitorear el prototipo y evitar estar presente en el proceso. El objetivo es monitorear y controlar el brazo robótico mediante una interfaz gráfica de usuario, simulando la carga de materiales peligrosos en empresas. Se obtuvo el resultado esperado, debido a que toma 1 pieza y hace la rutina predeterminada, además se puede visualizar el prototipo mediante la colocación de cámaras físicamente y verlas en la interfaz y poder visualizarlas en tiempo de ejecución del código.

Palabras clave— Arduino, openCV, Interfaz, SolidWorks, Brazo robótico.

Introducción

La robótica es una herramienta vital en la tecnología de hoy en día, por su eficacia y precisión al realizar actividades complejas y su gran desempeño debido a que los robots pueden trabajar eficazmente durante varias horas o largos periodos y sin reducir la precisión de sus movimientos, cada vez es mayor el uso de la robótica en distintas ramas y empresas en general (Cárdenas, 2015).

Actualmente, muchas empresas se dedican a cargar material mediante personas o trabajadores, como contenedores con desechos tóxicos, químicos y/o frágiles. Lo que conlleva accidentes producidos por el manejo inadecuado de las personas y como consecuencia conlleva accidentes en todo el mundo. Ya que las personas no pueden hacer los mismos movimientos con la misma precisión y velocidad por lo que ponen en riesgo su salud al sufrir un accidente cuando se traslada algún material o cargarlo a una plataforma o camión. Además, dependiendo la sustancia que lleva el contenedor o el recipiente donde se lleva el material puede ocasionar accidentes extremos a la persona, ya que, en casos extremos surgen accidentes fatales perdiendo la vida de la persona se necesita evitar estos accidentes o reducirlos. Además, buscar la manera de que el proceso sea automatizado y se pueda monitorear. Para estar al pendiente, si ocurre un inconveniente y manipularlo a una distancia sin riesgo de algún accidente hacia la persona.

Existe una diversidad de prototipos de brazos robóticos los cuales realizan alguna secuencia predeterminada pero solo tienen esa secuencia y no cuentan con más funciones, por lo que es importante incorporar una interfaz gráfica de usuario para poder ejecutar diferentes comandos a los brazos robóticos y sea más práctico y accesible que cualquier persona pueda manipular y entienda fácilmente el funcionamiento de la interfaz (Milanés, 2016). Además de incorporar la visualización de cámaras en distintas áreas enfocadas al brazo robótico para ver los movimientos del brazo robótico y evitar accidentes tomando las distancias adecuadas al trabajar con el brazo robótico. Así, no se pone en riesgo la salud de quien manipula el brazo robótico.

En esta investigación se propone un prototipo de brazo robótico que detecte objetos mediante un sensor de color, mediante una interfaz gráfica de usuario, y si se detecta, realice una secuencia predeterminada moviendo el objeto de un punto a otro (Pérez, 2014), además incorporar la visualización de 2 cámaras agregadas a la interfaz para manipularlo desde una distancia adecuada (Mijares, 2019).

¹ Juan Abisai Briano Rosales alumno de la carrera de Ingeniería en Mecatrónica de la Universidad Autónoma de Ciudad Juárez.
al167992@alumnos.uacj.mx

² Dr. Luis Carlos Méndez González es profesor investigador del departamento de Ingeniería Industrial y Manufactura en la Universidad Autónoma de Ciudad Juárez. luis.mendez@uacj.mx

³ Ing. Fátima Sánchez Molina es profesora del Colegio Nacional de Educación Profesional Técnica. Plantel Ciudad Juárez 2.
fatima.sanchez@chih.conalep.edu.mx

⁴ Dr. Isidro Jesús González Hernández es profesor Investigador del departamento de Ingeniería Industrial en la Universidad Autónoma del Estado de Hidalgo. igonzalez@uah.edu.mx

Metodología

A. Diseño del prototipo

El primer paso fue realizar el diseño en 3D del prototipo y así tener una herramienta visual de como sería el resultado final, además de tenerlo a la disposición y poder editarlo si se requiere alguna modificación, el diseño de este proyecto se hizo en SOLIDWORKS, el cual es un software para el desarrollo de piezas y ensambles en 3D. Además de que permite el dibujo de los planos del prototipo por si se usa para la posteridad. Ver figura 1.



Figura 1. Render del proyecto realizado en SolidWorks.

B. Construcción del prototipo físicamente

Para este proyecto se usó un kit de brazo robótico adquirido en el mercado con un precio accesible (entre 1500-2000 pesos mexicanos), este kit está disponible en diferentes plataformas a la venta por lo que es muy accesible adquirirlo este proyecto cuenta con 5 servomotores Tower Pro MG995 de 180 grados, la base del servomotor, la rueda para acoplar el engrane del servomotor y el brazo donde se conecta el servomotor para realizar el movimiento y un gripper para tomar objetos, además se diseñó una pieza tipo prisma rectangular para tener más alcance en el brazo robótico, se agregó una base tipo prisma cilíndrico para montar el brazo robótico y por último se usó una base de Delrin para montar el proyecto. Ver figura 2.

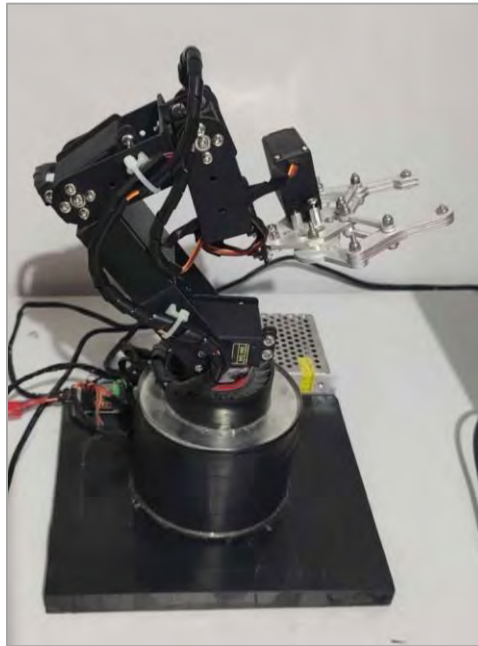


Figura 2. Proyecto armado físicamente.

C. Programación del prototipo

Para este proyecto se usó la programación en C++, mediante Arduino que es una plataforma de programación gratuita (se usó un Arduino Mega 2560) para controlar el brazo robótico (Ordoñez, 2024). además, se usó un módulo PCA9685, que es un dispositivo que controla 16 canales o salidas PWM (en este proyecto controla los servomotores) por medio de comunicación I2C. En este proyecto se utilizó para controlar 5 servomotores Tower Pro MG995 de 180 grados, también se usó un sensor de color Tcs34725, para detectar el color del objeto. Para trabajar con el módulo PCA9685 y el sensor de color Tcs34735, se instalaron las librerías disponibles y gratuitas en Arduino, se usó una fuente de voltaje de 5 Volts y 10 Amperes. Se realizó un código para controlar el movimiento de cada uno de los servomotores asignando el valor de desplazamiento de los servomotores, por ejemplo; servomotor 1 con movimiento de 0-180 grados mediante el serial monitor y así con el resto de los servomotores, para poder moverlos a diferentes puntos, por ejemplo; A30 el primer servo se mueve a 30 grados o B40, C50, D70, además se agregaron otros comandos como; G para grabar posiciones, P para reproducir posiciones, R para resetear posiciones y 1 para realizar la secuencia predeterminada, con la condición de detectar el color verde para poder ejecutar la secuencia mediante el serial monitor.

C. Diseño de interfaz

Para realizar la interfaz gráfica se usó la programación en código abierto de Python, que es un lenguaje de programación principalmente orientado a objetos, además de tener diferentes módulos para crear interfaces como Qt designer o Tkinter. En este proyecto se usó el módulo de Qt designer para el desarrollo de la interfaz gráfica de usuario (GUI) Se diseñó la ventana principal con un botón donde se encuentran las instrucciones del funcionamiento del prototipo del brazo robótico, inicialmente se deshabilitan los botones para mover los servomotores, grabar posiciones, resetear posiciones, reproducir posiciones, generar un archivo CSV y mandar un dato por el puerto serial que se mandará a Arduino, también el cuadro de texto donde se recibe la información y muestra las acciones realizadas en el sistema deshabilitadas. Finalmente, tenemos un botón llamado LOG IN que permite habilitar los botones deshabilitados cuando se inicia sesión. Ver figura 3.

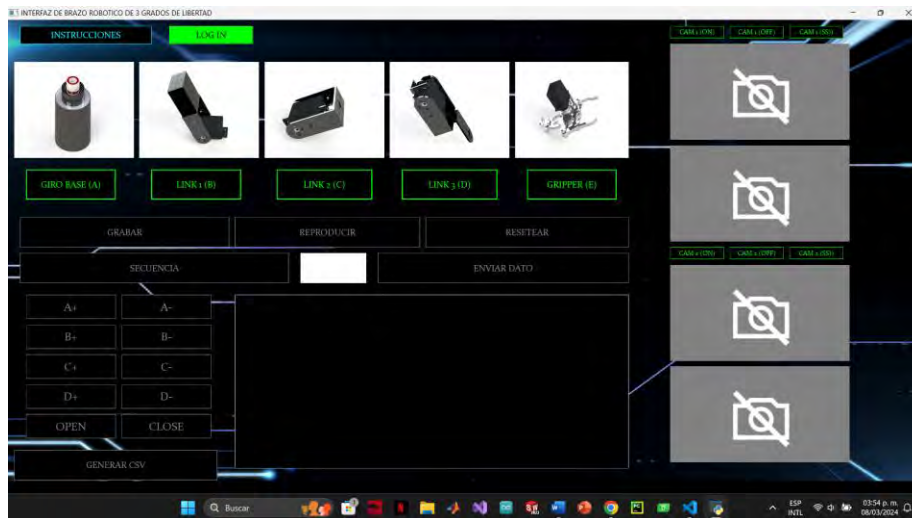


Figura 3. Imagen inicial de interfaz.

Para poder acceder a la configuración del sistema es necesario ingresar el usuario y contraseña en log in, para habilitar todos los botones accediendo desde el botón de LOG IN. Ver figura 4.

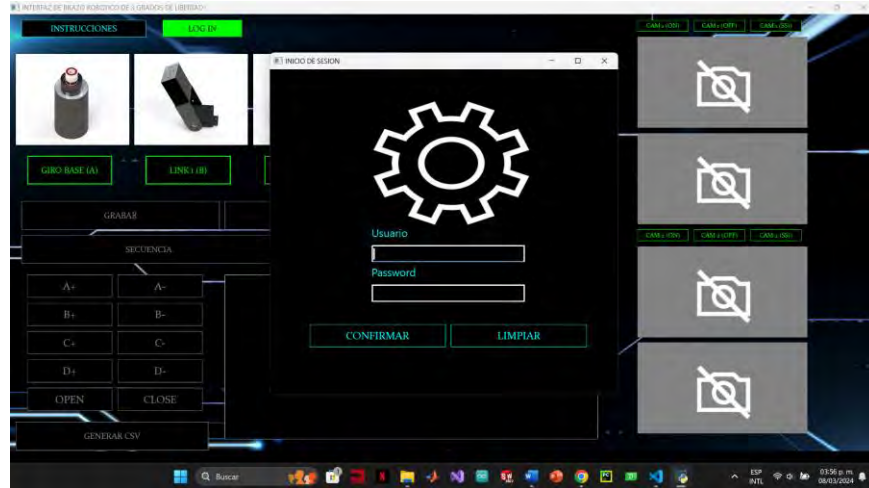


Figura 4. Ventana de log in.

Una vez que se ingresa el usuario y el password correctos se habilitan todos los botones deshabilitados y te permite mover cada uno de los servomotores y ejecutar diferentes comandos, una vez iniciada la sesión los colores de los botones cambian indicando que están habilitados. Ver figura 5.

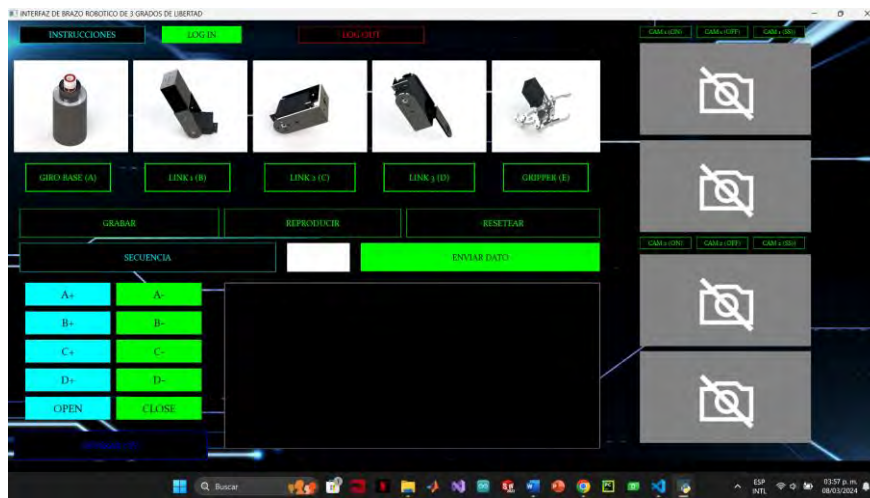


Figura 5. Interfaz con todos los botones habilitados después de iniciar sesión.

Podemos visualizar 2 cámaras conectadas al pc sin iniciar sesión e iniciando sesión y verlas mediante la interfaz presionando CAM 1 (ON) y CAM 2 (ON) y presionando los botones de CAM 1 (OFF) y CAM 2 (OFF) desactivar las cámaras y con los botones de CAM 1 (SS) y CAM 2 (SS), permiten tomar captura de las cámaras y generar una carpeta llamada “captures” en la cual crea 2 carpetas más llamadas “cam1” y “cam2” en las cuales se guardaran las capturas de la cámara 1 y de la cámara 2, además en la carpeta de captures se guardara el archivo CSV (en Excel) de los datos mostrados en el cuadro de texto de la interfaz. Ver figura 6.

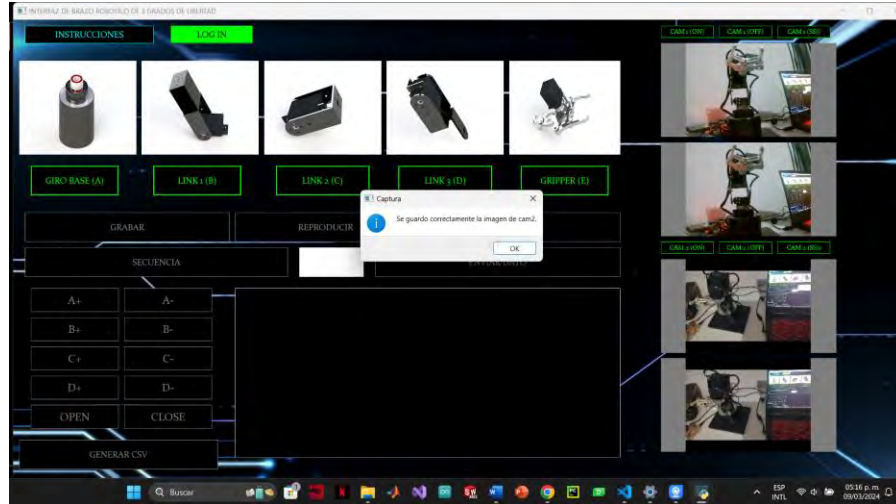


Figura 6. Captura y video de cámaras conectadas a interfaz sin iniciar sesión.

Las capturas de imágenes de ambas cámaras se guardan con una extensión .png además en la misma carpeta de capturas podemos visualizar el archivo generado por el botón de Generar CSV Ver figura 7.

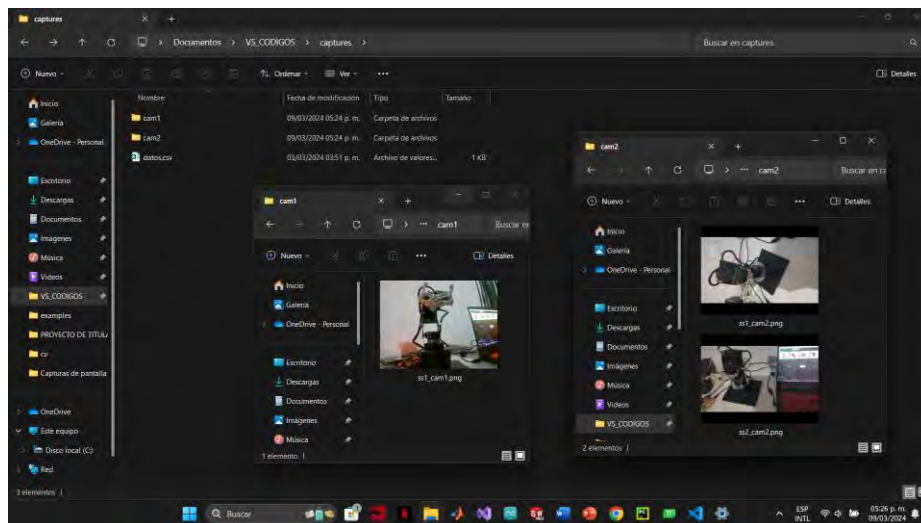


Figura 7. Carpetas generadas a partir de la captura de cámaras de la interfaz.

Resultados

Se observaron resultados correctos, debido a que este proyecto realiza exitosamente las acciones que están programadas con los botones digitales de la interfaz gráfica. Al presionar el botón de secuencia en distintas ocasiones (alrededor de 20 pruebas) el sistema funcionó correctamente indicando la detección de escala de color RGB del sensor del color, si se encontraba en el rango de color verde, muestra el mensaje que se ha detectado el color verde e inicia a realizar la secuencia predeterminada, llevando una pieza u objeto de un punto a otro, el robot no presentó pérdida de potencia en sus movimientos, tampoco movimientos que no estuvieran programados en el código, por lo que la secuencia hecha por el brazo robótico es correcta.

Además, las cámaras conectadas a la interfaz funcionan correctamente visualizando el video en la interfaz, con un pequeño retardo de milisegundos, además guardando correctamente las capturas de pantalla de las cámaras en la carpeta creada en el código de Python llamada "captures" y dentro de ella creando otras dos carpetas llamadas "cam1" y "cam2", correspondientes a las 2 cámaras conectadas al pc y dentro de las 2 carpetas anteriores mostrando todas las capturas realizadas en la interfaz gráfica de usuario. También se genera un Excel al presionar el botón de la interfaz llamado Generar CSV, que guarda los datos del cuadro de texto de la interfaz donde se muestran mensajes de información cuando se manda un comando o un movimiento al brazo robótico. Ver figura 6 y 7.

Conclusiones

Este proyecto funciona adecuadamente como se planteó desde un inicio por lo que se considera como acertado, mostrando la visualización de las cámaras en la interfaz, viendo en tiempo casi real con milisegundos de retardo, al igual que los movimiento de los servomotores teniendo un pequeño retardo de milisegundos pero funcionando correctamente, este proyecto sirve de base para trabajar en futuras investigaciones relacionadas a brazos robótico, por lo que se sugiere buscar una manera de incorporar una manera de conectarse por Bluetooth al brazo robótico o por medio de Wifi para hacer el alcance más largo de la interfaz del prototipo, debido a que este proyecto debe de tener conectado las cámaras a la pc, por lo que la distancia del cable indica la distancia del prototipo a las cámaras.

También se sugiere usar una expansión a la placa Arduino para poder grabar más posiciones, ya que en este proyecto las posiciones que se pueden grabar son de 19 como máximo. Otra limitante es el uso de fuente de voltaje, ya que, debe estar conectada la fuente de voltaje a un tomacorriente por lo que debe de estar fijo el brazo robótico, como alternativa se sugiere el uso de baterías recargables y sería una manera más practica de usar el prototipo en diferentes partes.

Referencias

- Cárdenas, M. M., Barrios, P. P., Moreno, K. M. G., Arismendy, J. F. S., & Ávila, M. C. O. (2015). Diseño y Construcción del Prototipo de un Brazo Robótico con Tres Grados de Libertad, como Objeto de Estudio. *Ingeniare*, (18), 87-94.
- Mijares, J. S., Fraire, L. J. F., De La Torre, R. E. C., & Juárez, A. R. (2019). CONTROL DE BRAZO ROBÓTICO DE TRES GRADOS DE LIBERTAD CON DE PROCESAMIENTO DIGITAL DE IMÁGENES PARA SEGUIMIENTO DE UN BRAZO HUMANO (THREE DEGREE OF FREEDOM ROBOTIC ARM CONTROL WITH DIGITAL IMAGE PROCESSING FOR FOLLOW-UP A HUMAN ARM). *Pistas Educativas*, 41(134).
- Milanes Hermosilla, D., & Castilla Pérez, A. (2016). Generación de trayectorias para el brazo robótico (ArmX). *Ingeniería Electrónica, Automática y Comunicaciones*, 37(3), 58-71.
- Ordoñez, R. C. O. (2024). Brazo Robótico de Tres Articulaciones Mediante Servomotores con Tecnología Arduino. *Ciencia Latina Revista Científica Multidisciplinar*, 8(1), 3099-3113.
- Pérez, J. Z. (2014). Diseño e implementación de una máquina automática clasificadora de objetos según su color detectados mediante un sensor de color y clasificados por un brazo robótico. Universidad de las Fuerzas Armadas, Ecuador.