Sawyer Rubik's Solver

CSE Senior Design

Chase, E., Medina, J., Gregory, B., Lopez, J., Nguyen, C.

Executive Summary

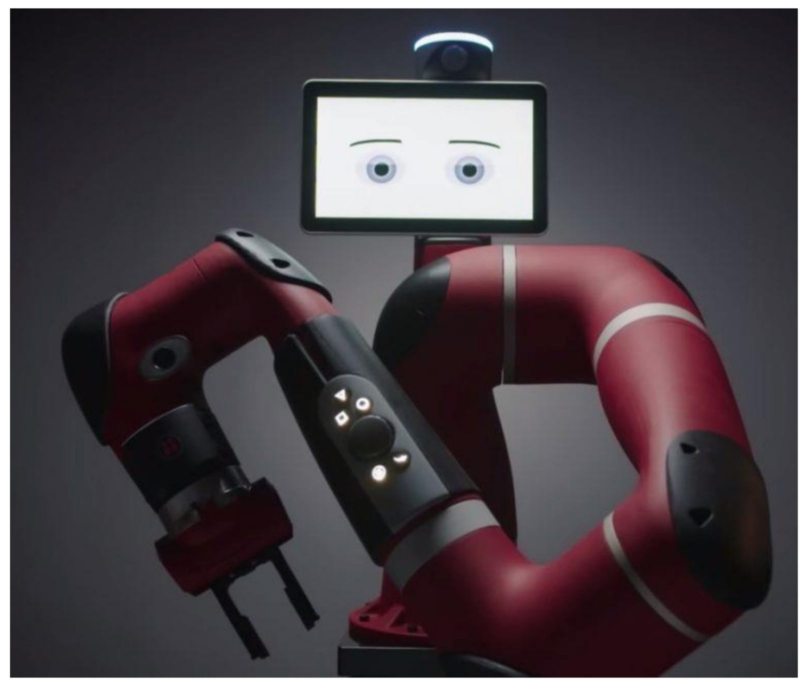
The project aims to facilitate outreach presentations by developing a robotic arm system that can solve a Rubik's Cube. The Sawyer Robotic Arm is programmed using ROS and InteraSDK, while Python and OpenCV are used to analyze a video feed of the cube. A mount is also incorporated to securely hold the cube in place while the robotic arm manipulates it.

Background

- Our client, Dr. McMurrough, has commissioned this project with the objective of facilitating outreach presentations. The primary aim is to provide a portable setup that can be taken to schools to conduct engineering demonstrations for elementary and middle school students. By conducting these demonstrations for elementary and middle school students, this project will offer them an opportunity to gain exposure to the field of engineering. This may pique their interest in the subject and encourage them to consider pursuing it as a future career option. Additionally, the project may help in enhancing their understanding of engineering concepts, as well as provide them with hands-on experience, which can aid in their overall academic and personal growth.

Robot

The Sawyer Robotic Arm is programmed using ROS and InteraSDK. The robot must be running in SDK mode. We were able to log the angles of each joint and use them to create poses which the robot will go through to solve it. The robot normally operates at 30% speed however this was lowered due to this particular robot not having great precision.



Computer Vision

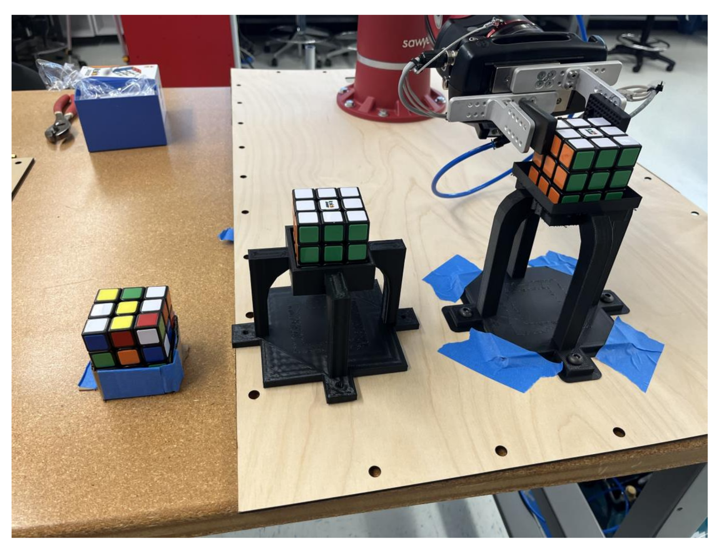
To capture the Rubik's cube state, we used Python and OpenCV to analyze a video feed of the cube. We started by converting each frame of the video into the HSV color space, as this provided a more robust method for color detection compared to RGB.

To isolate each of the nine tiles on each face of the cube, we used OpenCV to draw a template of circles on each face of the cube. This template consisted of nine circles in a 3x3 grid pattern, with each circle encompassing a single tile.

To identify the color of each tile, we compared the hue value of the key pixel location. We also used a range of hue values to account for slight variations in lighting conditions. Once we had identified the color of each tile on each face of the cube, we translated the colors into directions and passed the cube state to the Kociemba algorithm to generate a solution.

Mount

The mount would need to securely hold the Rubik's Cube in place, while allowing the robot arm to manipulate it. This could be done by using the robotic gripper that is capable of manipulating the cube in three-dimensional space. Constructing multiple mount protypes gave us a better idea of what was needed for the gripper to correctly read the cubes sides. The finalized mount was made taller than the others, added a funnel up top for the cube to slide in easier, and screw holes were made bigger to properly fit the platform.



Program Flow

1. Take pictures of all sides of the Rubik's Cube
2. Use OpenCV to convert the pictures of the cube into strings
3. Send the Rubik's Cube state string through Kociemba to get solve string
4. Interpret the solve string output and call necessary arm positions to solve the cube

Conclusions

Due to unfortunate circumstances, we were only able to work on this project for one semester. Given more time we would be able to improve the product to make it more efficient. This would involve reducing the amount of robot movements needed to solve the cube and keeping track of the orientation of the cube. We would also change the mount shape to have access to more sides of the cube at once.

By using the Sawyer Robotic Arm to solve the Rubik's cube, it creates a fun and engaging way to teach children about technology and problem-solving skills. By introducing children to the concept of robotics and automation, we can help them develop an interest in STEM fields and potentially pursue careers in these areas in the future.

We'd like to thank Chris McMurrough for continuous help throughout the project and guidance when we've been lost.
We'd also like to thank Steve Lopez for his assistance with the design and creation of the platform that the robotic arm rests on.
Without their help, this would've been a much more difficult experience.

Libraries

- Open CV
- Rospy
- Kociemba