Solving Rubik's Cube
- With Denso and DC Motor

OVERVIEW

Motivation

1.Flexibility in Functionality:

Blind robots need reprogramming for application with different location, size and funtion

2.Collaborative Work

vision system is typically necessary for safety reasons

Application

- Pick and Place – (Locate, Read, Guide)
- Assembly – (Locate, Read, Guide, Check)
- Quality Inspection - (Check, Measure, Read)
- Packaging and Palletizing - (Locate, Read, Guide, Check)
- Lab Analysis and Testing - (Locate, Measure, Read, Guide, Check)
- Screw Driving - (Locate, Guide, Check)
- Labeling - (Locate, Read/Verify)
- Gluing, Dispensing and Welding - (Guide, Check)
- Polishing - (Check)
- Injection Molding - (Check, Measure)

.....

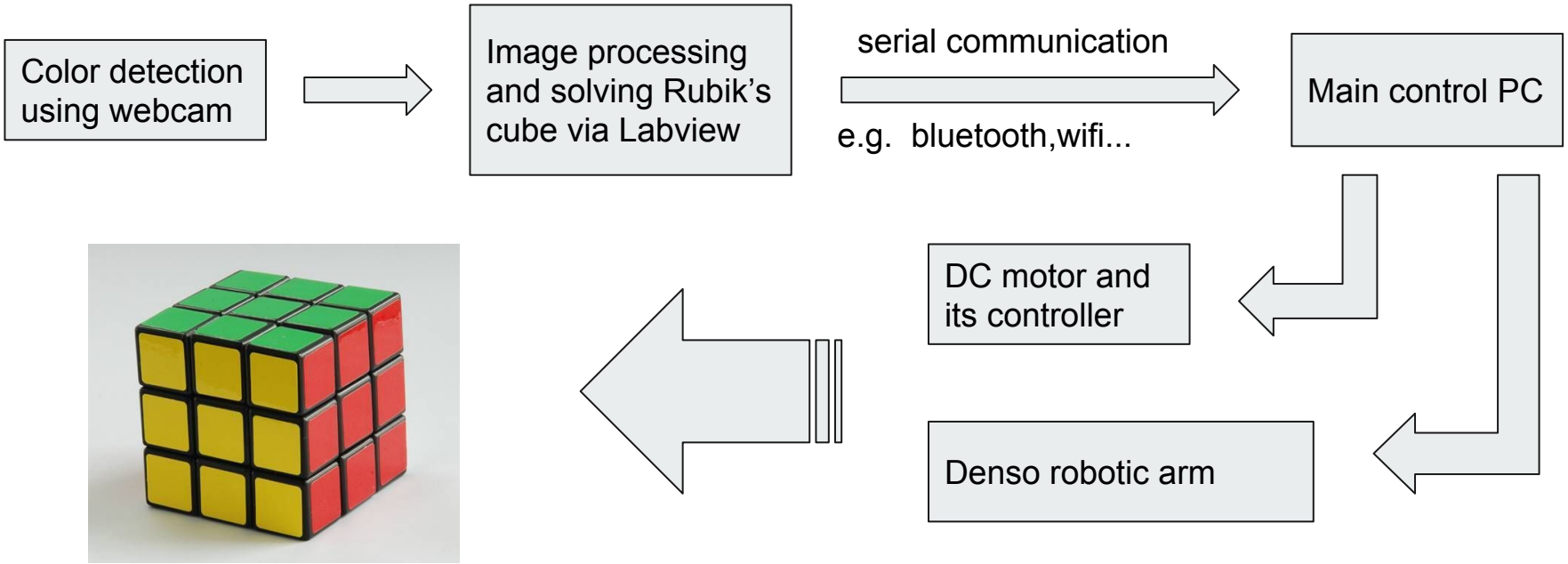
Purpose

A web camera acts as a set of eyes that detect color and position of a randomly shuffled Rubik's cube. Corresponding solve method and related motion commands are processed and sent to denso robotic arm.

Objective

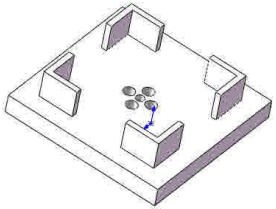
- Color detection which is immune of position and intensity of light and environment error.
- Trajectory following and pose maintenance of Denso robotic arm.
- Real-time Gripper’s force control to grasp Rubik’s Cube.
- Precise rotation of Rubik’s Cube mounted on DC motor

System Review



Hardware

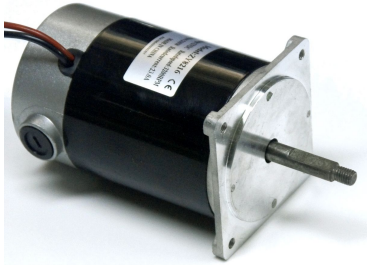
3D Printed Mount



Logitech HD Webcam C310

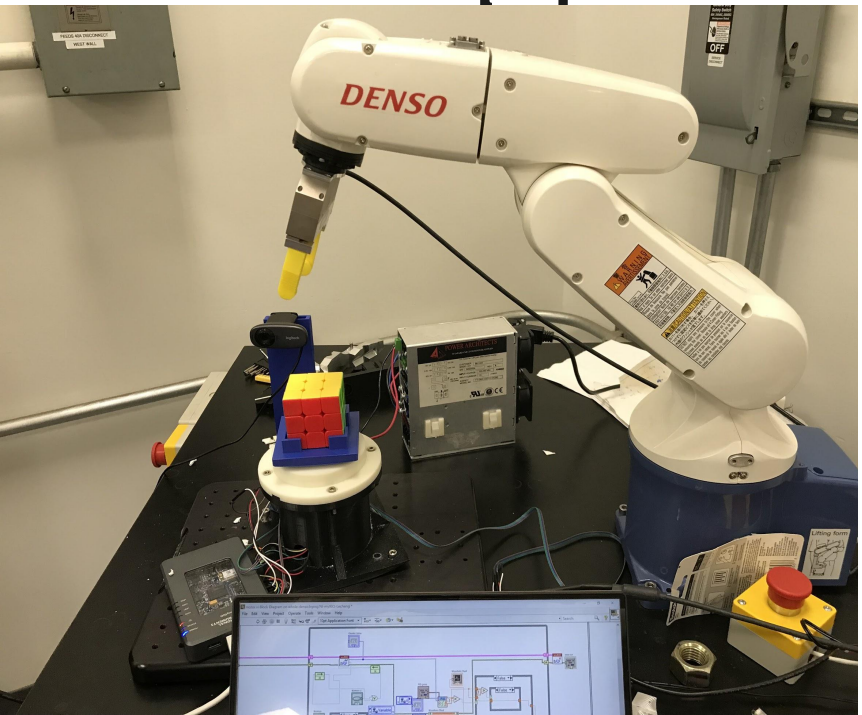
MyRio

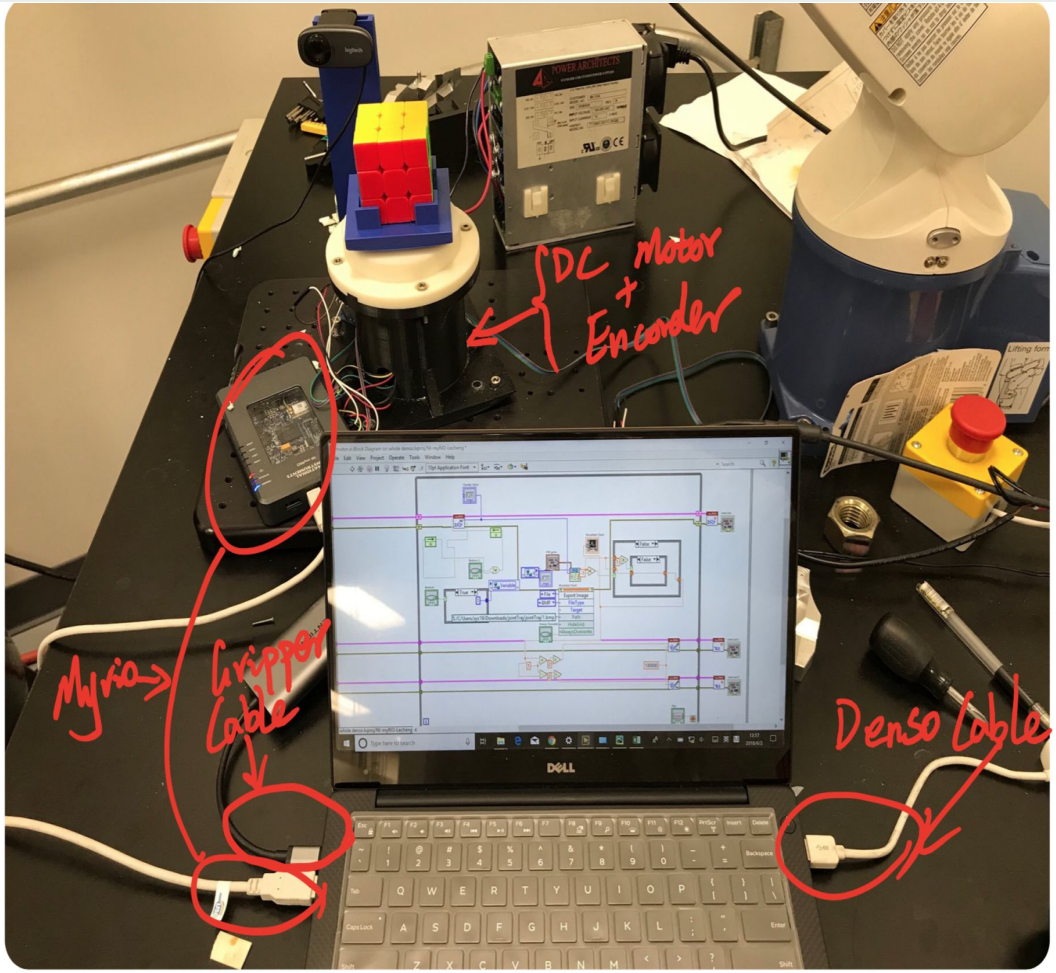
Denso Robotic Arm



DC motor

Whole equipment





METHODS

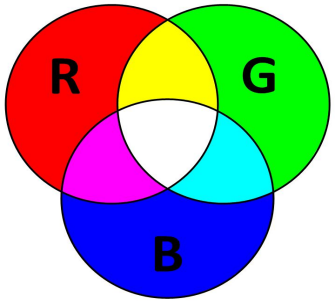
How does it all work?

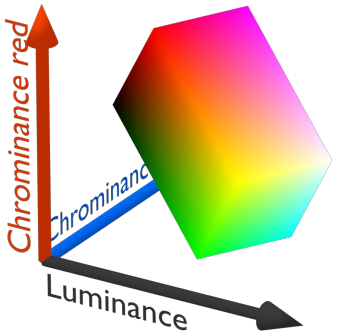
Color detection & Noise immunity

1. Transfer RGB to YCrCb for better color distinction at low resolution

2. Apply median filter to remove salt and pepper noise which is generated by image sensor & transmission channel

3. Assign color to sample areas of 9 grids in each surface of Rubik's Cube basing on different threshold value (brightness Y and color difference CrCb)





Control Methods

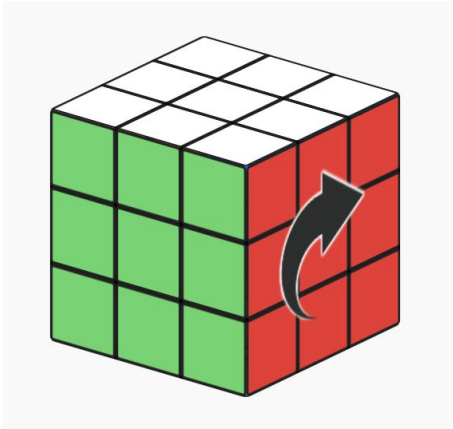
- Trajectory following and pose maintenance of the center the Rubik's cube by position control of Denso robotic arm using inverse dynamics.
  - Use color detection program to get the solution
  - Put the face to be rotated in the bottom for each step
- Use DC motor to control the mount of Rubik's cube to conduct rotation of each step of solution.
  - Based on PID controller to eliminate the steady state error
- Use Gripper's force sensor to control the force when grasping object

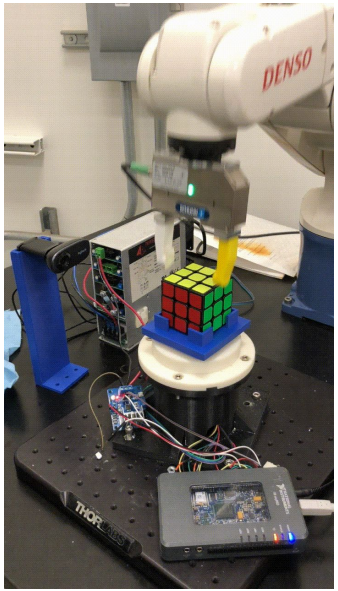
Kinematics of Denso Robot Arm

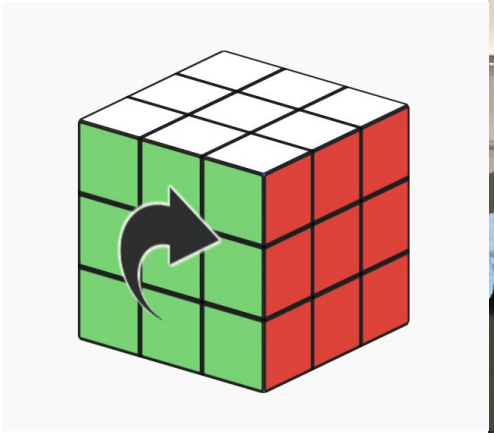
DH Parameters :

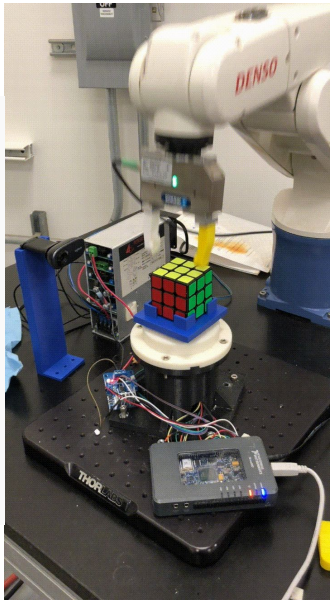
i	alpha	a	d	theta
1	0	0	280	θ1
2	-90°	0	0	θ2
3	0	-210	0	θ3
4	90°	-75	210	θ4
5	-90°	0	0	θ5
6	90°	0	70	θ6

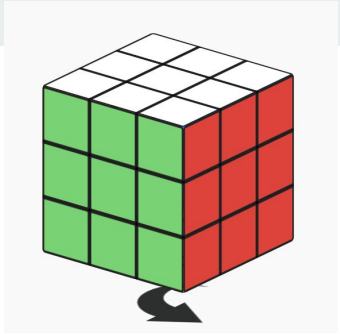
Robotic Arm Trajectories for Each Rotation

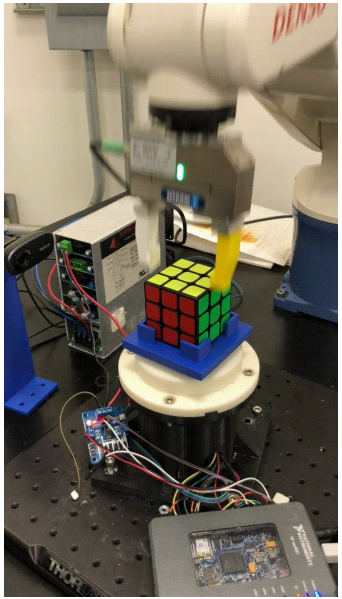


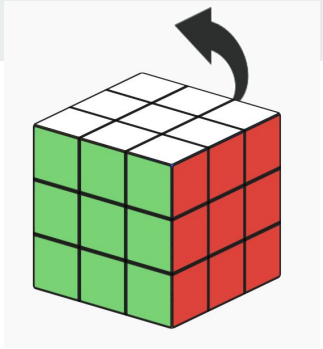


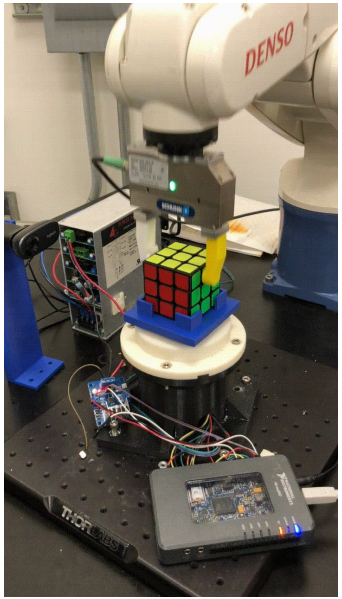


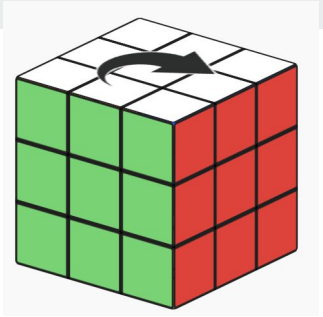


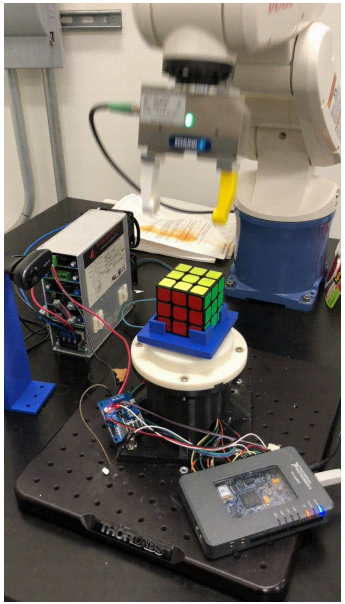






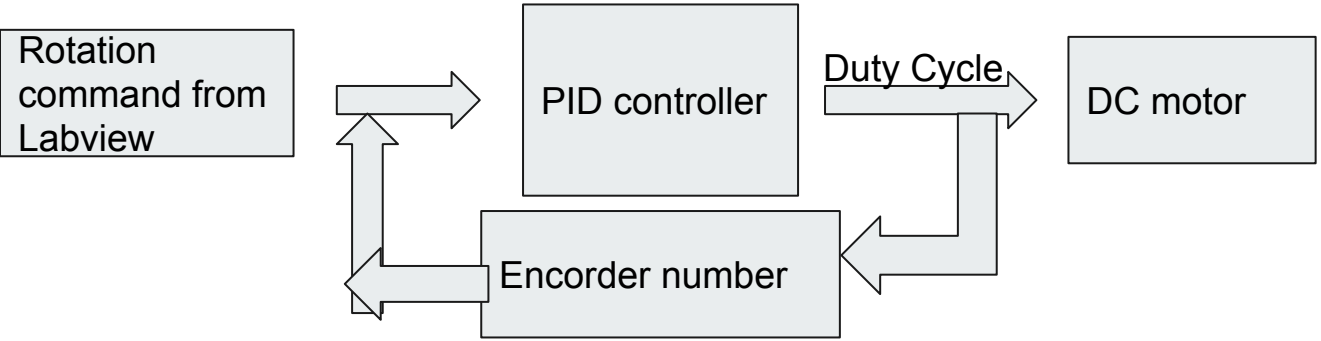




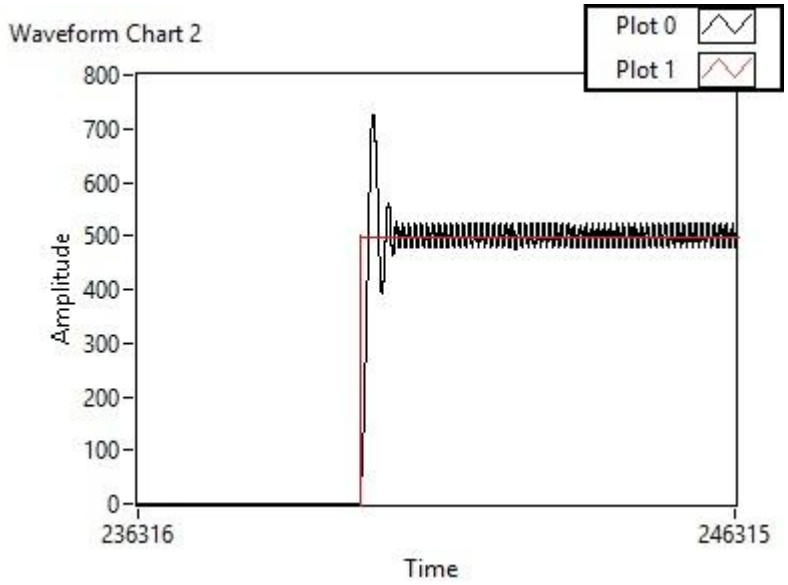


Controller

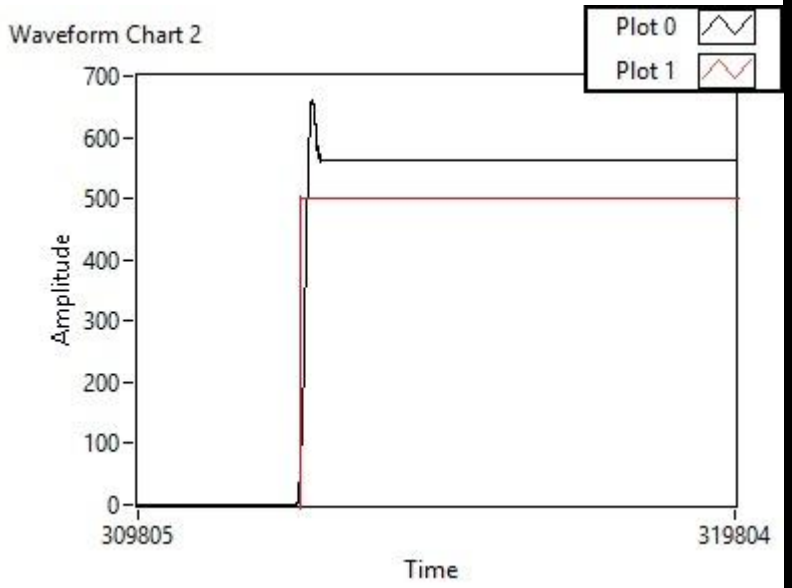
Use PID controller to control the mount for rotating to certain degree

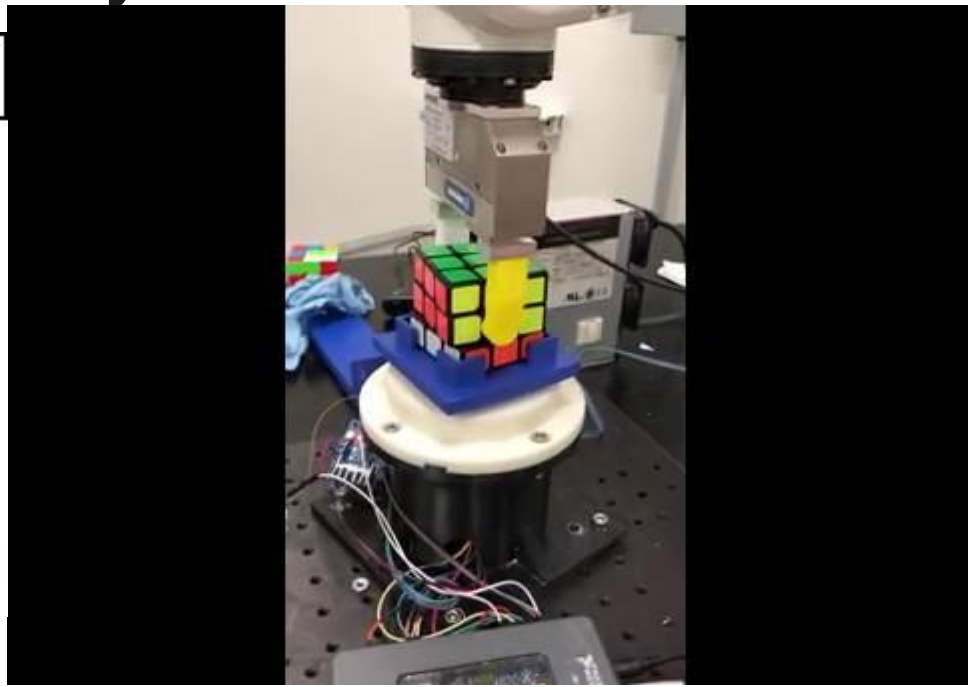


Tune PID to reduce steady error : large P vibrate

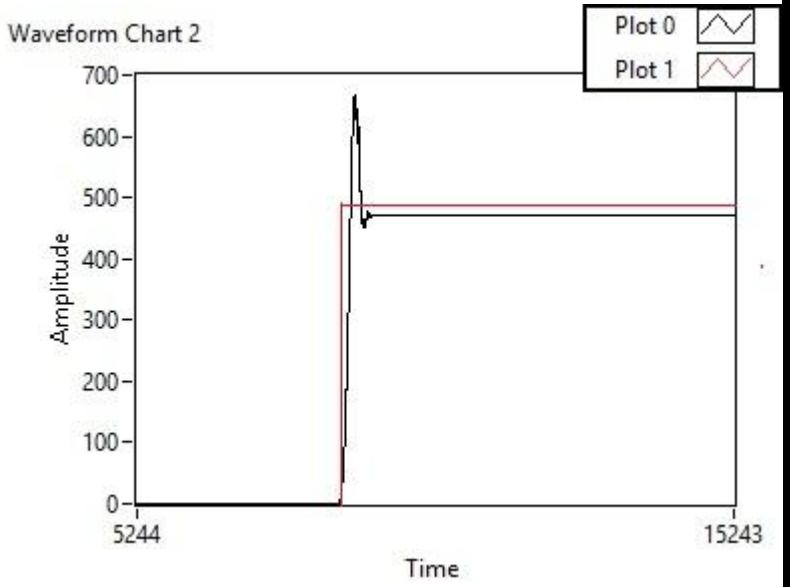


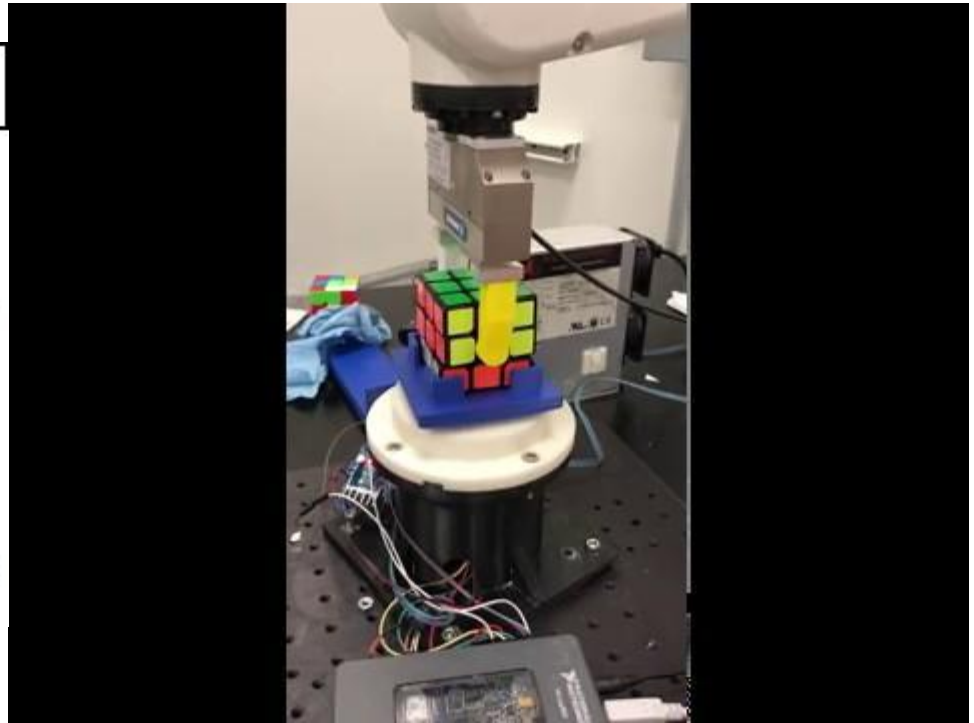
Tune PID to reduce steady error: small P



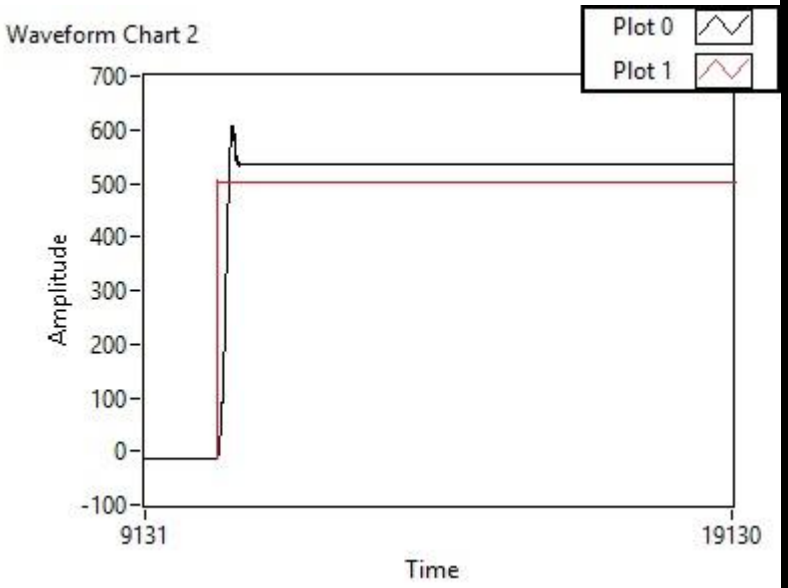


Tune PID to reduce steady error : good P

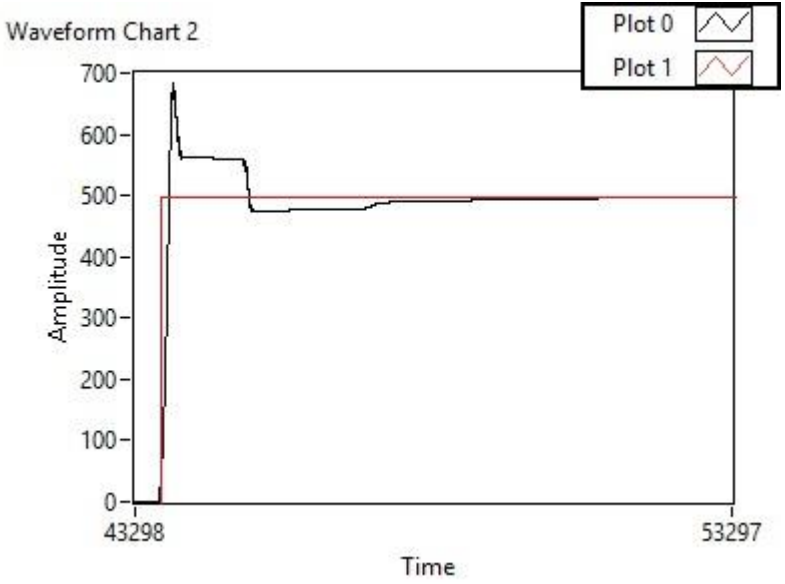


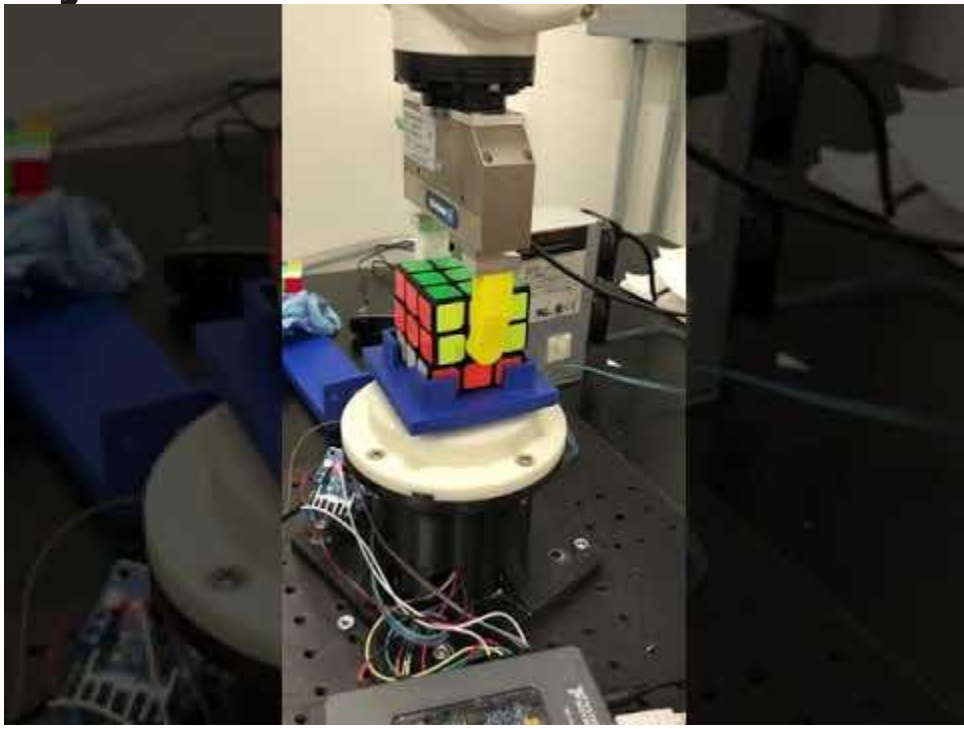


Tune PID to reduce steady error : PD



Tune PID to reduce steady error : With PID





Acheive real-time control

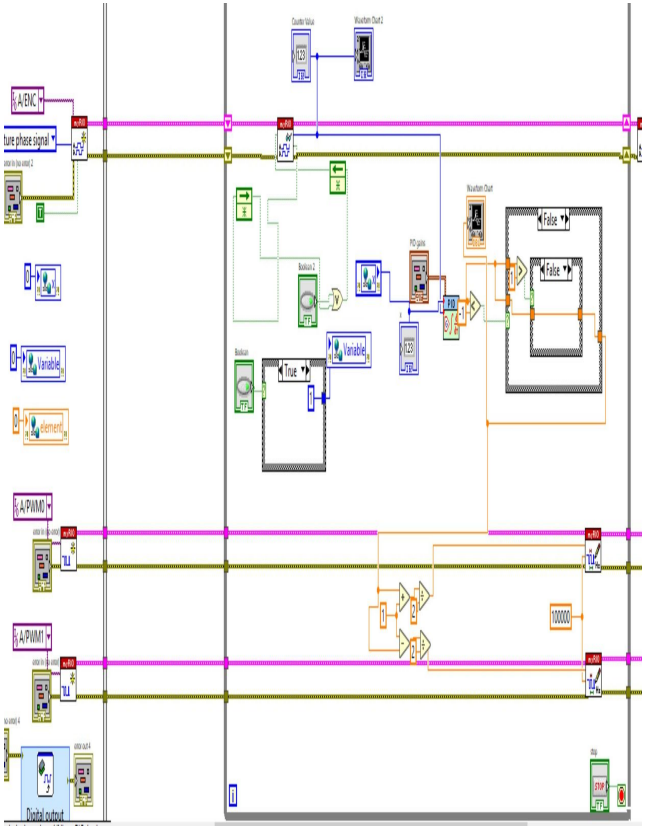
Launch myrio to control Motor at 0 position

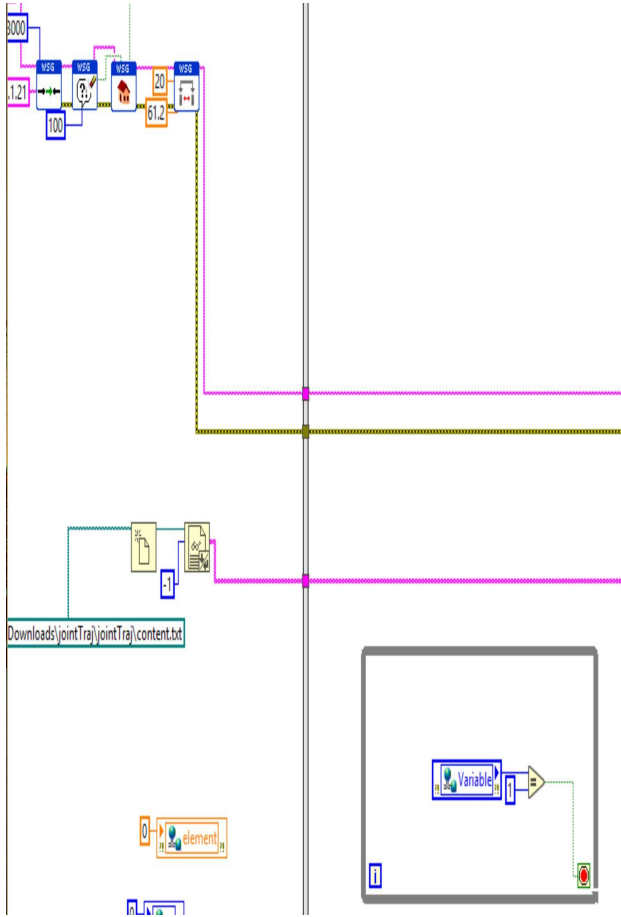
Launch denso, gripper and camera to attain solution

Use Matlab(inverse kinematics) to transform and integrate solution

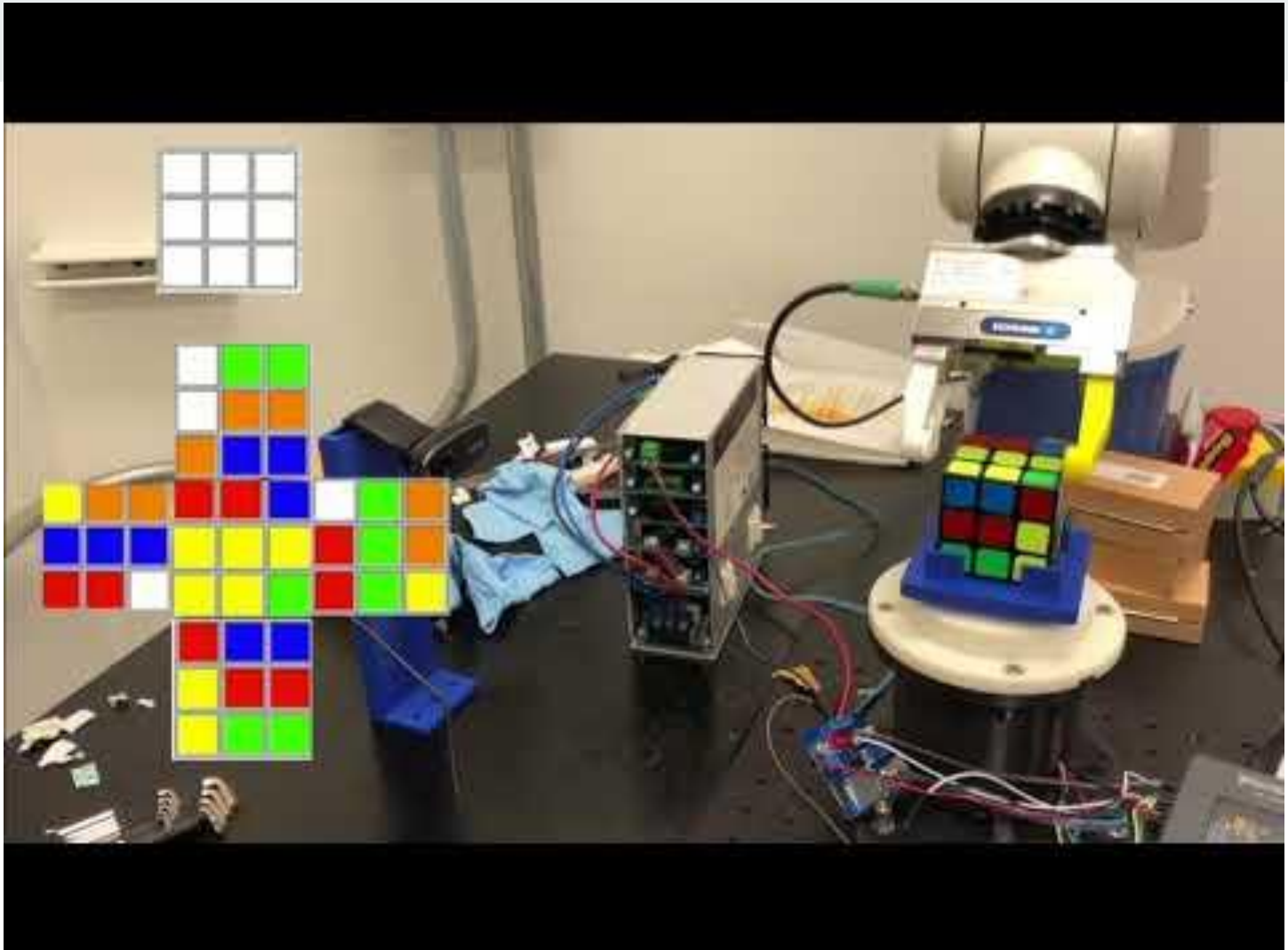
data

Push start button to run whole manipulate process





Simple Video Demonstration



Future works

- Improve the accuracy of hardware to ensure more accurate position control.
  - especially the gripper and the mount.
- Design a model-based controller so that we could do logical analysis of the whole system.
  - such as state-feedback observer control
- Use two denso robotic arm instead of motor to complete the cube solving task.
- Add computer vision as feedback into the controller design.
  - use the vision part to check whether each step is perfectly complete.
    - such as position detection of edges.

Thank You!

Q&A