Sensors and Motor Control Lab

Individual Lab Report – 3

Nithin Subbiah Meganathan

Team D: CuBi

Team Members

Jorge Anton Garcia

Paulo Camasmie

Laavanye Bahl

Changsheng Shen

Carnegie Mellon University

7th March 2019

Individual Progress

My main task was to achieve the control of Dynamixel motors. During the last demonstration, we controlled a Dynamixel AX-12A through Dynamixel API on Windows. This time we were supposed to achieve control of all the Dynamixel motors using Robot Operating System (ROS). We used two Dynamixel MX-106 motors for the arm and two Dynamixel AX-12A for the gripper. The other required components were USB2Dynamixel connector which connects the motors to the computer and an ArbotiX controller to power the motor. Control and power for all the motors can be achieved by connecting them in serial fashion.

First, I tried to control the motors on Linux using Dynamixel SDK. But it did not work. The ID's and the baud rate of the motors were unknown. We got the motors from the inventory and thus it had old firmware on it. So, switched to Windows in order to utilize the Dynamixel API to flash the firmware and set the ID and baud rate to desired values.

But even then, all the motors were not detected on the Dynamixel API. Only one MX-106 was detected which had a baud rate ID of 34. The API did show something being detected on baud rate ID 1 (1000000 bps) but still the device was hidden. The firmware needed to be flashed, and the properties changed for the motors to be used but the problem was the motors not even found by the software.

After multiple attempts, it was found that the order in which the motors and circuit connections were made, mattered until the motors were flashed and properties known. Till then, the USB2Dynamixel was connected to the controller, and from the controller to the motors. The motors were being detected when the order of the connections was made as follows: USB2Dynamixel to a motor, motor series connection to other motors, last motor connected to the controller (for power).

Now as all the motors were detected, ID's were assigned sequentially, and baud rate was uniformly set to ID 34 for all motors. Switched to Ubuntu to try to control through the ROS Dynamixel package. The motors were detected and executed actions according to the commands give. But when a program called Mixcell, a Dynamixel API on Ubuntu, was run the motors crashed. The computers were not even able to detect them let alone control. Even the Windows Dynamixel API couldn't recognize any of the motors. So, for this demonstration we were not able to control the motors. The motors placement in the manipulator and their connections are shown in the figure

I had also created short-term scheduling and tasks to be met by the team.

Challenges

Dynamixel motors seem to be unreliable and very problematic at least in our case. I suspect that we encounter these problems because we are not using new motors but still, we should be able to flash the motors and start using them. We did not consider controlling motors as a high-risk task, but it proved to be so. This poses a huge challenge of predicting correctly which systems pose high-risk.

Teamwork

The task of controlling the motors was undertaken by Jorge, Bobby, and me. All three of us worked towards the same objective, helping each other, and made progress. Paulo had designed the complete manipulator and had made grooves for wiring of the motors. This was made possible by communicating about the design to accommodate the wires, prior to printing. Paulo had also made calculations on payload and the torque produced by attaching the manipulator to the robot base. Laavanye worked on the vision side, obtaining real-time point cloud data of the scene and removing the outliers (floor). Objects are also segmented in this process by the point clouds populating objects in each frame distinctively.

Figures

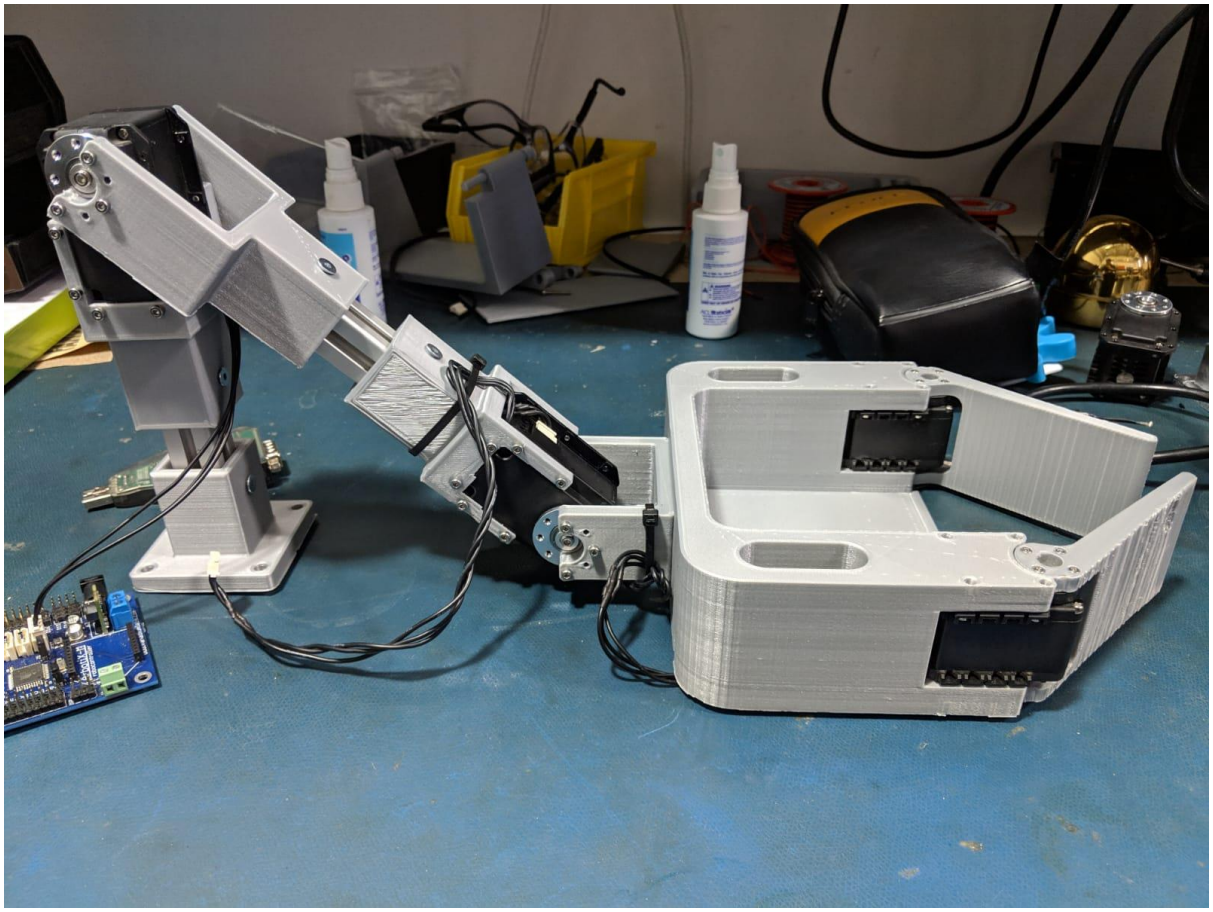

Figure 1. Manipulator with motors assembled and connected in series

Plans

The immediate plan is to achieve reliable control of the Dynamixel motors. I need to contact Dynamixel support centre to get help in this regard. If it is not possible, then an alternative must be found. After this is achieved, a mount for the camera must be designed and fabricated. All the sensors and manipulators should then be assembled on the robot. This will be teamwork, and this will be a milestone for our project. I will then be involved on the vision and learning aspects for the robot.