Jorge Anton Garcia
Team D – CuBi
Team mates: Laavanye Bahl, Paulo Camasmie, Changsheng Shen (Bobby), Nithin Subbiah Meganathan

ILR05
April 11, 2019

Individual progress

For CuBi, the biggest things I focused on was controlling manipulator and performing the local planning.

Dynamixel

For our previous lab demonstration, we assembled Cubi but we had issues reaching some configurations. Whenever we sent some joint angles, the dynamixels would rotate forward and then rotate backward because we were trying to send commands that were outside of the angle limits (set to one rotation). Due to the assembly of the motors, the wrist joint in a straight position was at an angle of 1.75π, but when we wanted to lower the wrist to a position of 2.25π it would exceed the 2π angle limit. To fix this, we used Mixcell software to configure the motors to allow for multiple rotations. For the dynamixels that do not allow for this configuration (finger joints), I had to reassemble them, so that the desired rotations were between 0 and 2π.

We also faced problems when closing the fingers because they are both longer than half of the tray width. If both are closed at once, they will crash into each other. We created a policy where the right finger would always be on top of the left finger. In this way, we would always close the right finger first.

Finally, I also worked with Bobby to define certain configurations and link these to button presses of the controller. We found the corresponding angles for the following configurations:

- Travel mode: tray 2 cm above ground
- Picking mode: tray placed touching the ground
- Open/close fingers: Close one finger than the other
- Shoulder up: Cubi's arm is horizontal to the ground in an elevated position
- Wrist straight or down: Wrist pointing to the ground

We then combined simple configurations to do more complex tasks. For example: to drop off an object we first move the shoulder up, then we move the wrist down, and finally we open and close the fingers.

Planning

I worked with Nithin to create a local planner. The local planner node subscribes to a global planner's topic which sends desired relative poses of where to go. The local planner then maps this relative pose to an angular velocity and linear velocity. Currently a very simple algorithm is used. If cubi is not aligned towards the goal it sends an angular velocity to align itself. Once cubi receives a relative desired pose within an angle threshold, then it considers itself aligned. At this point it moves forward towards the object. It will stop moving once it is within a threshold distance of the desired relative location.

The local planner also publishes its status: whether or not it is stuck and if it has reached the desired location. To determine whether we are stuck, we see if cubi is trying to move, but has not changed position in the past ten frames.

CuBi Project Management

We finally got Cubi assembled and we can all start the software integration of the vision system, manipulator, and planner. This week we spent a lot of time planning exactly what needs to be finished to be ready for the Spring Validation Demonstration. We left 10 days for integration and testing.

Challenges

Technical

The biggest problem we are facing is ensuring correct picking and dropping objects. When the floor is very slippery, cubi moves in the right finger and it pushes the block towards the left finger. When the left finger tries to close, the fingers get stuck because the right finger is closed, and the left finger cannot push the object into the tray. We are working with Paulo to test different finger configurations to solve this.

When dropping a toy, we inevitably do so from a distance. This causes the toy to bounce on the ground and can fall far from the desired location. In addition, when we drop a toy, the manipulator is upwards and obstructs the vision of the camera. We are unable to ensure that when we lower the manipulator, the toy is not under it. We are considering changing the location of the camera. We might need to use an ultrasound.

Team Work

Nithin: Has worked with me to program the local planner. We worked together to create the software architecture of how the global planner will interact with the local one.

Paulo: Manipulator was re-designed and the motors were integrated to actuate the grippers. He worked with me and several others to redesign several prototypes of fingers.

Laavanye: I worked with him to discuss solutions of how we can check whether or not there is something under the tray. He then wrote code to get the relative pose of alvar tags with respect to the base. The position was used by the local planner I wrote with Nithin.

Bobby: I worked directly with Bobby. We designed the pre-configured set of positions that will be used to pick up and drop objects.

Plans

In the future, I will be working with Nithin to create a more sophisticated local planner. Right now, the planner uses constant angular velocity and linear velocities to move. However, relative pose is equivalent to the error in position. We could use PID control to go towards the goal state. We also may want to not only go to the desired position, but also do so and end up in a desired orientation. This could be helpful when approaching a toy to pick it up.

I will also transition into working more on the vision side with Nithin. I will be using Deep Learning to classify the most common objects in classrooms (desks, chairs, etc…), as well as potential toys we can find in day cares. I will need to create a process to gather data.