Summer Internship Final Report

Haoran Li
haoranli@gatech.edu
August 8th – September 12nd

1 | Introduction to the Project

The project is about to achieve remote control of the robotic arm through a teleoperation interface, users can make operations such as click or drag using their mouse upon this interface to specify a grasp pose of the robotic arm and then control it.

In this research, we use three methods to help users identify the grasp pose:
1) Free positioning
2) Constrained positioning
3) Point-and-click

We selected these three techniques to incorporate increasing use of scene information into the grasp specification process.

The interface had already been completed by former researchers (Christian and Chidi), my work mainly fell in modifying the code to make it compatible on Ubuntu16.04 & Kinetic (original on Ubuntu14.04 & Indigo), which required me to look into both several dependencies and libraries' change-logs and the interface's code. After that, I also had to write arm-controlling node and then integrating it with the interface. I also made the visualization of the robotic arm in Rviz to run simulation first before activating the arm in real world. Finally I made sure that the real grasp pose corresponds to the pose specified by users through the interface, and tried to reduce the deviation between the expected pose and actual grasp pose.

2 | Meaning of the Project

The achievement of this project could serve to help the disabled, to help them get things conveniently, or serve as tools replacing humans to operate in dangerous situation. And the ultimate goal of the project is not only to achieve the remote control, but also to reduce the total workload for human during the operation.

Also Fu-Jen told me that my work would be incorporated into their project which aims at achieving the robotic arm's teleoperation through movement of tongue and also serves as tool to help the disabled to live a more convenient life.

3 | My Plan (copy from my Summer Plan document)

- Understand former intern's work (mostly reference Christian's daily working record). Most important thing is the rail_agile_grasp and the nodes network I think, so I will get down to them first.
- Determine the messages including information of the pose of gripper in the nodes network, then get that information.
- Write nodes to control the robotic arm and visualize the arm model in Rviz.
- Integrate these nodes to control robotic arm in the real world.
- Explore more… (particularly about why there is no reduction in total workload)

4 | What I Have Actually Done

- Read paper *A Comparison of Remote Robot Teleoperation Interfaces for General Object Manipulation*
- Read paper *Comparison of Tongue Interface with Keyboard for Control of an Assistive Robotic Arm*
- Understood the total project by looking into the code
- Modified the original code to make it compatible with Ubuntu16.04 & Kinetic
- Visualized the arm model in Rviz
- Set the relative position between the arm model and point cloud in Rviz
- Determined where the target grasp list stores, and added a service in the Point-and-click method to transfer the selected gripper's position
- Made the arm-controlling node as a client to call the service provided by the Point-and-click method to acquire the coordinate of target pose
- Understood TF tree and used TF API to transform the coordinate between different frames
- Got the robotic arm correctly moving to the target pose

5 | What Need to Be Done

- the grasp method after robotic arm attach the object (maybe hold the object and move to another position selected by user)
- Integrate my part with Fu-Jen and Ruinian's part

6 | What I Have Learned

To my own, I think the most important thing I learned is that you should always keep patient and hold on to your goal whatever happens. During this six weeks, I came across quantities of problems and what I can only do is to solve them one by one. Sometime I almost have the idea to give up, especially in first two weeks. At that time I was struggling to debug the project to make it compatible with Ubuntu16.04 & Kinetic. Because I was not very familiar with both ROS and Linux then, planting the project to 16.04 from 14.04 was a very tough and boring task. At first I just tried to change the dependencies and libraries to the former version, but I found it was implausible for PCL and VTK. So I

had no choice but to look into the code to find where the problem is. I was upset sometime because I couldn't make any progression. So I tried to have a conversation with Dr. Vela to ask for permission to continue the project on 14.04. However when Dr. Vela asked me where the problem is, I just can't tell him clearly which annoyed Dr. Vela and he criticized me harshly. From that day I started to recognize that there was no other way but to hold on to keep advancing. Things gradually turned better, at last I successfully fixed all problems and ensured that the project could run on 16.04 without suddenly shutdown. I also learned that it is so normal to meet with problems when do research, so just keep patient.

Of course, I got much much more familiar with ROS, Linux, and Moveit!. Now I could create ROS package, add executable source code, write topic publisher and listener, write service server and client, use TF frame transformation, understand what a launch file express breezily. About the Moveit! part, I get to know how to generate robot model from URDF file, how to set relative position between the arm and point cloud, how to control arm, how to define a robot, how to initialize a expected Rviz window by editing rviz file… At the beginning of July I was still almost a neophyte to Linux, now I can type command in terminal without uncertainty. All in all, I had a very substantial time here, at Intelligent Vision and Automation Lab. I will miss this place.

Besides these, this precious experience fostered my skill of self-studying which is another important quality required by nowadays techniques' rapid advance. Dr. Vela and PhD students here are all very arduous and have their devotion and passion in what they do. Due to their passion, I found myself more interested in studying things about robot and control. I plan to apply for graduate school in U.S. and continue study this.

After staying here for almost six weeks, I find my speaking English has been improved. That's really nice I think! Thanks Dr. Vela for hosting me, otherwise I wouldn't have such a great chance to expose myself to such complicated but urgent to be solved automation problems. I really learned a lot here.

7 | Daily Progress

8.11 Fri.
- Meet with Fu-Jen and get to know the whole project
- Start working on paper "A comparison of remote robot teleoperation interfaces.."

8.13 Sun.
- Finish reading the paper "A comparison of remote.."
- Scan the files and codes roughly

8.14 Mon.
- Reference Christian and Chidi's work record to understand what they have done and how
- Setup kinect (but come up with a little problem, and add a solution in the Start-up file)
- Learn about how to operate in Rviz
- Run three approached according to the start-up instruction
- Write the research plan

8.15 Tue.
- First meeting with Dr. Vela
- Setup the environment of the new computer

- Start to look into the code, to find out its organization

8.16 Wed.

- Continue setting environment
- Continue working on the code’s organization
- Prepare for the presentation about code’s organization and future work and schedule

8.17 Thur.

- Continue setting environment
- Make presentation and discuss what to do next with Fu-Jen

8.18 Fri.

- Continue setting environment
- Review the tutorial of ROS

8.20 Sun.

- Learn the type of the message and service in the project
- Identify the topic contains information about the gripper’s pose
- Start reviewing tutorial of MoveIt!

8.21 Mon.

- Move to new office
- Set environment on 16.04&Kinetic

8.22 Tue.

- Meet with Ruinian
- Start debugging of the code
- Continue setting environment on 16.04&Kinecit

8.23 Wed.

- Meet with Chidi and learn more about the project
- Set environment on 14.04&indigo (to make controlled experiments with 16.04)
- Review the paper “A Comparison of Remote Robot Teleoperation...”

8.24 Thu.

- Continue setting environment on 16.04&Kinetic
- Look into PCL and VTK package and their change-log
- Modify the code to be consistent with the news in PCL and VTK

8.25 Fri.

- Continue setting environment on 16.04&Kinetic
- Find out a old-version plugin and remove the corresponding code
- Start doing controlled experiment to find out the problem
- Get the project working successfully on 16.04&Kinetic

8.26 Sat.

- Find the heuristic-based algorithm could only find several gripper poses, and start looking into code to solve it
- Get more familiar with the code of the project

8.27 Sun.

- Find that it is a wrongly initialized variable cause the problem
- Summarize this week’s work and write report
- Plan next week’s tasks

8.28 Mon.

- Fixed the Eigen problem and got the whole project working correctly
- Generated package from arm’s URDF model, and visualized in RViz

8.29 Tue.

- Visualized the robotic arm and point cloud in the same RViz window
- Reviewed the tutorial of Moveit!
- Tried to set the position of robotic arm in RViz

8.30 Wed.

- Reviewed the tutorial of Moveit! again
- Second meeting with Dr. Vela
- Looked into the code of the robotic arm model

8.31 Thu.

- Read paper named Comparison of Tongue Interface with Keyboard for Control of an Assistive Robotic Arm
- Summarized two papers
- Learned more about tf and RViz
- Looked into the Point & Click code again and got more familiar with its interfaces

9.1 Fri.

- Started to write code to control the robotic arm in RViz
- Continued to find way to set the relative position between arm and point cloud
- Discussed with Fu-Jen about what to do next

9.3 Sun.

- Continued writing code to control robotic arm
- Summarized the past week's work

9.5 Tue.

- Debugged code of Moveit_pr2 to make it compatible with 16.04 and Kinetic
- Reviewed tutorial of Moveit!
- Wrote code to control the arm
- Discussed with Dr. Vela and Fu-Jen and got to know Fu-Jen's work

9.6 Wed.

- Set the relative position between point cloud and arm model
- Kept writing code to control arm
- Modified arm code given by Fu-Jen to make it launch smoothly on 16.04
- Started to learn about the frames and TF API

9.7 Thu.

- Successfully wrote code to control the robotic arm
- Started writing code to achieve coordinate's transformation

9.8 Fri.

- Finished coordinate's transformation part
- Started to integrate the arm-control node with the whole project
- Discussed with Fu-Jen about my following plan

9.9 Sat.

- Finished integrating the project using publisher and subscriber
- Discarded the publisher and subscriber's style and started writing code using server and client to maintain the grasps-cycle function

9.10 Sun.

- Finished integrating the project using server and client
- Started to created the start-up file and other documents
- Summarized the past week's work

9.12 Tue.

- Created suitable TF transformation between base_link and camera_link to make robotic arm fit the table more exactly
- Reduced the deviation between realistic target pose and expected target pose
- Got the whole project working smoothly
- Simplified the start-up steps by integrating the launch file and rviz file

9.13 Wed.

- Finished writing start-up instruction and final report
- Videoed the step to run the project and also its result
- Third meeting with Dr. Vela and presented the project

9.14 Thu.

- Uploaded the project onto Google Drive and Github