Individual Lab Report - 4

Autonomous Reaming for Total Hip Replacement

HIPSTER | ARTHuR

Gunjan Sethi

Team C:

Kaushik Balasundar | Parker Hill | Anthony Kyu
Sundaram Seivur | Gunjan Sethi

March 23 2022

Carnegie Mellon University
The Robotics Institute

Contents

1 Individual Progress

1.1 Landmark Collection

The autonomous reaming pipeline begins with the surgeon collecting landmark points on the pelvis using a registration probe. The probe is shown in Figure 1. The marker geometry will be screwed on top of the tip. This will help in tracking the marker.

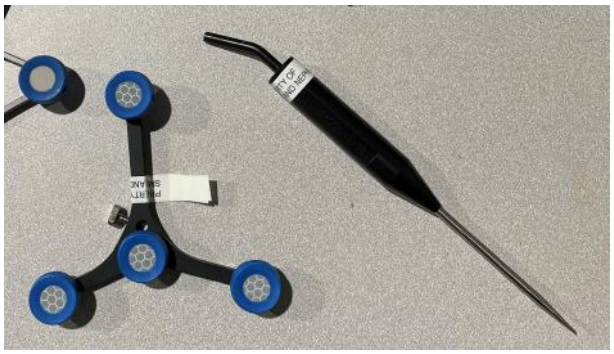

Figure 1: Registration Probe

1.1.1 Typecasting to PointCloud2 Type

To develop the functionality, the first task involved using the 6DOF marker pose tracked by the camera to extract the translation x, y, z values. Collecting this translation values as a landmark for the pointcloud means collecting the centroid of the marker geometry. This point is considerably offset from the probe tip and hence, current efforts involve understanding how to account for this offset. To typecaste the pose into Pointcloud2, `PointCloud2Modifier modifier` and `PointCloud2Iterator<float> iter` were used. The modifier resizes the pointcloud in tandem with the number of points collected. At this time, this value is hardcoded to 500 points. The iterator is used to populate the data array of the pointcloud. This array stores the x, y, z values in form of a byte array.

1.1.2 Publishing and Visualizing

Next, the point is published as a PointCloud2 message. At this time, the publishing happens continuously at 50Hz. However, in the future, this will be changed to a slower frequency to account for the time the surgeon might take to navigate the probe to a new point of interest. Once the message was published, it was visualized in RViz. Figure 2 shows a screenshot of the same.

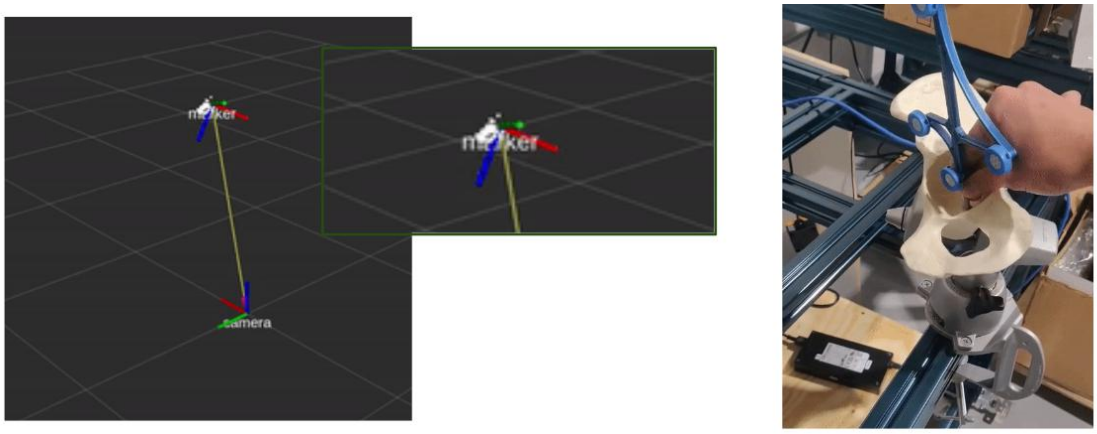

Figure 2: (Left) RViz Showing Collected Pointcloud (Right) Using Probe to Collect Points

1.1.3 Testing

To test if the pointcloud was collected at the right scale, an object of known geometry was used. Once the actual physical dimensions of the object were noted as ground truth, points on the object were collected using perception system. The dimensions of the pointcloud were calculated. Tests to study the effect of holding the probe in various orientations were also performed. The accuracy varied between 1-2cm which satisfied the qualitative tests for pointcloud collection. In the future, further development will be done to make the collection within 1-3mm accuracy. Figure 3 shows the object used for the test.

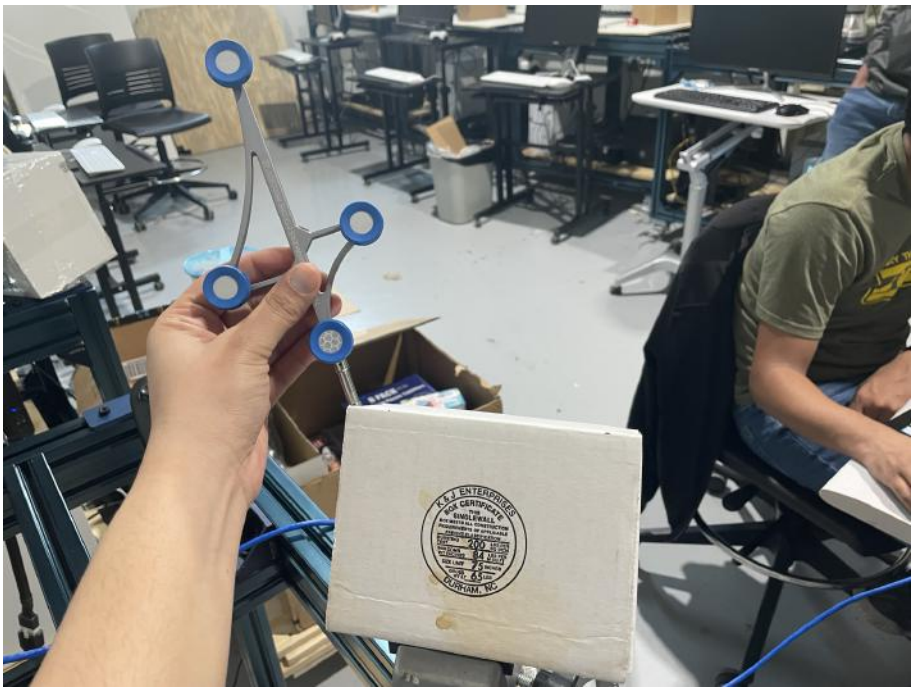

Figure 3: Scale Test with Object of Known Geometry

1.2 Project Management Progress Review

All the project management components including risks, schedule, SVD and FVD test plans were consolidated and presented for the PDR presentation.

2 Challenges

2.1 Understanding ROS PointCloud2 Message Type

Since the PointCloud2 message type stores x, y, z values as a byte array, a lot of time was spent reading the documentation and exploring code samples online to understand the data structure thoroughly. Further, there is no direct way to add values to the pointcloud without using the modifier and iterator APIs. Working with these in tandem with incoming values was challenging and involved a considerable time in debugging for small errors.

2.2 Picking Publishing Frequencies

Currently as a proof-of-concept, points are added at 50Hz. This involved ensuring that the pointcloud is resized to a considerable number to accomodate for all incoming points. At times, every alternate point was added to the pointcloud. This is still a challenge and various pointcloud collecting frequencies are being explored.

2.3 Obtaining Pose of Probe Tip

Offset from marker geometry centroid and varied orientation in holding the probe, introduces a significant amount of error upto 2cm. Since the tip is lined up with 2 fiducials, offsetting using translation was tried. However, accurate probe tip position was not obtained. To address this challenge, conversations with the sponsor were crucial. Going forward, the probe CAD model will be used to exactly locate the probe tip position given the marker geometry centroid.

3 Team Work

Following are the tasks accomplished by the team members since the previous ILR.

- ***Kaushik Balasundar*** worked alongside Gunjan in developing the landmark capture capability to convert the marker pose into pointcloud2 messages. This was then visualized on RViz and verified with ground truth to ensure the data is of the right scale. He also modified the URDF to incorporate new design changes and also added a force-torque plugin into the Gazebo simulation. He assisted Sundaram in extracting a trajectory planned by the Pilz planner for preliminary testing of the MPC controller.

- ***Parker Hill*** worked on creating a end-effector reaming adapter for this progress review, going through the process of designing, 3D printing, assembling, and analyzing the assembly. He also worked on the PCB assignment, going through the steps to create the board, finding parts, and verifying performance for submission. Finally, Parker helped to perform the motor speed and torque test as well as helped to create a ROS node which runs using Julia.

- ***Anthony Kyu*** worked on further developing the Model Predictive Controller, refactoring the code to use previously overlooked functions implemented in the RigidBodyDynamics.jl Library, and simplifying the dynamics model to only include joint positions and velocities in the state to simplify the overall MPC to get an initial code base. Then, in order to get the MPC to converge, Anthony worked to create an appropriate initial guess for the input/torque trajectory based on the given state trajectory, working on several approaches. Once the MPC converged Anthony worked on visualization, optimization and initial code implementation of the MPC into ROS using RobotOS.jl library in Julia. Anthony also collaborated with Sundaram to get a sample trajectory to work with and sanity checked Parker's PCB.

- ***Sundaram Seivur*** worked on generating the trajectory for the arm to follow and worked on setting up the necessary packages and functions to do so. He first worked on setting up the arm and connecting it with a computer via ethernet. To do this, he set up the IP address of the system and validated if the joint parameters are visible via the KINOVA API. Sundaram also worked on setting up the Pilz Industrial Motion Planner in MoveIt! to be used as a trajectory planner. This was a challenging process as a completely new planning-pipeline had to be set up and had to be included in the convoluted code written by Kinova. He also changed the IK solver from KDL to IKFast for which he set up a docker image and created an IKFast plugin. With all the improvements, Sundaram was successfully able to generate deterministic trajectories and send these commands to the arm to validate it's motion. Sundaram also collaborated with Anthony to send joint-state trajectories to the MPC controller and computed the cartesian pose of the end-effector.

- ***Gunjan Sethi*** worked on developing the landmark capture functionality along with Kaushik. During this, the main tasks involved understanding the PointCloud2 message type in ROS, prototyping a simple pointcloud collection function and visualizing it in RViz. Later, they performed a test to verify the scale of the collected pointcloud and to study the effect of orientation of registration probe on the collected points. Gunjan also prepared slides for and presented the project management portion of the PDR.

4 Plans

For future work, the following (individual) tasks have been planned for the MRSD project.

4.1 Three-Point Landmark Collection

As discussed with the sponsor, for an initial step for registration, three point-to-point correspondences will be recorded. The next sprint will involve developing and integrating this functionality into the perception subsystem.

4.2 Incorporate Probe CAD Model into Pipeline

As requested, the sponsors have shared the CAD model files for the registration probe. This can be used to obtain the exact transformation between the marker geometry centroid and the probe tip. Thus, the use of the CAD model file needs to be incorporated into the current pipeline.

4.3 Prototype Registration Techniques

Finally, current registration techniques need to refined and integrated into the perception system, in collaboration with Kaushik. For the next progress review, some metrics will have to evaluated to showcase the performance/robustness of the registration techniques.