u2.cloud

Software Engineering Project
Mahlet Birhanu
Brianna Burton
Oleh Kozynets
Doiriel Vanegas

The first 3 meetings detail our progress before we were asked to write a formal weekly report.

10/11/17

- Problems with understanding the theory. Tasks to read the articles that Cansen gave
- Divided up the projects to read through (1-4) and find out the most important things. Report on trello
- Find out about PCL library
- Build 3D Korn project

13/11/17

- Decided on a SDLC - combination of waterfall and agile
- Try to build 3D korn and project 3 and choose one to improve
  - Whichever one works first we will optimize and analyze
- Find out about hardware requirements

21/11/17

- Doriel and Oleh have made Progress on building projects 1 and 3
  - New ideas from Roger so might build easier
- Brianna has extra info about filters and smoothing and alternatives to implement
- There is a up-and-down moveable setup for kinect...implement this in project
- Mahlet is finding out about registration libraries

26/11/17

Everyone has connected to slack

What we did this week:
Oleh:

- Built 3D Korn project
- Built project 3 (crashed upon opening)
- Tried to connect kinect to PC...will try it again this week
- Helped Doiriel with VTK to run the project

Doirel:

- Built 3D Korn project

Mahlet:

- Bought a new laptop for the cpp project (old laptop too old for this project)
- Looked at registration algorithms for all groups https://drive.google.com/open?id=196DaWLs_zjsl51yuly6L5ILBidueyaHock1hi8OyuTs
    - Conclusion: major difference between algorithms
        - 1st used pre-alignment (converged very well)
        - ICP:
            - 3 groups used the minimum distance between groups
            - 3rd group used point to plane algorithm
            - 1st group used ICP and SVD and multiplied two matrices

Brianna:

- Installed all related components and am trying to build 3D korn on my laptop
- Looked at filters and smoothing for all the groups https://drive.google.com/open?id=11otyPpxLThg6ALObZNGD4zuQN_2uF3EibLXFL2WyJUA
    - Conclusion:
        - Most groups used pass through and voxel filters but first group used less filters
        - All groups used the same meshing Greedy Triangulation (GT), Grid Projection (GP), Poisson reconstruction
        - Bilateral Filter used by group 4 good idea to filter raw daata
        - Possible alternative filters include ApproximateVoxelGrid, CropBox is a filter that allows the user to filter all the data inside of a given box, MedianFilter

3/12/17

Oleh:

Oleh fixed the problems in the project with the debugger, and now the project can be debugged if it has problems and also installed and tried to run open CV with the PCL examples. He built VTK with the debugger release. He will work with other groups to get this working. He helped Mahlet to run the project.

Doiriel:

Doiriel looked at the source code to find where the filters were applied and determined how they work and which parameters are required. She opened some point clouds from the previous group and used their program to register them and found that the initial registration doesn’t work very well. The system doesn’t register them correctly and the program does not get a 3D object as a whole but parts of it.

Brianna:
Brianna analyzed the pre-alignment of the first group and how it works before using ICP. They used SLAM through ELCH to filter their data, but their initial clouds were not good enough. Their method does not work well and the pre-alignment really needs to be improved, perhaps by feature detection. This also means we need to obtain improved initial point clouds. She also was able to build the project this week and obtained the VTK with debugger release from Oleh.

Mahlet:
Mahlet's new laptop arrived, and she was able to install all the components to build the project including QT, VTK, PCL, etc. She built the project. She also determined that point to plane registration as used in group 3's is the best type of registration and we should change the registration of 3D Korn to use point to plane.

During our meeting, we came up with a clear project plan for the coming weeks. We have **5 main goals** for the project:

1. Improving the field of view of the sensor by moving it closer to the object and obtaining multiple point clouds for the same angle of the turntable (for ex, one for head and shoulders, torso, legs, and feet). We will register these point clouds together, and then register the resulting point cloud for each of the angles, and obtain a mesh.
2. Changing pre-alignment to include feature detection such as descriptors
3. Group 3 used ICP point-to-plane – use this instead of mean distance (point to point) like the other groups
4. Change the filters and smoothing to include MedianFilter, Cropbox, etc
5. Improve the GUI: more specifications to come. It needs a button to indicate the program is working and to improve the way it works with the code.

The first 3 goals are the most important, and we will work on those first to obtain a watertight mesh. We will base our project on the 3D Korn project and implement ideas from all of the groups as well as our own ideas.

Tasks for this week:

- Go to the robotics lab and begin trying to scan and be sure our laptops work with the kinect
- Write new code for the registration of the point clouds, incorporating ideas from group 3, who had excellent point cloud results

10/12

Mahlet
--progress
She worked on incorporating a registration function that uses ICP of normals of points.
To first experiment on its results on the available point clouds, the function performs registration only on two point clouds. It reads point clouds from a database, registers them and writes an

output point cloud into a given directory. The goal was to compare the result with the registration result of 3D korn.
--challenges
Since some of the PCL library seems to be missing, she had a hard time compiling the new project.

Oleh
Along with other group members we tried to connect our PCs to kinect, as a result, Mahlet and Doiriel can work with kinect, no problems with the connection. His PC still does not work, but taking into account that other group members can make a connection easily, this is not a problem in our work on project anymore. Cansen's OpenCV and PCL sample projects were compiled on his Windows machine. The main obstacle was that the samples were initially written for Linux. OpenCV sample project was made as Visual Studio 2015 project, due to the problems with adding OpenCV to Qt Creator project on Windows. The PCL project in made as Qt project for Windows, and everyone in our group can run it easily on their PC. He will proceed with the tasks that we have now and will look at Mahlets and Doiriel work on registration and help them to overcome obstacles with PCL.

Brianna
She also worked on integrating the registration of the third group in the first group's code, specifically writing new code that implements the filters from group 3 onto group 1's registration. As group 3 implemented filters much earlier, it seems that this is much better and more robust for our project. Although nothing is working quite yet, she will implement and merge this filtering procedure with the work of Doiriel and Mahlet this week to create a new registration procedure (including both the new filter procedure and the point to plane registration). She was having issues with windows update when we had the Kinect sensor (which have now been resolved) so she is not sure if Kinect works with her laptop, but is satisfied that it worked with both Mahlet and Doiriel's PCs.

Doiriel

Find inside the source codes of groups 3 and 1 the specific parts where the different registration algorithms were implemented. Since registration of group 1 is not good enough and group 3 seems to have the best registered point clouds, I tried to include group's 3 registration into group's 1 source code. However, there are some incompatibilities between the source codes that make more complicated trying to merge them, for instance regarding the use of filters inside registration functions. For this reason, Mahlet and I decided to create some functions to import, register, and save point clouds using ideas from group 3 (iterative closest point with normals) and adding positive elements from group 1, this way we can have more control on the structure of the code.

Shown below are some of our screenshots from using the Kinect with the laptops