UNIVERSITÉ DE BOURGOGNE

Master IN COMPUTER VISION AND ROBOTICS

PROJECT REPORT-COMPUTER SCIENCE

PROJECT TITLE: "3D Scanner "

Group NAME
Group 2

CO-ORDINATOR
Yohan Fougerolle

Centre Universitaire Condorcet - uB, Le Creusot

October 2016

Summary

This report includes brief and key information about the task which we have achieved successfully during last week in addition to the task on which we are currently working on with future plans and backup.

In our last weekly report, we have mentioned that we are working on interfacing Kinect V2 with Qt, solving problems and to come up with a sample GUI design and filtering images by depth with saving data. By the dedication and hard work of the group members we have been able to complete all the three task including interfacing Kinect V2 with Qt, a sample GUI for basic building block and real time filtering images by depth and saving data in .pcd or ply format from Kinect V2 till the end of this week.

Half of the group members are now working on writing module for data acquisition and depth filtering, ICP and on the GUI with rest working on Marching Triangles, Marching cubes, Triangulation algorithms.

We have also planned some optional steps that need to be done for the upcoming week such that 3D Reconstruction. To teach everyone the general knowledge and try to nivelate skills of all the group members, we planned seminars in group and in this 4^{th} week. We will have a seminar about Planar Removal taught by Sepideh about the available techniques for performing planar-removal operations. Which will lead us to apply better algorithm which suits our results and will be used in our project.

1. Task Achieved

a. Interfacing Kinect with Qt

We successfully interfaced Kinect V2 with Qt by changing the PCL version from 1.6 to 1.8 and installing Visual Studio compiler 2015 with Qt version (MSVC 2015) which is compatible with Kinect V2. This new PCL version 1.8 has solved almost all of our problems for interfacing the Kinect V2 removing most of the errors which we were facing before. Like this, we've been able to perform live streaming of data.

Till now PCL 1.8 is configured on 4 of our computers, as rest of the system are not compatible with Kinect v2 because it accepts only USB3, but as our group are divided into two developing team with researchers so it isn't a problem, also for some of our mac users it was not possible to have the systems working with Kinect V2 and some PCL components weren't compatible (So they have installed Windows in a separate partition in order to work with PCL).

b. Filtering by Depth

Below here is the result which we got after filtering by depth which we displayed by using MeshLab, which can also be viewed through PCL_Viewer, We have filtered depth pixels with a range from 0.5m to 1m with Kinect V2 providing a range of 4.5m. We can modify depth depending on the camera position and the environment.

Figure : 1 (PCD with filtering by Depth)

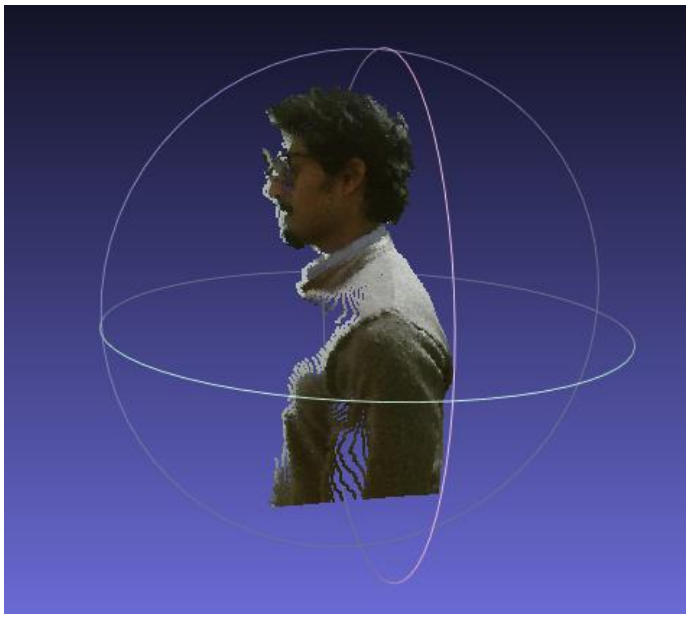

Figure : 2 (PCD with filtering by Depth)

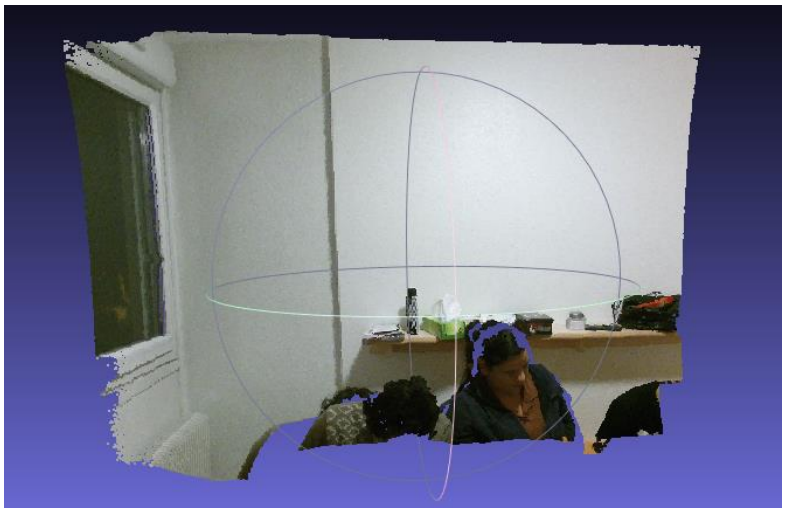

Figure : 3 (PCD without filtering by Depth)

c. Storing data .pcd and .ply format

After Filtering by depth we have been able to store data in real time in .pcd or .ply format with a sequence of images below here is the link for the data.

Link: https://1drv.ms/f/s!AmOsVDUME6ulh3oFIJkFgPSrGLb7

d. Sample GUI

We have been come up with a sample GUI. We designed a user interface that is able to divide all the principal functionalities, taking as reference other 3D scanning software's such as MeshLab,3D builder and Skanect.

The distribution of the functionalities is done in such way that it follows the basic steps of performing data acquisition and processing, as follows (this is just a sketch, more functionalities will be added and some of the already included can be modified):

Figure below shows options for importing and saving pcd data with capturing RGB Camera ,3D reconstruction and Live point cloud data we will be working on linking Kinect V2 with GUI for next weekend with functionalities mentioned below.

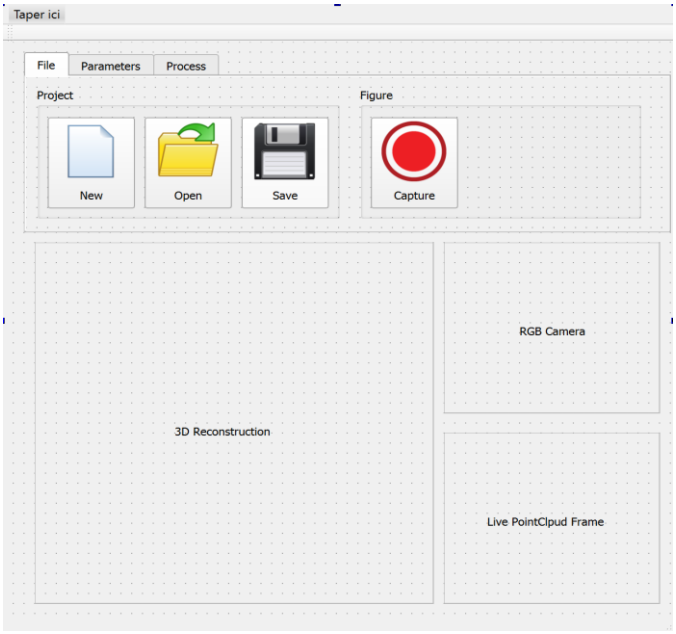

Figure : 4 (GUI Sample = File)

Figure below shows options for setting Parameters and process which can be done in order to build a 3D model.

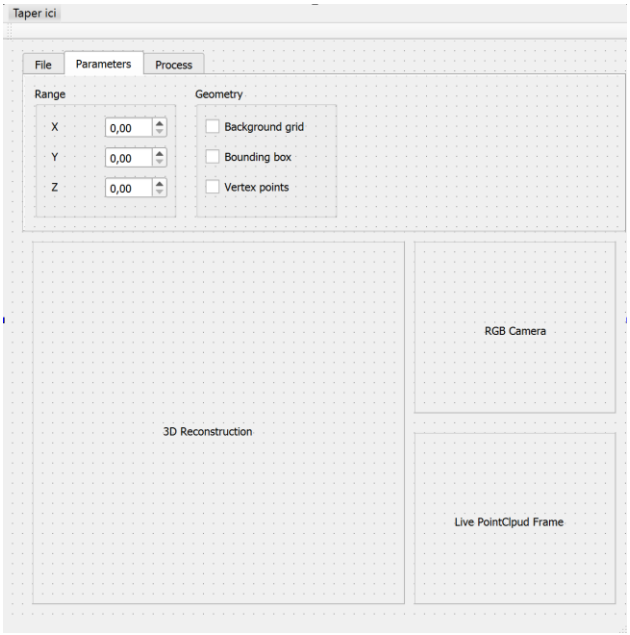

Figure : 4 Figure : 4 (GUI Sample = Parameters)

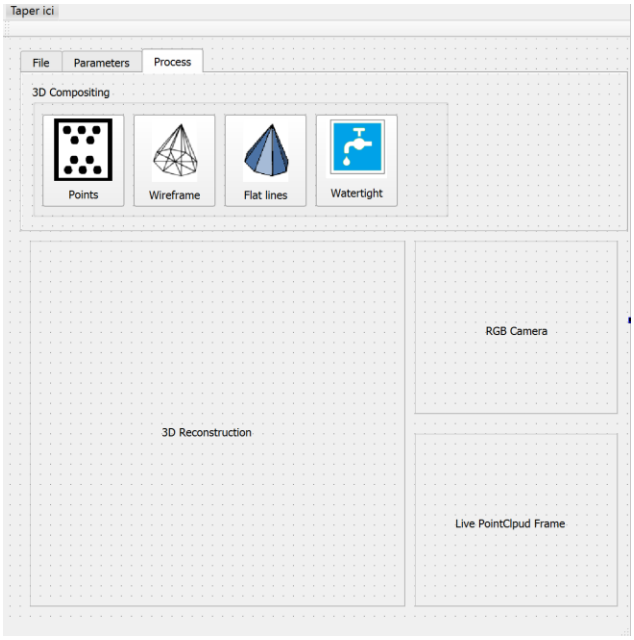

Figure : 4 (GUI Sample = Process)

Fig. : Sample GUI showing the arrangement of possible functionalities into the software

According to the proposed sketch, the functionalities to be included into the program are meant to be classified according to the common order of work for 3D scanning applications. The idea also is to show in real time the 3D reconstruction of the object/person to be scanned, along with other useful information such as the bounding box, live pointcloud, and different sensor information from the Kinect.

2. Task Assigned for Next Week:

As things are getting clearer we are assigning our tasks accordingly:

Acquisition	Algorithms	Interface
• Building Classes for data acquisition	• ICP • Planar removal • Triangulation • Feature Tracking	• Research on possible functionalities • Integration of already working algorithms

1. All of the individual classes developed by the group members will be integrated into the main interface, always checking for performance and solving possible conflicts between the different algorithms.
2. We are now working on tasks that we assigned for one week. We can extend the deadlines but the main goal is to succeed developing theses functionalities in a way of being able to provide a better software at the end of the project for our final goal.

3. We will also focus on the individual progress of our group members. We need to teach each other's our knowledge in order to improve our general group skills. As mentioned in the summary till this week we will include in our schedule a group seminar to explain notions that are non-general for everyone.

4. Figure out the algorithm for automatic planar removal (when we will get data from Kinect of a body, we will certainly have a planar table that the body is standing with and we need to remove it. we think we can find a nice automatic algorithm for this).

5. Study and implement the algorithms for compression of RAW data from Kinect. This must increase the processing speed, lower the amount of used memory etc. The algorithm will accept a set of pcd images as input, and minimizes the number of points without losing a lot of quality. (Down sampling, Outlier filter, etc.)

3. Future Work:

As we have mentioned in our previous report that The Project will be done after going through different steps firstly by basic handling, Designing the interface and lastly adding additional features in the software to make it more user friendly but for now we are working on how to stream through GUI and saving data with 3D reconstruction and meshes which we can apply with a single click.

Project Code:
Gitlab Link:
https://gitlab.com/Akhtar52/3D-Scanner-VIIA/tree/master
Google Skills Sheet of all the members with sample Gantt chart:

https://docs.google.com/spreadsheets/d/1A3AcKmMm6rLwND0CRhzrzFeirF5GqxXA3xCpi3jQ3xQ/edit?usp=sharing