Machine learning and pose estimation for autonomous grasping with collaborative robots tutorial

Tutorial part 1 : How to save point-cloud and RGB image for R200

Victor Talbot

August 17, 2018

This tutorial aims to present how to obtain a cloud point and an RGB image thanks to the application developed by me in order to be able to use them in the neural network.

I will start by showing you the different tools needed and then show you how the program works.

1 Tools installation

You will need several different tools to compile and create the program.

1. linux 16.04

2. ROS kinetic

3. Open CV

4. Librealsense

5. Realsense

1.1 Linux 16.04

If you are a new user of Linux i suggest to follow this tutorial : `http://lea-linux.org/documentations/Installer_Ubuntu_16.04`.
You can also work with Ubuntu 14 but you will need to adapt some command.

1.2 ROS kinetic

ROS is a software design to control robot. It's works with different nodes who exchange information. I recommended the kinetic version which is the stable one for working with Librealsense.
You can find how to install ROS in your machine here: `http://wiki.ros.org/kinetic/Installation/Ubuntu`. I also recommend to follow the ROS tutorial if you are a beginner.

1.3 Librealsense

Librealsense is an SDK created to use Intel cameras like the R200 or SR300 and Realsense is a develloped layer to use it under ROS. You can find the source files on their github respectively here: `https://github.com/IntelRealSense/librealsense.git` and here `https://github.com/intel-ros/realsense.git`. And you can find instalation tutorial here `http://wiki.ros.org/realsense_camera/Tutorials/Building_librealsense_from_Sources`

1.4 Verifies installation

Once you have correctly installed all the tools you can verifies your installation.
First open a terminal, and run the following commands

```
roscore
```

If ROS is setup correctly you will see the core running like this :
You can check the correct installation of realsense by running the following command with a SR300 plugged in a new terminal.

```
roslaunch realsense_camera sr300_nodelet_rgbd.launch
rosrun rqt_image_view rqt_image_view
```

and this one in a third terminal.

```
rosrun rqt_image_view rqt_image_view
```

You will a windows with the output of the camera.

```
Checking log directory for disk usage. This may take awhile.
Press Ctrl-C to interrupt
Done checking log file disk usage. Usage is <1GB.

started roslaunch server http://kebsox-N751JK:40397/
ros_comm version 1.12.13


SUMMARY
=======

PARAMETERS
 * /rosdistro: kinetic
 * /rosversion: 1.12.13

NODES

auto-starting new master
process[master]: started with pid [5963]
ROS_MASTER_URI=http://kebsox-N751JK:11311/

setting /run_id to 51d8da20-762f-11e8-8bde-4851b73661a4
process[rosout-1]: started with pid [5976]
started core service [/rosout]
```
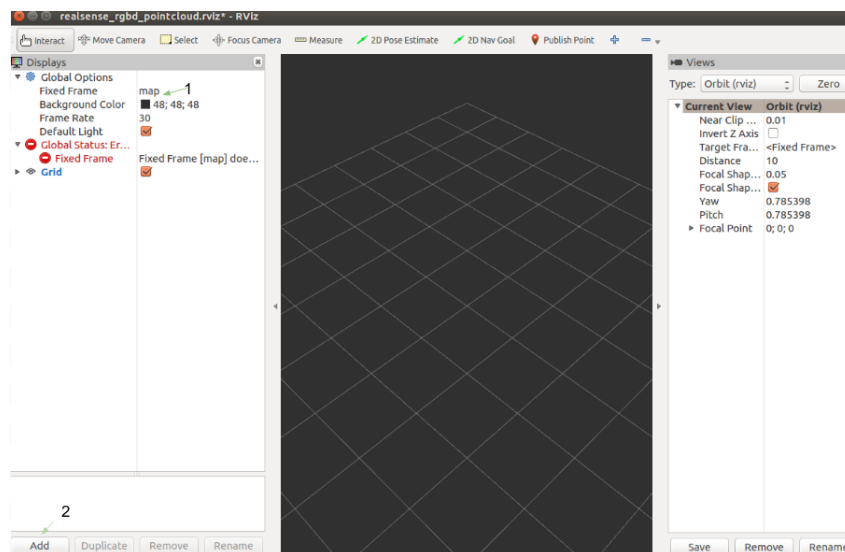
1.5 Using RVIZ to get your first frame

A more complete viewer is available with the classic librealsense installation. You can use it using the command below.

```
rosrun rviz rviz -d rviz/realsense_rgbd_pointcloud.rviz
```

In order to uze RVIZ you will need to do some modification

1. change map to camera link

2. in the add windows choose whats you want to display.

2 Get and compile the apk

In this section i will show you how to get my code and how to compile it. You can find the code for using the camera in my github here: `https://github.com/Kebsox/Robot-grasping`

First you need to create a catkin space. If you already do that you need to skips this step.

1. Create catkin

```
mkdir -p ~/catkin_ws/src
cd ~/catkin_ws/src/
```

After that you need to get the code

2. Clone repertory

```
git clone https://github.com/Kebsox/Robot-grasping.git
```

You can find in this directory the cmake with comments to show whats you need to include. .

3. Build the catkin

```
catkin_init_workspace
cd ..
catkin_make clean
catkin_make
source ./devel/setup.bash
```

Normally you should see somethings like

```
Linking CXX executable /home/kebsox/catkin_ws/devel/
...lib/projet_victor/main
```

4. Run the software

Now the final step you have to do is to run this command:ros

```
rosrun projet_victor main
```

3 Use the software

If everything went correctly in the previous section you should see two windows.

The first with the image in color and the second the cloud point of the image. You can save both information by clicking on one of the two images with the thumbwheel. The images will be saved in the root of the catkin following n name format. You can start numbering again by deleting the index.txt file that will have appeared at the root of the catkin.

Then you will be asked if you want to continue shooting or not.

If you want more information the code is fully commented and you can send me an email at vpj.talbot@gmail.com