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

Tutorial part 1 : software installation

Victor Talbot

July 12, 2018

In this tutorial we will see how set your computer for working with my project.
In this tutorial we will install :

1. linux 16.04
2. ROS kinetic
3. Open CV
4. Librealsense
5. Realsense
6. CUDA
7. cuDNN
8. Pointcloud library

1 Software installation

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 14.0 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 Open CV

OpenCV will allow us to create windows and process RGB image. You can follow this link for the installation : https://docs.opencv.org/2.4/doc/tutorials/introduction/linux_install/linux_install.html?

1.4 Librealsense and Realsense

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 Start by navigating to your catkin and open a terminal.

You will have to take the Realsense ROS v2.0.3 and Librealsense 2.10.2.

Be carefull you have to install Librealsense first

1.5 CUDA

The most difficult installation starts now. If you make a mistake you can have a lot's of problem with your computer.

First you have to verify you working capability. She have to be higher than 3.1. You can check the GPU list here : https://developer.nvidia.com/cuda-gpus

If you have a compatible GPU we can move forward.

I recommend to install first the latest driver for your nvidia cards. `http://www.nvidia.fr/Download/index.aspx?lang=uk` After that install CUDA 8.0: `https://developer.nvidia.com/cuda-80-ga2-download-archive` The 8.0 version is the stable one for my code. Dowload the debian file and follow the instruction on your screen. They will ask you if you want to install the latest driver for your graphic card. Press no. Then follow the instruction. To verify if your installation works properly open a terminal and type :

```
# nvcc -V
```

if you see a Nvidia message the installation work properly. Then try to run a sample.

1.6 cuDNN

cuDNN is a extension for CUDA. Is really simple to install it. First you have to create a Nvidia developer account then go to this website : `https://developer.nvidia.com/rdp/cudnn-archive` and download the version 5.1. you just have to add the two files in your cuda directories.

1.7 Pointcloud library

Pointcloud library is design to works with pointcloud. She will allow us to save a process pointcloud from the camera.

You just have to follow this link : `http://pointclouds.org/documentation/tutorials/compiling_pcl_posix.php`

To verify your installation I recommend you to use the example given with your installation. You just have to launch one code, if it's works correctly everything should work.

If you have any troubleshooting contact the support of each software to get help.

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