Asus XTion PRO Live: OpenNI2 compilation and install instructions

[*Last update: 2015 , 8th January*]

Contents [hide]

Table of Contents

MyzharBot uses an Asus Xtion Pro Live RGB-D sensor to detect obstacles, to create a 3D map of the environment and to localize itself in the map. Asus Xtion Pro Live is fully supported by the ROS framework (follow the guide), but if you want to use it outside ROS you need to compile OpenNI2 by yourself following this guide.

You can compile OpenNI2 by yourfself following this guide, or you can download the precompiled package and install it jumping directly to “Installation”.

Compilation phase

1. Open a terminal (Ctrl+Alt+t)
2. Install dependencies:

Shell

```
sudo apt-get install -y g++ git python libusb-1.0-0-dev libudev-dev freeglut3-dev doxy
```

3. Create a “devel” folder:

Shell

```
mkdir ~/devel
```

4. Navigate to “devel” folder:

Shell

```
cd  ~/devel
```

5. Clone OpenNI2 repository:

Shell

```
git clone https://github.com/occipital/OpenNI2.git
```

6. Navigate to OpenNI2 folder:

Shell

```
cd OpenNI2
```

7. Set the compile flags to build for the Jetson:

   A. Run:

   Shell

   ```
   find . -iname platform.arm
   ```

   You will get:

   ./ThirdParty/PSCommon/BuildSystem/Platform.Arm

   B. Run gedit:

   Shell

   ```
   gedit ./ThirdParty/PSCommon/BuildSystem/Platform.Arm
   ```

      i. Find:

      CFLAGS += -march=armv7-a -mtune=cortex-a8 -mfpu=neon -mfloat-abi=hard

      ii. Replace with:

      Vim

      ```
      CFLAGS += -march=armv7-a -mtune=cortex-a15 -mfpu=neon-vfpv4 -mfloat-abi=
      ```

      iii. Save the file and exit

8. Add support for pthread library:

Shell

```
gedit ThirdParty/PSCommon/BuildSystem/CommonCppMakefile
```

   A. Search the line

   [Line 97] *LDFLAGS += -Wl,-rpath ./*

and the line

[Line 98] *OUTPUT_COMMAND = $(CXX) -o $(OUTPUT_FILE) $(OBJ_FILES) $(LDFLAGS)*

Add the code between the two lines:

LDFLAGS += -Wl,-rpath ./

Vim

```
ifneq (“$(OSTYPE)”,”Darwin”)
    LDFLAGS += -lpthread
endif
```

OUTPUT_COMMAND = $(CXX) -o $(OUTPUT_FILE) $(OBJ_FILES) $(LDFLAGS)

B. Save the file and exit

9. Add Samples to the build chain to test the correctness of the installation:

A.

Shell

```
gedit Makefile
```

i. Add this line at the end of the file:

Vim

```
core_samples: $(CORE_SAMPLES)
```

B. Save the file and exit

10. Start compilation:

Shell

```
PLATFORM=Arm make
```

11. Compile tools to get “NiViewer” (NiViewer needs GLUT enabled):

Shell

```
GLUT_SUPPORTED=1 make tools -j4
```

OpenNI2 default Test

1. Navigate to samples folder:

Shell

```
cd ~/devel/OpenNI2/Bin/Arm-Release
```

2. Connect Asus Xtion Pro Live to the front USB Port
3. Test "SimpleRead":

Shell

```
./SimpleRead
```

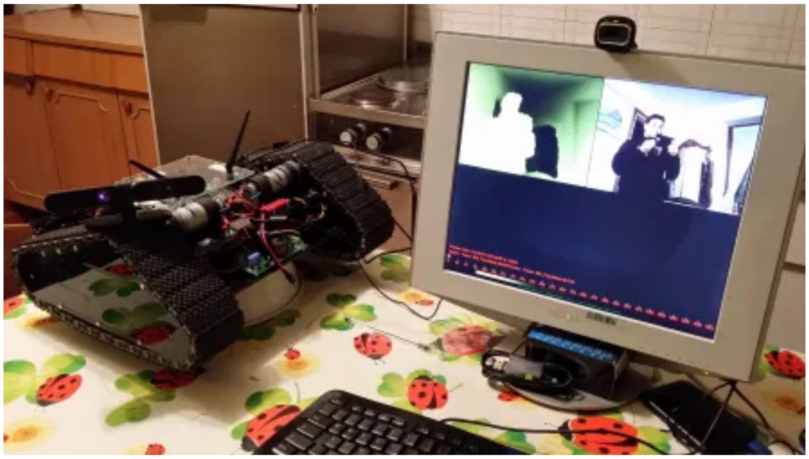
If everything is ok you can see a list of number constantly updating… this is the distance from the center point of the sensor to the nearest object in front of it.

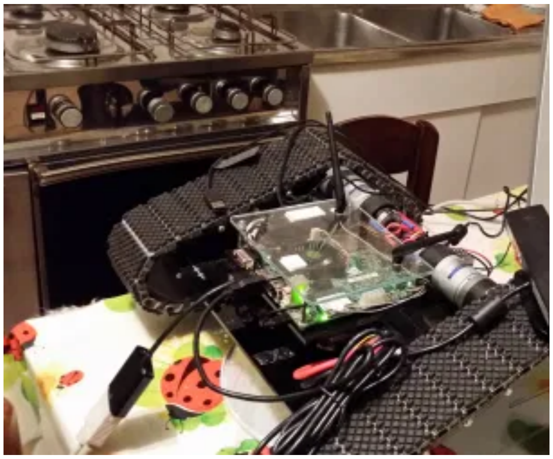
4. Test "NiViewer":

Shell

```
./NiViewer
```

If everything is ok you can see two images as in the following figures. The image on the left is the depth map, the image on the right is the RGB image. You can try many settings, press "?" for the full list of commands.





Generate redistributable package

1. Navigate to Packaging directory:

Shell

```
cd Packaging
```

2. Generate redistributable:

Shell

```
./ReleaseVersion.py Arm
```

3. At the end of the long process you will find the redistributable in the folder “ *~/devel/OpenNI2/Final*“

Installation

Install the compiled OpenNI2 library:

1. Move to “Package” folder:

Shell

```
cd ~/devel/OpenNI2/Packaging/OpenNI-Linux-Arm-2.2
```

2. Go to the new directory:

Shell

```
cd OpenNI-2.x.x-arm
```

3. Run the installation script to install the udev rule:

Shell

```
sudo sh install.sh
```

4. Copy “lib” files and “include” files to system path:

Shell

```
cd ~/devel/OpenNI2
sudo cp -r Include /usr/include/openni2
sudo cp -r Bin/Arm-Release/OpenNI2 /usr/lib/
sudo cp Bin/Arm-Release/libOpenNI2.* /usr/lib/
```

5. Update the library cache:

Shell

```
sudo ldconfig
```

6. Set permission to be able to use the sensor:

Shell

```
sudo usermod -a -G video ubuntu
```

7. Create a Package Config file:

    A. Create the file and edit:

    Shell

    ```
    sudo gedit /usr/lib/pkgconfig/libopenni2.pc
    ```

B. Fill the file with:

Vim

```
prefix=/usr
exec_prefix=${prefix}
libdir=${exec_prefix}/lib
includedir=${prefix}/include/openni2

Name: OpenNI2
Description: A general purpose driver for all OpenNI cameras.
Version: 2.2.0.0
Cflags: -I${includedir}
Libs: -L${libdir} -lOpenNI2 -L${libdir}/OpenNI2/Drivers -lDummyDevice -lOniFil
```

8. To make sure it is correctly found, run:

Shell

```
pkg-config --modversion libopenni2
```

You should get *2.2.0.0* as reply

Precompiled OpenNI2

You can avoid to compile OpenNI2 by yourself downloading the precompiled version from the following link:

Precompiled OpenNI2 for NVidia Jetson TK1

OpenNI-Linux-Arm-2.2.tar.bz2

Version: 2.2.0

2.7 MiB

433 Downloads

Details

1. Extract the archive:

Shell

```
tar -xvf OpenNI-Linux-Arm-2.x.x.tar.bz2
```

2. Follow the above installation procedure starting by “Step 2”

ROS

If you want to use OpenNI2 and Asus Xtion Pro Live with ROS framework you can follow the apposite guide.