Install Kinect Drivers

http://www.20papercups.net/programming/kinect-on-ubuntu-with-openni/

Kinect on Ubuntu with OpenNI – 20papercups

Kinect on Ubuntu with OpenNI

UPDATE October 2015: Verified working in Ubuntu 14.04 LTS and 15.04!
I've spent all this morning trying to talk to the Microsoft Kinect using OpenNI. As it turns out, the process is not exceptionally difficult, it's just there doesn't seem to be any up to date documentation on getting it all working. So, this post should fill the void. I describe how to get access to the Kinect working using Ubuntu 12.04 LTS, OpenNI 1.5.4, and NITE 1.5.2.

Please note that since writing this tutorial, we now have OpenNI and NITE 2.0, and PrimeSense have been bought by Apple. This tutorial does not work with versions 2 (though 1.5 works just fine), and there is talk of Apple stopping public access to NITE.

To talk to the Kinect, there are two basic parts: OpenNI itself, and a Sensor module that is actually responsible for communicating with the hardware. Then, if you need it, there is NITE, which is another module for OpenNI that does skeletal tracking, gestures, and stuff. Depending on how you plan on using the data from the Kinect, you may not need NITE at all.

Step 1: Prerequisites

We need to install a bunch of packages for all this to work. Thankfully, the readme file included with OpenNI lists all these. However, to make life easier, this is (as of writing) what you need to install, in addition to all the development packages you (hopefully) already have.

```
sudo apt-get install git build-essential python libusb-1.0-0-dev freeglut3-dev
openjdk-7-jdk
```

There are also some optional packages that you can install, depending on whether you want documentation, Mono bindings, etc. Note that on earlier versions the install failed if you didn't have doxygen installed, even though it is listed as optional.

```
sudo apt-get install doxygen graphviz mono-complete
```

Step 2: OpenNI 1.5.4

OpenNI is a framework for working with what they are calling *natural interaction* devices.Anyway, this is how it is installed:

Check out from Git

OpenNI is hosted on Github, so checking it out is simple:

```
git clone https://github.com/OpenNI/OpenNI.git
```

The first thing we will do is checkout the Unstable 1.5.4 tag. If you don't do this, then the SensorKinect library won't compile in Step 3. From there, change into the Platform/Linux-x86/CreateRedist directory, and run the RedistMaker script. Note that even though the directory is named x86, this same directory builds 64 bit versions just fine. So, don't fret if you're on 64bit Linux.

```
cd OpenNI
git checkout Unstable-1.5.4.0
cd Platform/Linux/CreateRedist
chmod +x RedistMaker
./RedistMaker
```

The RedistMaker script will compile everything for you. You then need to change into the Redist directory and run the install script to install the software on your system.

```
cd ../Redist/OpenNI-Bin-Dev-Linux-[xxx]  (where [xxx] is your architecture and this
particular OpenNI release)
sudo ./install.sh
```

Step 3: Kinect Sensor Module

OpenNI doesn't actually provide anything for talking to the hardware, it is more just a framework for working with different sensors and devices. You need to install a Sensor module for actually doing the hardware interfacing. Think of an OpenNI sensor module as a device driver for the hardware. You'll also note on the OpenNI website that they have a Sensor module that you can download. Don't do this though, because that sensor module doesn't talk to the Kinect. I love how well documented all this is, don't you?

The sensor module you want is also on GitHub, but from a different user. So, we can check out the code. We also need to get the kinect branch, not master.

```
git clone https://github.com/avin2/SensorKinect
cd SensorKinect
```

The install process for the sensor is pretty much the same as for OpenNI itself:

```
cd Platform/Linux/CreateRedist
chmod +x RedistMaker
./RedistMaker
cd ../Redist/Sensor-Bin-Linux-[xxx] (where [xxx] is your architecture and this
particular OpenNI release)
chmod +x install.sh
sudo ./install.sh
```

On Ubuntu, regular users are only given read permission to unknown USB devices. The install script puts in some udev rules to fix this, but if you find that none of the samples work unless you run them as root, try unplugging and plugging the Kinect back in again, to make the new rules apply.

Step 4: Test the OpenNI Samples

At this point, you have enough installed to get data from the Kinect. The easiest way to verify this is to run one of the OpenNI samples.

```
cd OpenNI/Platform/Linux-x86/Bin/Release
./Sample-NiSimpleViewer
```

You should see a yellow-black depth image. At this point, you're left with (optionally) installing the higher level NITE module.

Step 5: Install NITE 1.5 (optional)

Firstly, you need to obtain NITE 1.5.2. Go to the following link and download NITE 1.5.2 for your platform..

Extract the archive, and run the installer:

```
sudo ./install.sh
```

At some point, you may be asked for a license key. A working license key can be found just about anywhere on the Internet. I don't think PrimeSense care, or maybe this is a non-commercial license or something. But whatever, just copy that license into the console, including the equals sign at the end, and NITE will install just fine.

Conclusion

After following these steps, you will be able to write programs that use the Microsoft Kinect through OpenNI and NITE middleware. I hope this helps someone, because I spent a lot of time screwing around

this morning trying to get it all to work. Like I said, the process is pretty straight forward, it just hasn't been written down in one place (or I suck at google).

Published by

Michael Marner

My name is Michael, and this is my website. I am currently working as a Research Fellow at the University of South Australia's Wearable Computer Lab. I am also the chair of the board and an announcer at Adelaide based community radio station Three D Radio 93.7FM. [View all posts by Michael Marner](#)

Posted on