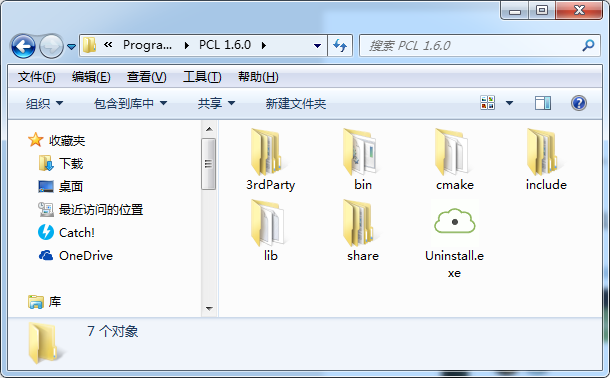
# 使用PCL库和XTION获取深度图像

## 下载并安装PCL库

网址：<http://pointclouds.org/downloads/windows.html>



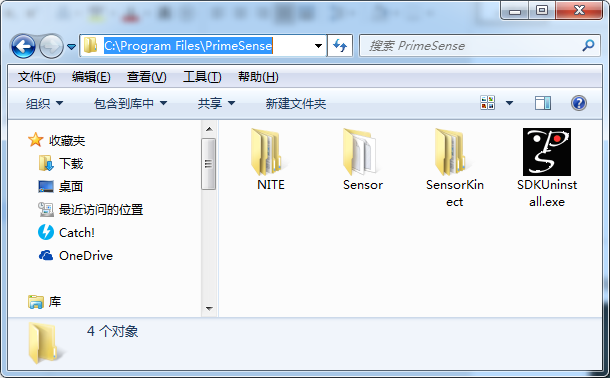
安装后：



这里有include目录，在3rdParty下的各个文件夹下也有include目录，把这些目录全设置到Visual Studio 2010的工程目录中，同样，lib目录设置成库目录，把bin目录加入到环境变量中。

## 安装XTion驱动

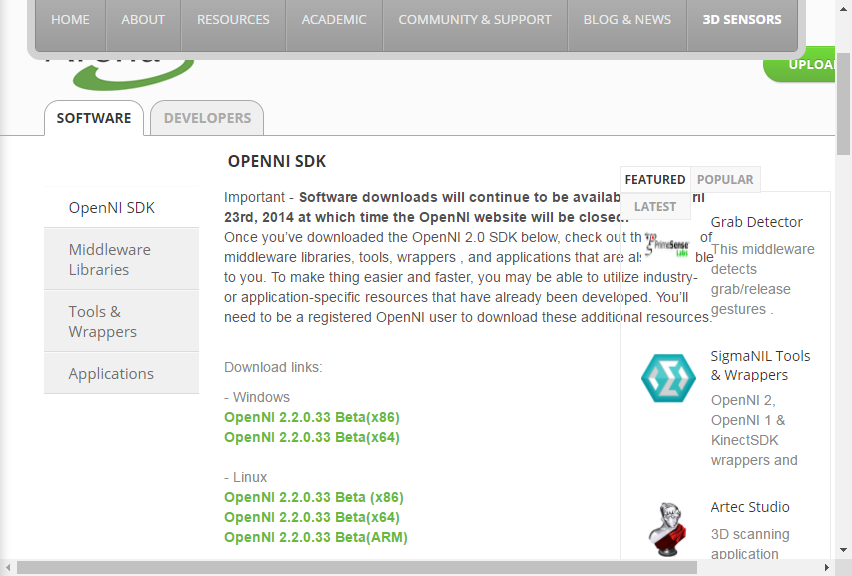
下载地址：<https://dlsvr04.asus.com/pub/ASUS/MM/Xtion_Pro/V1164_1202.zip>，安装到了如下位置：



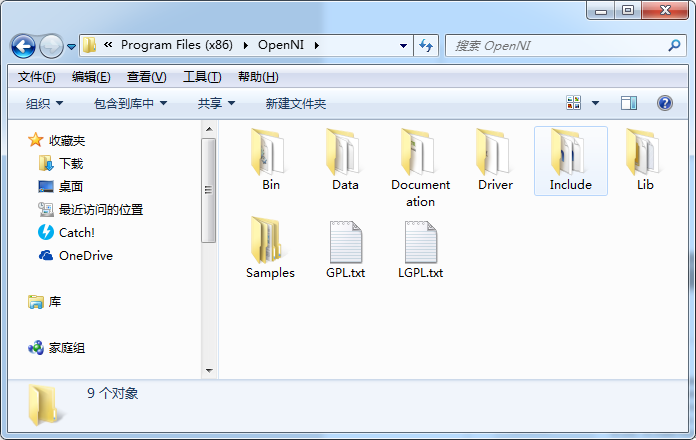
把C:\Program Files\PrimeSense\Sensor\Bin64添加到环境变量。

## 安装OpenNI

我没有使用pcl自带的OpenNI，而是自己下载的OpenNI，下载地址：<http://www.openni.ru/openni-sdk/index.html>



安装后：



同样，这里有include目录，lib目录和bin目录，与上面处理方法相同。

## 配置汇总

包含目录：

|  |
| --- |
| D:\Program Files\PCL 1.6.0\include\pcl-1.6;  D:\Program Files\PCL 1.6.0\3rdParty\Eigen\include;  C:\Program Files (x86)\OpenNI\Include;  D:\Program Files\PCL 1.6.0\3rdParty\Boost\include;  D:\Program Files\PCL 1.6.0\3rdParty\VTK\include;  D:\Program Files\PCL 1.6.0\3rdParty\VTK\include\vtk-5.8; |

库目录：

|  |
| --- |
| D:\Program Files\PCL 1.6.0\lib;  C:\Program Files (x86)\OpenNI\Lib;  D:\Program Files\PCL 1.6.0\3rdParty\Boost\lib;  D:\Program Files\PCL 1.6.0\3rdParty\VTK\lib\vtk-5.8; |

依赖项：

|  |
| --- |
| pcl\_apps\_debug.lib  pcl\_common\_debug.lib  pcl\_features\_debug.lib  pcl\_filters\_debug.lib  pcl\_io\_debug.lib  pcl\_io\_ply\_debug.lib  pcl\_kdtree\_debug.lib  pcl\_keypoints\_debug.lib  pcl\_octree\_debug.lib  pcl\_registration\_debug.lib  pcl\_sample\_consensus\_debug.lib  pcl\_search\_debug.lib  pcl\_segmentation\_debug.lib  pcl\_surface\_debug.lib  pcl\_tracking\_debug.lib  pcl\_visualization\_debug.lib  NiSampleExtensionModule.lib  NiSampleModule.lib  OpenNI.jni.lib  openNI.lib  boost\_chrono-vc100-mt-1\_49.lib  boost\_chrono-vc100-mt-gd-1\_49.lib  boost\_date\_time-vc100-mt-1\_47.lib  boost\_date\_time-vc100-mt-1\_49.lib  boost\_date\_time-vc100-mt-gd-1\_47.lib  boost\_date\_time-vc100-mt-gd-1\_49.lib  boost\_filesystem-vc100-mt-1\_47.lib  boost\_filesystem-vc100-mt-1\_49.lib  boost\_filesystem-vc100-mt-gd-1\_47.lib  boost\_filesystem-vc100-mt-gd-1\_49.lib  boost\_graph-vc100-mt-1\_49.lib  boost\_graph-vc100-mt-gd-1\_49.lib  boost\_graph\_parallel-vc100-mt-1\_49.lib  boost\_graph\_parallel-vc100-mt-gd-1\_49.lib  boost\_iostreams-vc100-mt-1\_47.lib  boost\_iostreams-vc100-mt-1\_49.lib  boost\_iostreams-vc100-mt-gd-1\_47.lib  boost\_iostreams-vc100-mt-gd-1\_49.lib  boost\_locale-vc100-mt-1\_49.lib  boost\_locale-vc100-mt-gd-1\_49.lib  boost\_math\_c99-vc100-mt-1\_49.lib  boost\_math\_c99-vc100-mt-gd-1\_49.lib  boost\_math\_c99f-vc100-mt-1\_49.lib  boost\_math\_c99f-vc100-mt-gd-1\_49.lib  boost\_math\_tr1-vc100-mt-1\_49.lib  boost\_math\_tr1-vc100-mt-gd-1\_49.lib  boost\_math\_tr1f-vc100-mt-1\_49.lib  boost\_math\_tr1f-vc100-mt-gd-1\_49.lib  boost\_mpi-vc100-mt-1\_49.lib  boost\_mpi-vc100-mt-gd-1\_49.lib  boost\_prg\_exec\_monitor-vc100-mt-1\_49.lib  boost\_prg\_exec\_monitor-vc100-mt-gd-1\_49.lib  boost\_program\_options-vc100-mt-1\_49.lib  boost\_program\_options-vc100-mt-gd-1\_49.lib  boost\_random-vc100-mt-1\_49.lib  boost\_random-vc100-mt-gd-1\_49.lib  boost\_regex-vc100-mt-1\_49.lib  boost\_regex-vc100-mt-gd-1\_49.lib  boost\_serialization-vc100-mt-1\_49.lib  boost\_serialization-vc100-mt-gd-1\_49.lib  boost\_signals-vc100-mt-1\_49.lib  boost\_signals-vc100-mt-gd-1\_49.lib  boost\_system-vc100-mt-1\_47.lib  boost\_system-vc100-mt-1\_49.lib  boost\_system-vc100-mt-gd-1\_47.lib  boost\_system-vc100-mt-gd-1\_49.lib  boost\_thread-vc100-mt-1\_47.lib  boost\_thread-vc100-mt-1\_49.lib  boost\_thread-vc100-mt-gd-1\_47.lib  boost\_thread-vc100-mt-gd-1\_49.lib  boost\_timer-vc100-mt-1\_49.lib  boost\_timer-vc100-mt-gd-1\_49.lib  boost\_unit\_test\_framework-vc100-mt-1\_49.lib  boost\_unit\_test\_framework-vc100-mt-gd-1\_49.lib  boost\_wave-vc100-mt-1\_49.lib  boost\_wave-vc100-mt-gd-1\_49.lib  boost\_wserialization-vc100-mt-1\_49.lib  boost\_wserialization-vc100-mt-gd-1\_49.lib  libboost\_chrono-vc100-mt-1\_49.lib  libboost\_chrono-vc100-mt-gd-1\_49.lib  libboost\_date\_time-vc100-mt-1\_47.lib  libboost\_date\_time-vc100-mt-1\_49.lib  libboost\_date\_time-vc100-mt-gd-1\_47.lib  libboost\_date\_time-vc100-mt-gd-1\_49.lib  libboost\_filesystem-vc100-mt-1\_47.lib  libboost\_filesystem-vc100-mt-1\_49.lib  libboost\_filesystem-vc100-mt-gd-1\_47.lib  libboost\_filesystem-vc100-mt-gd-1\_49.lib  libboost\_graph\_parallel-vc100-mt-1\_49.lib  libboost\_graph\_parallel-vc100-mt-gd-1\_49.lib  libboost\_iostreams-vc100-mt-1\_47.lib  libboost\_iostreams-vc100-mt-1\_49.lib  libboost\_iostreams-vc100-mt-gd-1\_47.lib  libboost\_iostreams-vc100-mt-gd-1\_49.lib  libboost\_locale-vc100-mt-1\_49.lib  libboost\_locale-vc100-mt-gd-1\_49.lib  libboost\_math\_c99-vc100-mt-1\_49.lib  libboost\_math\_c99-vc100-mt-gd-1\_49.lib  libboost\_math\_c99f-vc100-mt-1\_49.lib  libboost\_math\_c99f-vc100-mt-gd-1\_49.lib  libboost\_math\_tr1-vc100-mt-1\_49.lib  libboost\_math\_tr1-vc100-mt-gd-1\_49.lib  libboost\_math\_tr1f-vc100-mt-1\_49.lib  libboost\_math\_tr1f-vc100-mt-gd-1\_49.lib  libboost\_mpi-vc100-mt-1\_49.lib  libboost\_mpi-vc100-mt-gd-1\_49.lib  libboost\_prg\_exec\_monitor-vc100-mt-1\_49.lib  libboost\_prg\_exec\_monitor-vc100-mt-gd-1\_49.lib  libboost\_program\_options-vc100-mt-1\_49.lib  libboost\_program\_options-vc100-mt-gd-1\_49.lib  libboost\_random-vc100-mt-1\_49.lib  libboost\_random-vc100-mt-gd-1\_49.lib  libboost\_regex-vc100-mt-1\_49.lib  libboost\_regex-vc100-mt-gd-1\_49.lib  libboost\_serialization-vc100-mt-1\_49.lib  libboost\_serialization-vc100-mt-gd-1\_49.lib  libboost\_signals-vc100-mt-1\_49.lib  libboost\_signals-vc100-mt-gd-1\_49.lib  libboost\_system-vc100-mt-1\_47.lib  libboost\_system-vc100-mt-1\_49.lib  libboost\_system-vc100-mt-gd-1\_47.lib  libboost\_system-vc100-mt-gd-1\_49.lib  libboost\_test\_exec\_monitor-vc100-mt-1\_49.lib  libboost\_test\_exec\_monitor-vc100-mt-gd-1\_49.lib  libboost\_thread-vc100-mt-1\_47.lib  libboost\_thread-vc100-mt-1\_49.lib  libboost\_thread-vc100-mt-gd-1\_47.lib  libboost\_thread-vc100-mt-gd-1\_49.lib  libboost\_timer-vc100-mt-1\_49.lib  libboost\_timer-vc100-mt-gd-1\_49.lib  libboost\_unit\_test\_framework-vc100-mt-1\_49.lib  libboost\_unit\_test\_framework-vc100-mt-gd-1\_49.lib  libboost\_wave-vc100-mt-1\_49.lib  libboost\_wave-vc100-mt-gd-1\_49.lib  libboost\_wserialization-vc100-mt-1\_49.lib  libboost\_wserialization-vc100-mt-gd-1\_49.lib  MapReduceMPI-gd.lib  MapReduceMPI.lib  mpistubs-gd.lib  mpistubs.lib  QVTK-gd.lib  QVTK.lib  vtkalglib-gd.lib  vtkalglib.lib  vtkCharts-gd.lib  vtkCharts.lib  vtkCommon-gd.lib  vtkCommon.lib  vtkDICOMParser-gd.lib  vtkDICOMParser.lib  vtkexoIIc-gd.lib  vtkexoIIc.lib  vtkexpat-gd.lib  vtkexpat.lib  vtkFiltering-gd.lib  vtkFiltering.lib  vtkfreetype-gd.lib  vtkfreetype.lib  vtkftgl-gd.lib  vtkftgl.lib  vtkGenericFiltering-gd.lib  vtkGenericFiltering.lib  vtkGeovis-gd.lib  vtkGeovis.lib  vtkGraphics-gd.lib  vtkGraphics.lib  vtkhdf5-gd.lib  vtkhdf5.lib  vtkHybrid-gd.lib  vtkHybrid.lib  vtkImaging-gd.lib  vtkImaging.lib  vtkInfovis-gd.lib  vtkInfovis.lib  vtkIO-gd.lib  vtkIO.lib  vtkjpeg-gd.lib  vtkjpeg.lib  vtklibxml2-gd.lib  vtklibxml2.lib  vtkmetaio-gd.lib  vtkmetaio.lib  vtkNetCDF-gd.lib  vtkNetCDF.lib  vtkNetCDF\_cxx-gd.lib  vtkNetCDF\_cxx.lib  vtkpng-gd.lib  vtkpng.lib  vtkproj4-gd.lib  vtkproj4.lib  vtkRendering-gd.lib  vtkRendering.lib  vtksqlite-gd.lib  vtksqlite.lib  vtksys-gd.lib  vtksys.lib  vtktiff-gd.lib  vtktiff.lib  vtkverdict-gd.lib  vtkverdict.lib  vtkViews-gd.lib  vtkViews.lib  vtkVolumeRendering-gd.lib  vtkVolumeRendering.lib  vtkWidgets-gd.lib  vtkWidgets.lib  vtkzlib-gd.lib  vtkzlib.lib |

## 5测试

|  |
| --- |
| #include "stdafx.h"  #include <pcl/io/openni\_grabber.h>  #include <pcl/visualization/cloud\_viewer.h>  #include <pcl/io/pcd\_io.h>  #include <iostream>  #include <string>  #include <strstream>  #include <pcl/common/time.h>  #include <pcl/console/parse.h>    class SimpleOpenNIViewer  {  public:  SimpleOpenNIViewer () : viewer ("PCL OpenNI Viewer") {}  void cloud\_cb\_ (const pcl::PointCloud<pcl::PointXYZRGB>::ConstPtr &cloud)  {  if (!viewer.wasStopped())  {  viewer.showCloud (cloud);  std::stringstream ss;  ss << std::setprecision (12) << pcl::getTime () \* 100 << ".pcd";  pcl::PCDWriter w;  w.writeBinaryCompressed (ss.str (), \*cloud);  }  }    void run ()  {  pcl::Grabber\* interface = new pcl::OpenNIGrabber();    boost::function<void (const pcl::PointCloud<pcl::PointXYZRGB>::ConstPtr&)> f =  boost::bind (&SimpleOpenNIViewer::cloud\_cb\_, this, \_1);  interface->registerCallback (f);  interface->start ();  while (!viewer.wasStopped())  {  boost::this\_thread::sleep (boost::posix\_time::seconds (1));  }  interface->stop ();  }  pcl::visualization::CloudViewer viewer;  };    int main ()  {  SimpleOpenNIViewer v;  v.run ();  return 0;  } |

