

- **Visual Studio**


Install Visual Studio 2010 for C++ development.

In case VS 2010 is installed, but no C++ environment exist, please download and install:

- 64 bit: <http://www.microsoft.com/en-us/download/details.aspx?id=14632>
- 32 bit: <http://www.microsoft.com/en-us/download/details.aspx?id=5555>

**Microsoft Visual C++ 2010 Redistributable Package (x86)**


If your download does not start after 30 seconds, [Click here](#)

 [Install Instructions](#)

---

**Microsoft Visual C++ 2010 Redistributable Package (x64)**

If your download does not start after 30 seconds, [Click here](#)

 [Install Instructions](#)

---

- **PCL**

Before compiling PCL, install the next **dependencies**:

- FLANN 1.7.1
- Eigen 3.0.3
- Boost 1.49
- VTK 5.8.0 with QT support
- QHULL 2011.1
- QT 4.8.0
- OpenNI 1.3.2
- OpenNI SENSOR 5.0.3
- CUDA 5.5 (Should be installed by NVIDIA)

These can be found on the project's website, or downloaded from here:

<http://sourceforge.net/projects/pointclouds/files/dependencies/>

(If prompted, let the installations add the directories to PATH environment variable).

Download **PCL 1.6.0** source code:

This can be found on our project's website.

After downloading it, make a development directory and put pcl directory in it.

- **CMAKE:**

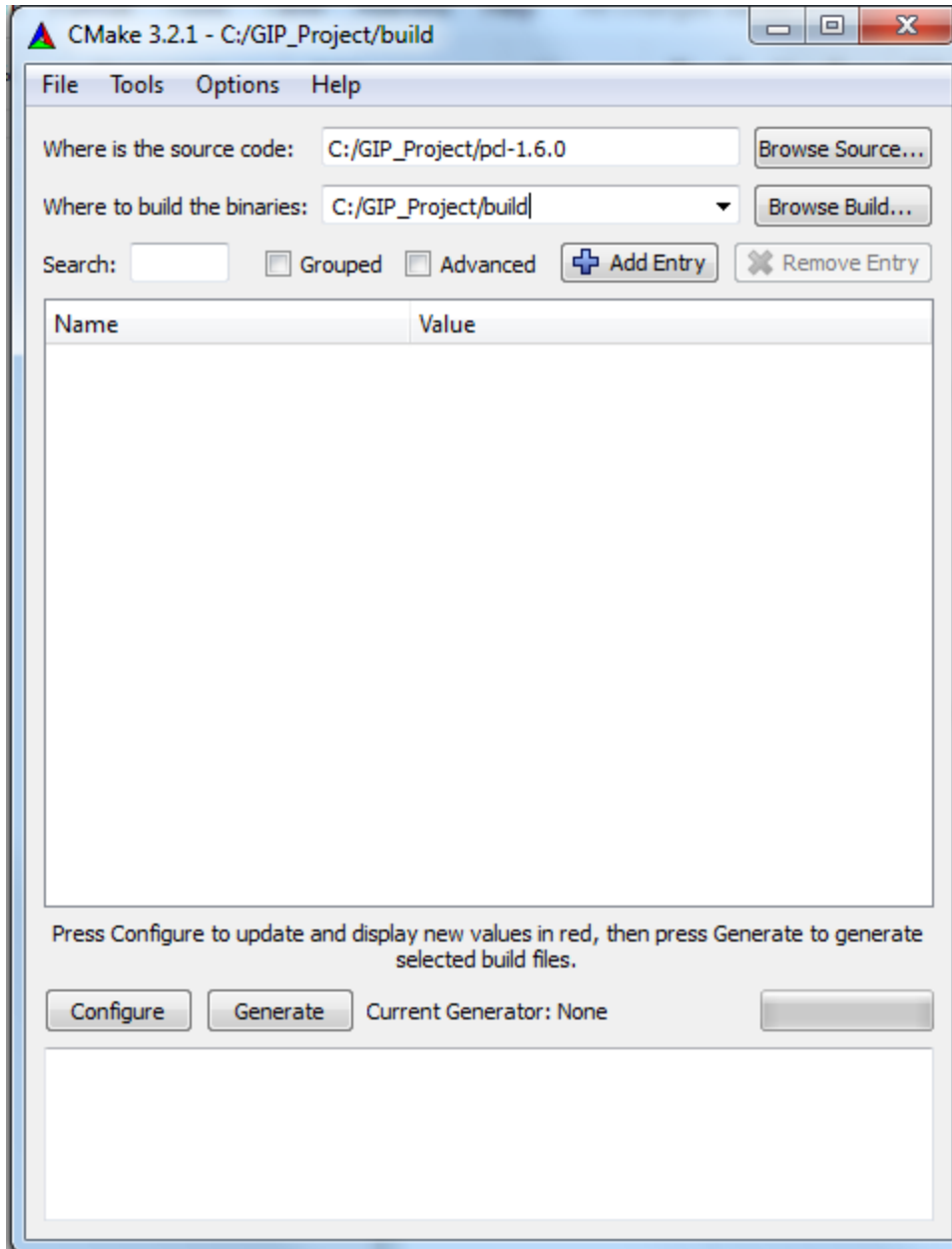
Install Cmake version 3.0.0 (can be found in the website) or:

<http://www.cmake.org/download/>

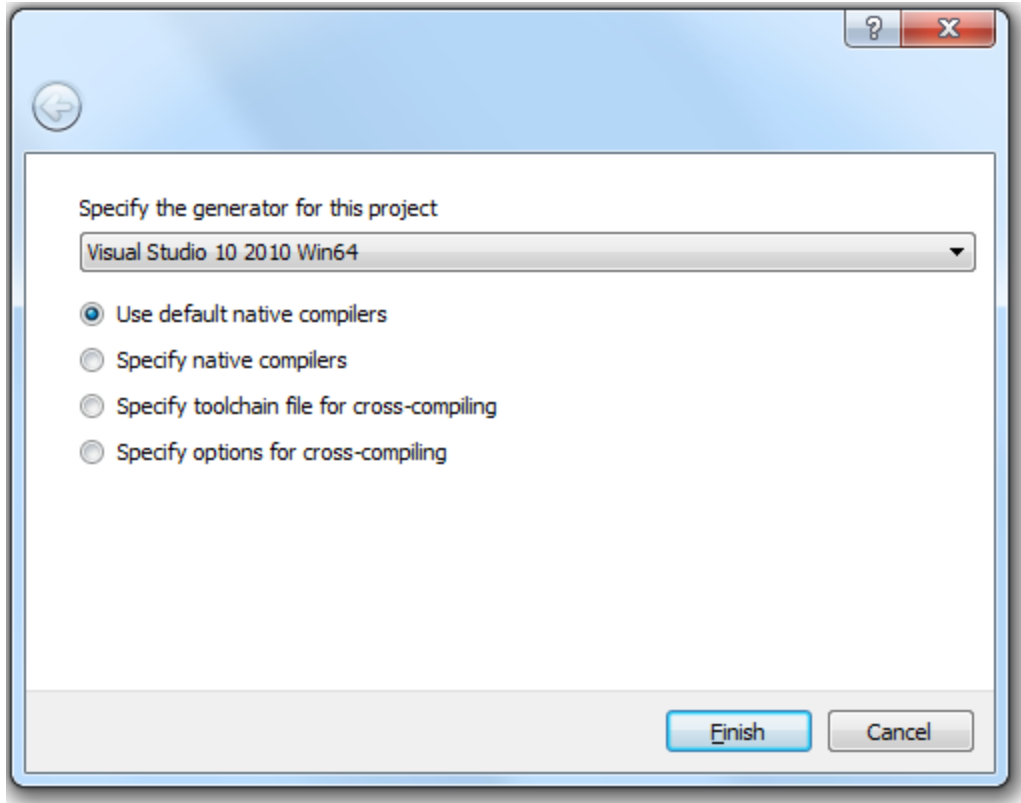
**IMPORTANT:** Before running Cmake, make sure .NET Framework version on your computer is 4.0. If higher, please downgrade:

<http://www.microsoft.com/en-us/download/confirmation.aspx?id=17851>

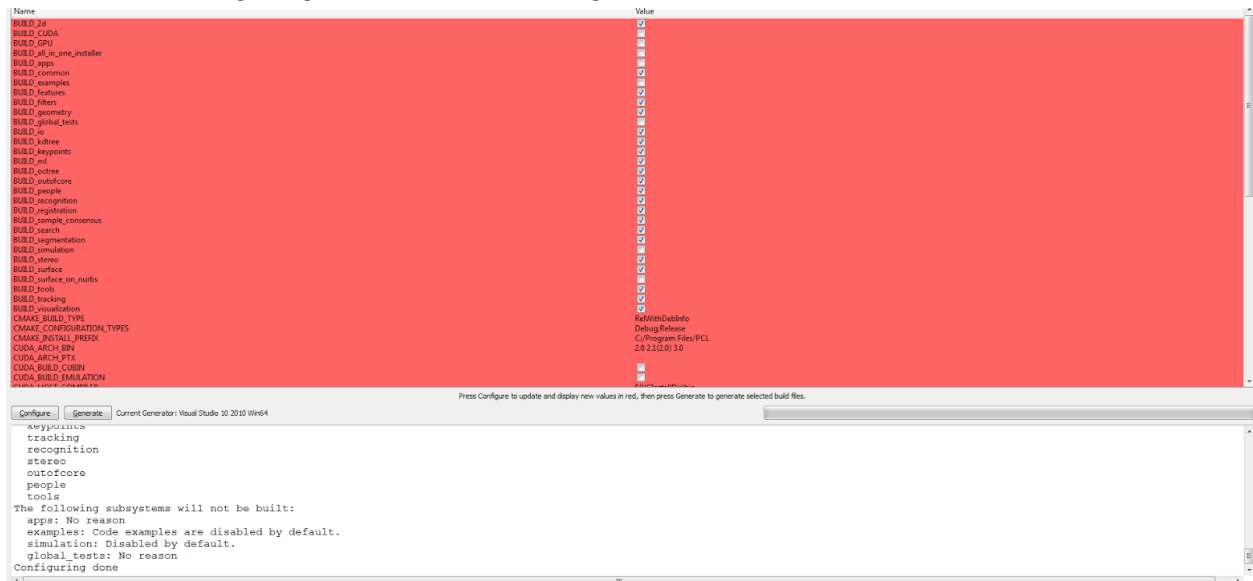
- Configuring Cmake:
- Run CMAKE:  
PATH\_TO\_CMAKE\CMake\bin\cmake-gui.exe  
You should get the next screen:



- Browse directories:  
Source code is the pcl directory you downloaded.  
Build the binaries in a new directory, to be located at the development directory.
- Press 'Configure'
- Choose your compiler:



- After configuring is done, the following screen will show:



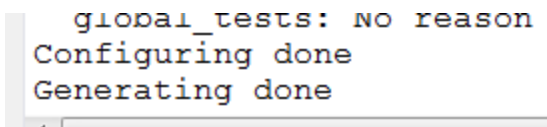
- Now it is time to inform cmake with the dependencies location.  
The dependencies installation should already updated PATH variable so the corresponding directories should already be set. In case they are not, add them manually from the installation directories. Make sure the following is found:  
CUDA\_SDK\_DIR: C:\Program Files\NVIDIA GPU Computing Toolkit\CUDA\v5.5  
QT\_QMAKE\_EXECUTABLE: C:\Qt\4.8.0\bin\qmake.exe

Each dependency mentioned above with 'NOT\_FOUND' as value, should be located. The following can stay not found: libusb, pcap, fzapi, pxcapi, qt5, tide.

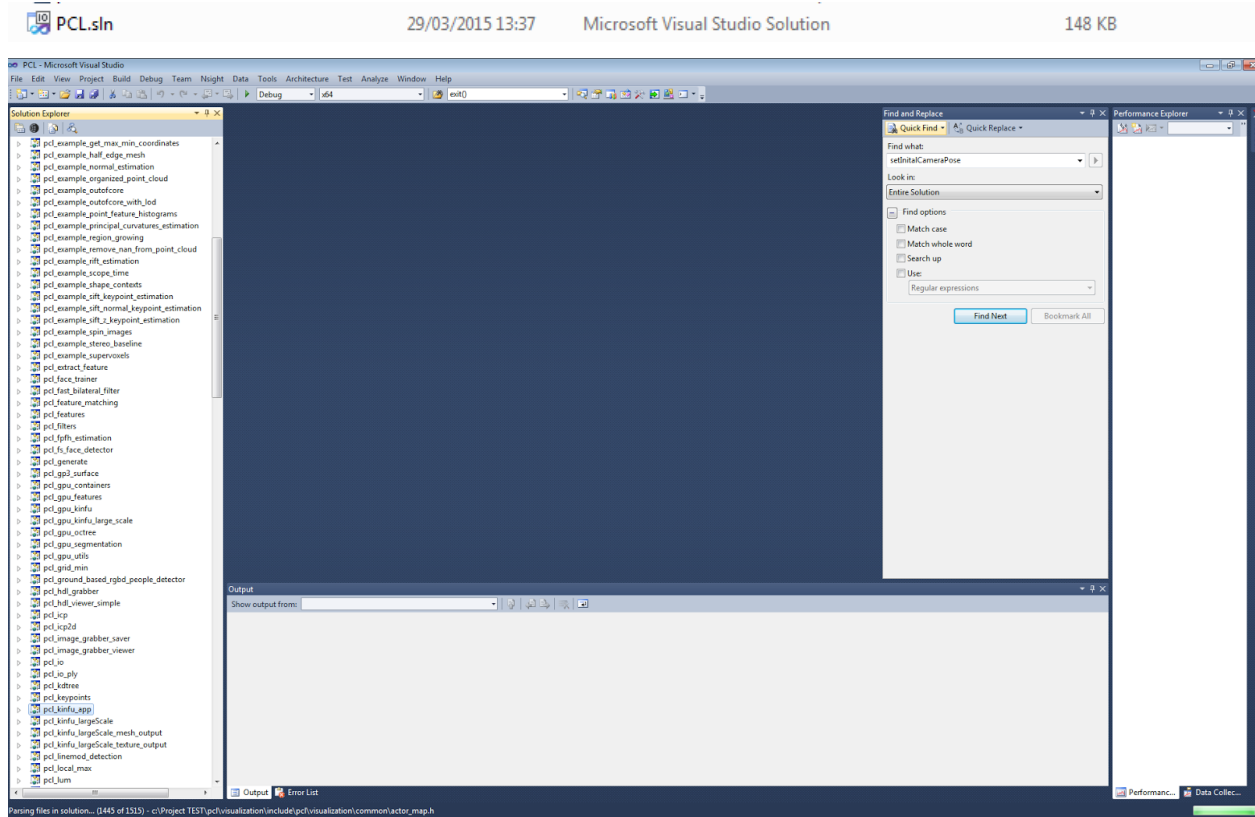
Mark the checkbox as follows (You might need to press configure again to show all of the options).



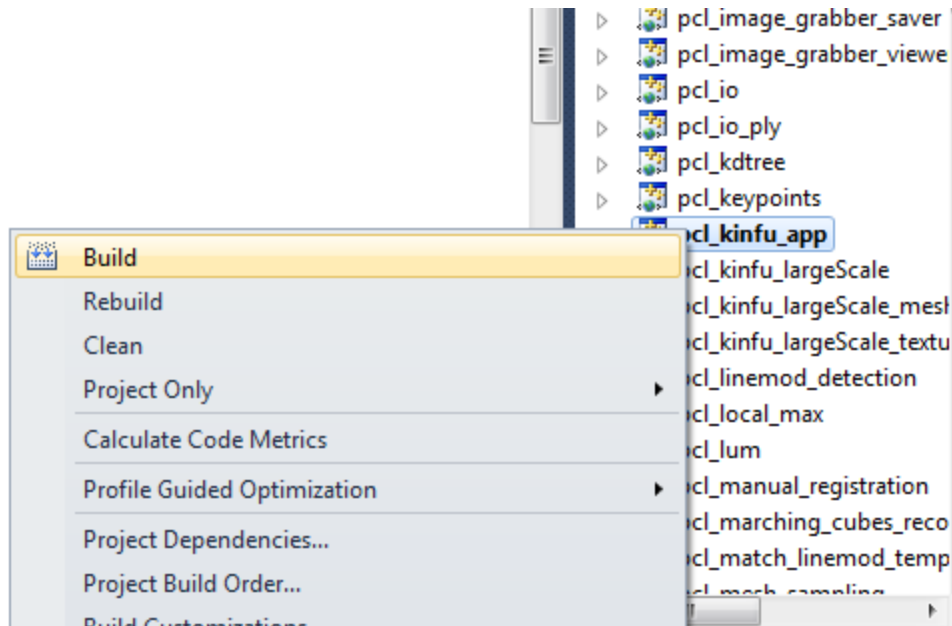
- After all set, press 'Generate', you should get:



- Go to the 'build' directory you just created with cmake, and run 'PCL.sln'. This will open the generated solution in VS 2010:



- In the solution explorer, locate 'pcl\_kinfu\_app'. right click → Build:



- **Running Kinfu:**

After compiling kinfu, try to run it. If it succeeded, you can progress installing our changes:

- Replace 'pcl\gpu\kinfu\tools\kinfu\_app.cpp' & 'pcl\gpu\kinfu\include\pcl\gpu\kinfu\kinfu.h' with our version.
  - Add '3DfaceDetection\_GIP.h' and 'FaceDetection.cpp' to the project.
- Add project dependencies:
    - octree
    - kdtree
    - surface

- **Install OpenCV:**

Install version 2.4.9.

- Add to 'Additional include directories': `$(OPENCV_DIR)\..\include`
- Add to 'Additional library list': `$(OPENCV_DIR)\lib`
- Add to 'Linker → Input → Additional Dependencies':
  - opencv\_calib3d249d.lib
  - opencv\_contrib249d.lib
  - opencv\_core249d.lib
  - opencv\_features2d249d.lib
  - opencv\_flann249d.lib
  - opencv\_gpu249d.lib
  - opencv\_highgui249d.lib
  - opencv\_imgproc249d.lib
  - opencv\_legacy249d.lib
  - opencv\_ml249d.lib
  - opencv\_nonfree249d.lib
  - opencv\_objdetect249d.lib
  - opencv\_ocl249d.lib
  - opencv\_photo249d.lib
  - opencv\_stitching249d.lib
  - opencv\_superres249d.lib
  - opencv\_ts249d.lib
  - opencv\_video249d.lib
  - opencv\_videostab249d.lib

Note: the 'd' is for debug configuration. For release, delete it.

(Note: there is a collapse between Cmake and OpenCV so the installation should be manually.)

- Run kinfu again, this time with flags: '-r -ic'.

Our application will start.

- Enjoy!