# Compiling HDR Fusion

(tested on Windows 10/8.1/8)

19/03/2016

## Install Dependency

1. install windows 10/8/8.1/7
   1. create a bootable usb
   2. format disk and install the ubuntu
2. install [Visual Studio 2013 Professsional with update 5rc](https://e5.onthehub.com/WebStore/ShoppingCart.aspx?ws=0fbbbbd6-0834-e211-aed3-f04da23e67f6&vsro=8)
3. install CUDA 7.5 with NVidia driver

a. download CUDA 7.5 runfile installer package from [here](https://developer.nvidia.com/cuda-downloads).

1. [TortoiseGit](https://code.google.com/p/tortoisegit/wiki/Download)
   1. [Git for Window](https://msysgit.github.io/)
   2. [Install TortoiseGit x64](https://code.google.com/p/tortoisegit/wiki/Download)
2. CMake

download cmake from [here](http://www.cmake.org/download/) (windows 32bit)

1. install GLEW (allow commercial use)
   1. Goto [Glew Website](http://glew.sourceforge.net/)
   2. Download the up to date binaries.
   3. Extract the zip file into the Dependency folder. For example “C:/all\_libs/glew/”
2. install Boost (allow commercial use)
   1. Download from [here](http://www.boost.org/users/history/version_1_58_0.html)
   2. $bootstrap.bat
3. $bjam

8. OpenNI 2 (license fee paid with the hardware)

1. download from [git](https://github.com/OpenNI/OpenNI2)
2. Follow the instructions in [README](https://github.com/OpenNI/OpenNI2/blob/master/README)
3. Eigen (allow commercial use)
   1. $ hg clone<https://bitbucket.org/eigen/eigen/>
   2. $ cd eigen
   3. $ mkdir build
   4. $ cd build
   5. $ cmake ..
   6. $ cmake-gui ..

remove BUILD\_TESTING

remove EIGEN\_SPLIT\_LARGE\_TESTS

specify CMAKE\_INSTALL\_PREFIX

* 1. Build project “Install” in VS2013 to install it into your specified folder

1. QT
   1. Download run file from [Qt website](http://www.qt.io/download-open-source/) and install it
   2. select “with OpenGL”
2. QGLViewer
   1. Download from [github](https://github.com/GillesDebunne/libQGLViewer)
   2. follow the instructions in [INSTALL](https://github.com/GillesDebunne/libQGLViewer/blob/master/INSTALL)
3. Piccante
   1. Download from [github](https://github.com/cnr-isti-vclab/piccante)
4. OpenCV
   1. Download from [github](https://github.com/Itseez/opencv) and [opencv\_contrib](https://github.com/Itseez/opencv_contrib)
   2. follow online instruction
   3. $ cd opencv
   4. $ mkdir build
   5. $ cd build
   6. $ cmake ..
   7. $ cmake-gui ..
   8. set CUDA\_ARCH\_BIN according the computational capability of the graphic card
   9. select OPENMP and OPENGL in WITH
   10. click Generate
   11. open the OpenCV.sln with VS2013
   12. compile and install the project
5. [Define environmental variables](https://docs.google.com/document/d/1r0OXgRnh1gmFXc_xkNgVFyqdBnQLCaIRzRFv2_Qe96c/edit?usp=sharing)

## 

## Compile the RGBD library and HDRFusionMain project

1. navigate to /hdr\_fusion/ folder
   1. $ cd hdr\_fusion/
2. edit line 27 in "hdr\_fusion\CMakeLists.txt" for various graphic cards according to the table
   1. set(CUDA\_NVCC\_FLAGS; -arch=sm\_30) #-arch=sm\_35) #-arch=**sm\_21**)

|  |  |
| --- | --- |
| NVidia Quadro 600 | -arch=sm\_21 |
| NVidia GTX680m /GTX680 | -arch=sm\_30 |
| NVidia GTX TITAN BLack | -arch=sm\_35 |

1. create build folder to hold all binary files
   1. $ mkdir build
   2. $ cd build
2. use cmake to create Makefile for compiling
   1. $ cmake ..
   2. $ cmake-gui ..
3. click “configure” button
4. click “generate” button
5. compile the library and executable files by open the “.sln” file
6. run the executable HDRFusionMain

BOOST\_INC

C:\all\_libs\boost\_1\_58\_0\_ver3

BOOST\_LIB

C:\all\_libs\boost\_1\_58\_0\_ver3\build\vc12\_x64\lib

EIGEN\_INC

C:\all\_libs\Eigen\include\eigen3

SOPHUS\_INC=C:\all\_libs\Sophus\include\sophus

GLEW\_INC=C:\all\_libs\glew-2.0.0\include

GLEW\_LIB=C:\all\_libs\glew-2.0.0\build\vs12\_ver2\lib

OPENCV\_NEW=C:\all\_libs\OpenCV

OPENNI2\_INC=C:\all\_libs\OpenNI2-2.2-beta2\Include

OPENNI2\_LIB=C:\all\_libs\OpenNI2-2.2-beta2\Bin\x64-Release

QGLVIEWER\_INC=C:\all\_libs\libQGLViewer-2.6.4

QGLVIEWER\_LIB=C:\all\_libs\libQGLViewer-2.6.4