Creating an OpenCV-OpenCL C++ Project with Visual Studio in Windows

Written By: Rich Shoff & Biruh Tesfaye

Course: Real-Time Computer Vision

Date Written: 3/16/2014

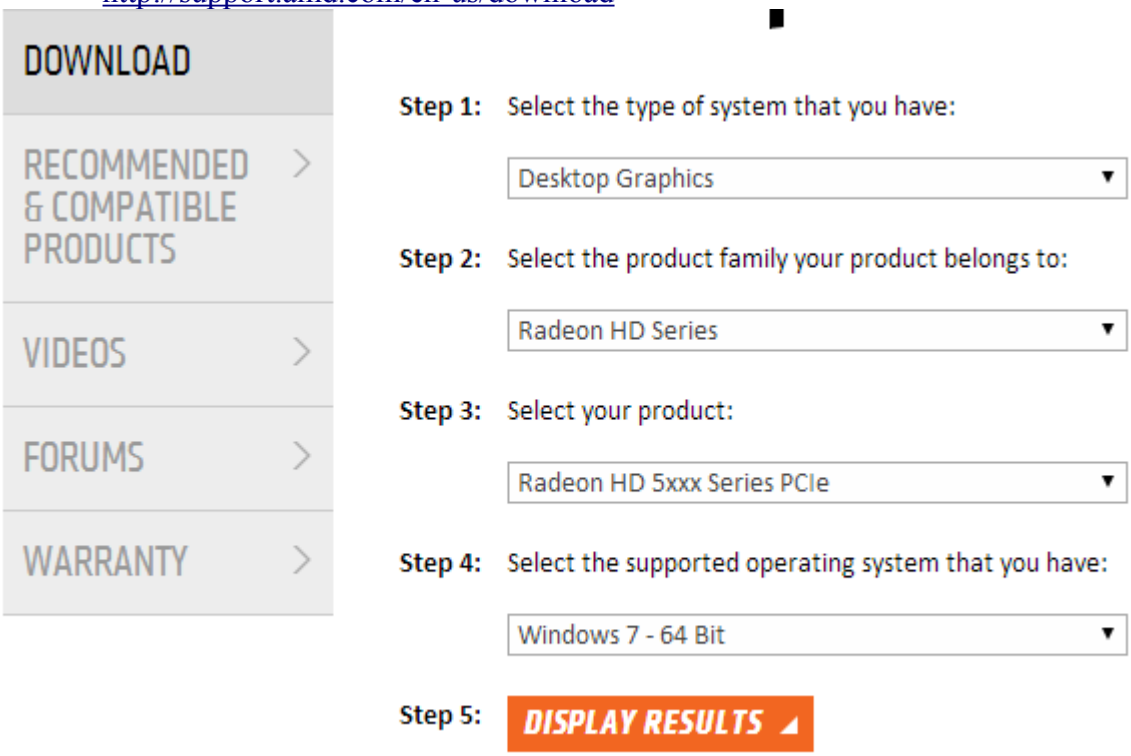
Driver and Application Installation

This walk though uses an AMD Radeon HD 5700 Graphics Card as an example.

1. Ensure that your Graphics card is supported by OpenCL. This info can be obtained from the Graphics Card's vendor's website, or possibly Wikipedia:
   http://en.wikipedia.org/wiki/OpenCL

2. Download and install Graphics card driver and SDK.

   a. Install latest AMD Catalyst Display driver.
      http://support.amd.com/en-us/download



b. Download and install the latest AMD APP OpenCL Software Development kit (SDK): http://developer.amd.com/tools-and-sdks/heterogeneous-computing/amd-accelerated-parallel-processing-app-sdk/downloads/

c. Download and install the latest AMD APP OpenCL Math Libraries (AMDBLAS & AMDFFT):
http://developer.amd.com/tools-and-sdks/heterogeneous-computing/amd-accelerated-parallel-processing-math-libraries/

3. Download and install **Visual Studio Express 2012** (or whatever you prefer) using the following tutorial:
http://www.microsoft.com/en-us/download/details.aspx?id=34673

4. Download and extract **OpenCV 2.4.8** (or whichever version you prefer) to the C:\ drive:
http://sourceforge.net/projects/opencvlibrary/files/opencv-win/

5. Download and install **Cmake** from the following location:
http://www.cmake.org/cmake/resources/software.html

Create Visual Studio Project from Source

6. Create a directory called C:\opencv**-cl-built**, or whatever you prefer.

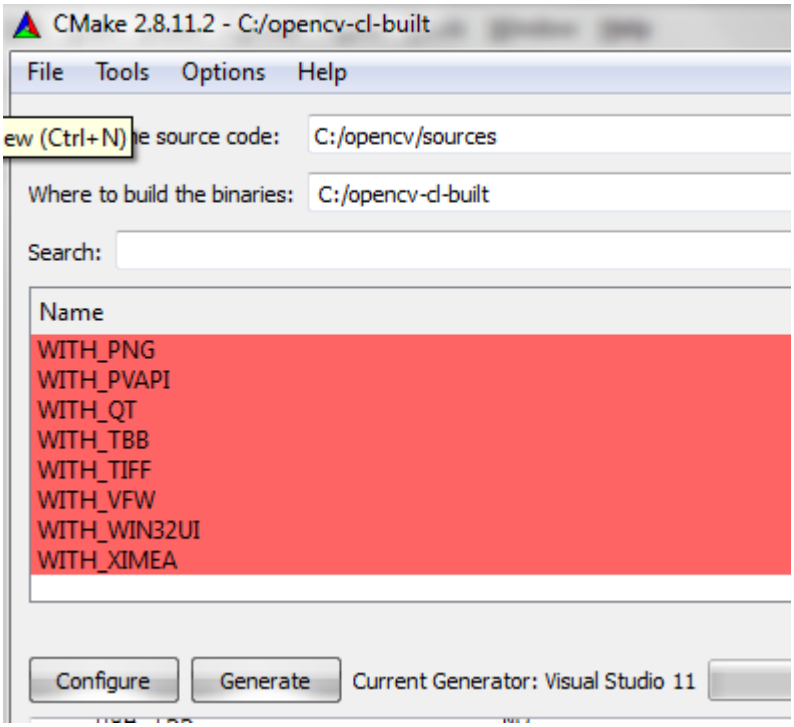
7. Open CMake-gui (Start > All Programs > Cmake-gui)

8. Fill the fields as follows (see the image below):

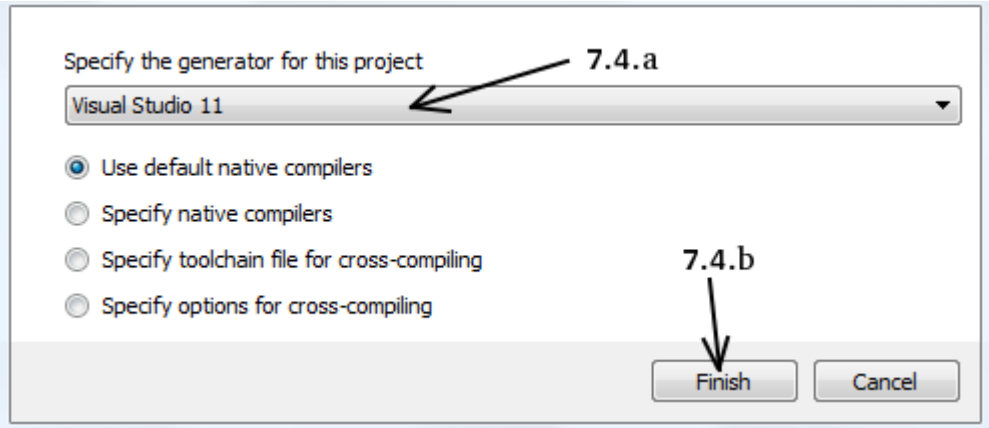
   8.1. Click on **Browse Source...** and locate the **opencv/sources** folder.

   8.2. Click on **Browse Build...** and locate the build folder we created.

   8.3. Click on **Configure**.

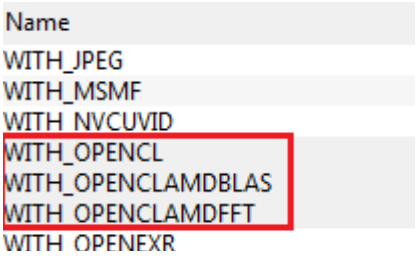


8.4. It will open a new window to select the compiler. Choose appropriate compiler (here, Visual Studio 11 for VS 2012) and click **Finish**.



8.5. Wait until analysis is finished.

9. You will see all the fields are marked in red. First few fields configure the build method. **Ensure that the following items are checked**. Leaving the others as default should be sufficient.:



10. Once checked, click **Configure** again. Everything should turn white.

***You can verify that the AMD SDKs were picked up by looking in the bottom pane, under OpenCL. It may be different depending on the versions of your software. Here is what I saw:*

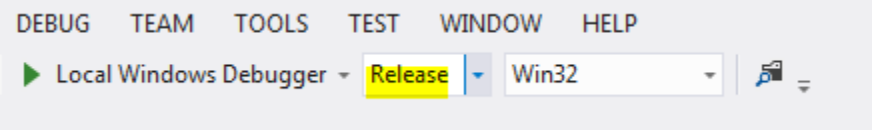
***OpenCL**:*

***Version:** dynamic*
***Include path**: C:/opencv/sources/3rdparty/include/opencl/1.2 C:/Program Files (x86)/AMD/clAmdFft/include C:/Program Files (x86)/AMD/clAmdBlas/include*
***Use AMD FFT**: YES*
***Use AMD BLAS**: YES*

11. Finally click the **Generate** button.

12. Now go to **C:\opencv_opencl_build**. There you will find the **OpenCV.sln** file. Open it with Visual Studio.

13. Check build mode as **Release** instead of **Debug**.



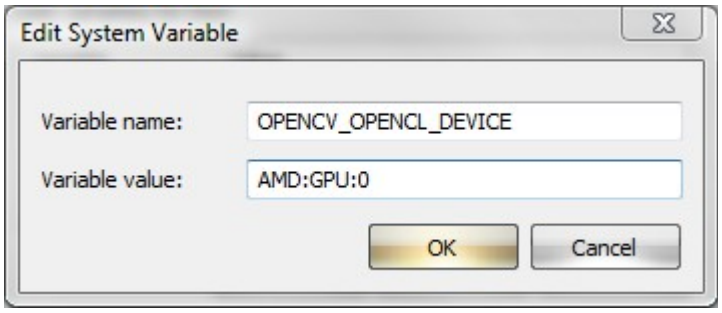
14. In Visual Studio, hit **F7** to build solution. **This will take a while to finish, and will amount to ~3 GB of project files.*

Configure Devices for use

Only one step needs to be taken to get OpenCL module to recognize device (GPU in this case).

15. Enable GPU device 1 of 2 ways:

    a. In Windows, go to **Control Panel > System > Advanced System Settings > Environment Variables**, and create the following **System Variable**:



More generically, you can enter **:GPU: as a value.*

    b. Later, in your C++ program, you can use the cv::ocl::setDevice function (with cv::ocl::getOpenCLPlatforms and cv::ocl::getOpenCLDevices)

Configure Project in Visual Studio

When configuring a project, there are two routes you can take to create an OpenCL project:

1. Using Existing Project:
   a. In Solution Explorer, go to **tests performance > opencv_perf_ocl > Src** > and click **main.cpp** to bring its contents into view.

   b. Insert your code into main.cpp, or create your own cpp file. Refer to **C:\opencv\sources\samples\ocl** for examples.

2. Creating New Project:
   a. If you plan to create a new project, the binaries, libraries, and dll's were placed in the following directories
      a. **Binaries** – C:\opencv-cl-built\bin
      b. **Libraries** – C:\opencv-cl-built\lib

   b. Create **New Project** in Visual Studio 2012.
   c. Choose **Release** (or Debug)as your *Solution Configuration.*
   d. In *Project Properties*, go to **Configuration Properties > C/C++**, and under *Additional Include Directories*, **copy/paste** the following:

C:/opencv/sources/modules/ocl/perf;C:/opencv/sources/modules/features2d/include;C:/opencv/sources/modules/highgui/include;C:/opencv/sources/modules/imgproc/include;C:/opencv/sources/modules/flann/include;C:/opencv/sources/modules/core/include;C:/opencv/sources/modules/ts/include;C:/opencv/sources/modules/ocl/include;C:/opencv/sources/modules/video/include;C:/opencv/sources/modules/objdetect/include;C:/opencv/sources/modules/ml/include;C:/opencv/sources/modules/calib3d/include;C:/opencv-cl-built/modules/ocl;C:/opencv/sources/modules/ocl/src;C:/opencv/sources/modules/ocl/test;C:/opencv/sources/3rdparty/include/opencl/1.2;C:/opencv-cl-built;C:/Program Files (x86)/AMD/clAmdFft/include;C:/Program Files (x86)/AMD/clAmdBlas/include;%(AdditionalIncludeDirectories)

   e. In *Project Properties*, go to **Configuration Properties > Linker > Input**, and under *Additional Dependencies*, **copy/paste** the following:

kernel32.lib
user32.lib
gdi32.lib
winspool.lib
shell32.lib
ole32.lib
oleaut32.lib
uuid.lib
comdlg32.lib
advapi32.lib
..\..\lib\Release\opencv_core248.lib
..\..\lib\Release\opencv_flann248.lib
..\..\lib\Release\opencv_imgproc248.lib
..\..\lib\Release\opencv_highgui248.lib
..\..\lib\Release\opencv_features2d248.lib
..\..\lib\Release\opencv_calib3d248.lib
..\..\lib\Release\opencv_ml248.lib
..\..\lib\Release\opencv_objdetect248.lib
..\..\lib\Release\opencv_video248.lib
..\..\lib\Release\opencv_ocl248.lib
..\..\lib\Release\opencv_ts248.lib
*Where **..\..** is the relative path of the lib directory copied from C:\opencv-cl-built\lib!!!

   f. Copy the .dll files from *C:\opencv-cl-built\bin\Release* into the same dir as your exe.
   g. Sample programs can be found in *C:\opencv\sources\samples\ocl.*

Troubleshooting

ERROR: C2977: 'std::tuple' : too many template arguments

SOLUTION: Include the following Preprocessor Definition: **_VARIADIC_MAX=10**