OPENCV INSTALL ON WINDOWS WITH CODE::BLOCKS AND MINGW

Code::Blocks is a *free C, C++ and Fortran IDE* built to meet the most demanding needs of its users. It is designed to be very extensible and fully configurable

OpenCV is released under a BSD license and hence it's free for both academic and commercial use. It has C++, C, Python and Java interfaces and supports Windows, Linux, Mac OS, iOS and Android. OpenCV was designed for computational efficiency and with a strong focus on real-time applications. Written in optimized C/C++, the library can take advantage of multi-core processing. Enabled with OpenCL, it can take advantage of the hardware acceleration of the underlying heterogeneous compute platform.

There are some issues with the latest pre-built binaries provided by the opencv developers. Therefore from now on we need to build libraries on our own. Instructions to make our own binaries is pretty straight forward and mentioned below.

INSTALLATION STEPS

Step 1: Install minGW

MinGW is a c/c++ compiler for windows, head to their website and download the latest version (right at the top where it says "looking or the latest version?")

http://sourceforge.net/projects/mingw/files/

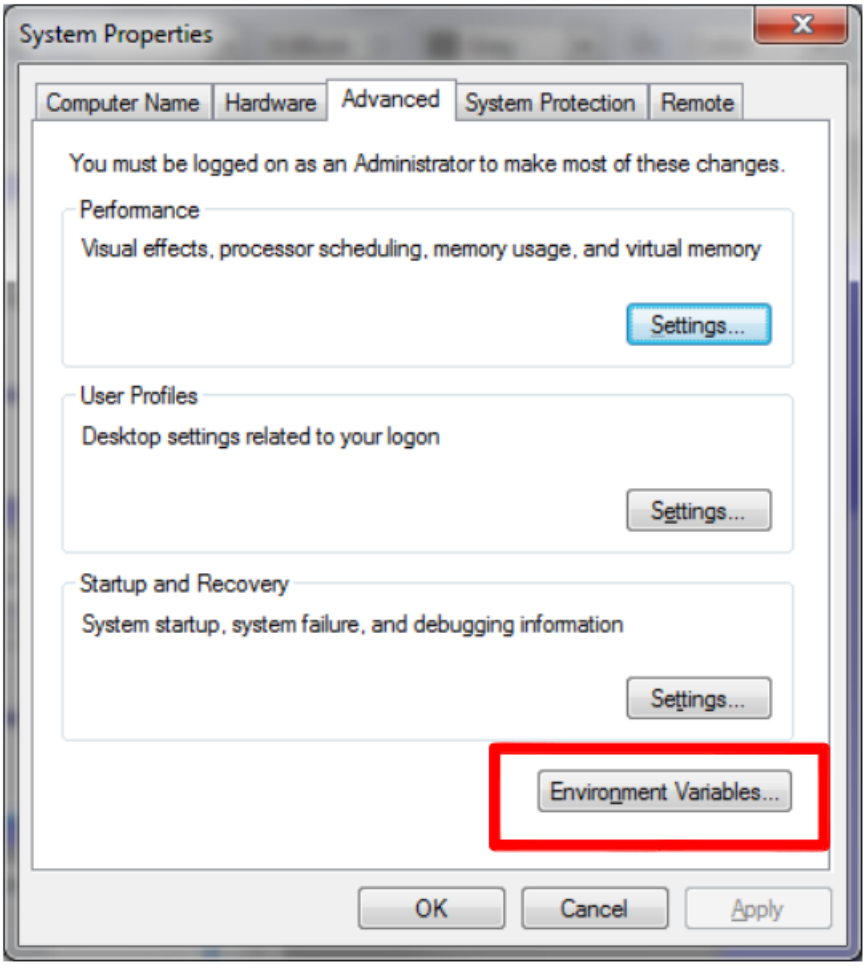
Install to the default location C:\MinGW

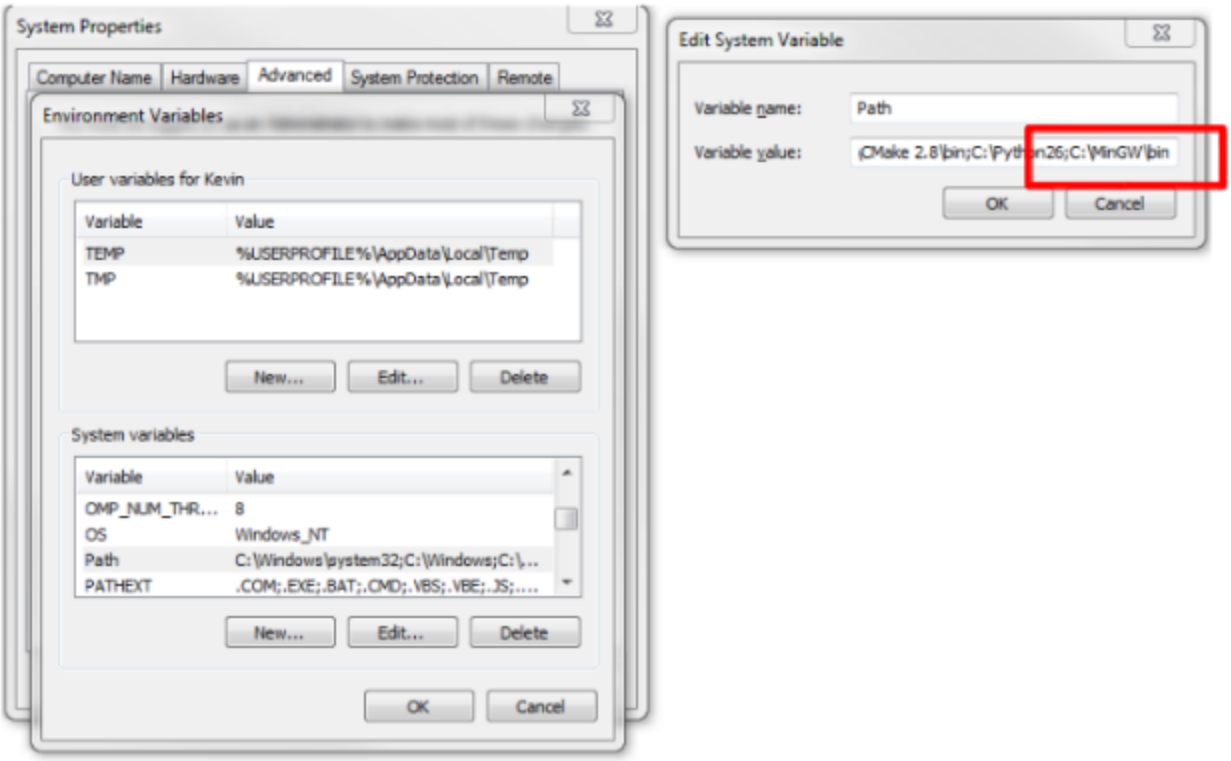
From the options install mingw32-base and mingw32-gcc-g++, you can also install the other components if you wish, but all you need is the c++ compiler (g++).

Step 2: Add minGW to system path

Navigate to Control Panel -> System -> Advanced System Settings

and then:





Type a semi colon after the last entry in "path" and then paste your MinGW path (it should be C:\MinGW\bin if you chose the default location).

Afterwords open up a command prompt and type "path" to make sure it worked (you should see minGW somewhere in the print out, probably near or at the end).

Programs will need to be restarted for this change to take effect.

Step 3: Install Code::Blocks

Code::Blocks is an IDE (integrated development environment). Head to their website and download the latest version (codeblocks-10.05-setup.exe)

http://www.codeblocks.org/downloads/binaries

Install it to the default location

When the installer finished click yes to run Code::Blocks

then go to Settings -> Compiler and Debugger

Under the Toolchain Executables select GNU GCC Compiler from the drop down and then press AutoDetect

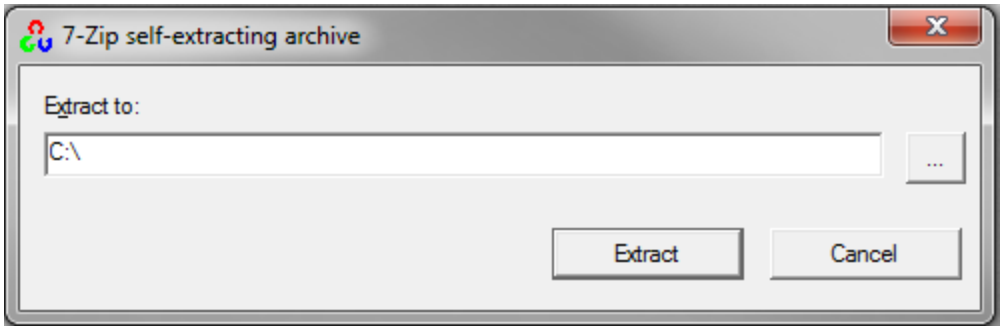
verify that Code::Blocks has found MinGW

If you like now might be a good time to test your Code::Blocks and MinGW setup with a simple Hello World C++ program.

Step 4: Install OpenCV

OpenCV is a library of Computer Vision functions. Head to their website and download the latest version (2.4.2 for Windows) http://opencv.org/downloads.html

Click on the OpenCV-2.4.2.exe and choose C:\ as the extract directory



OpenCV is now installed – but not configured with Code::Blocks

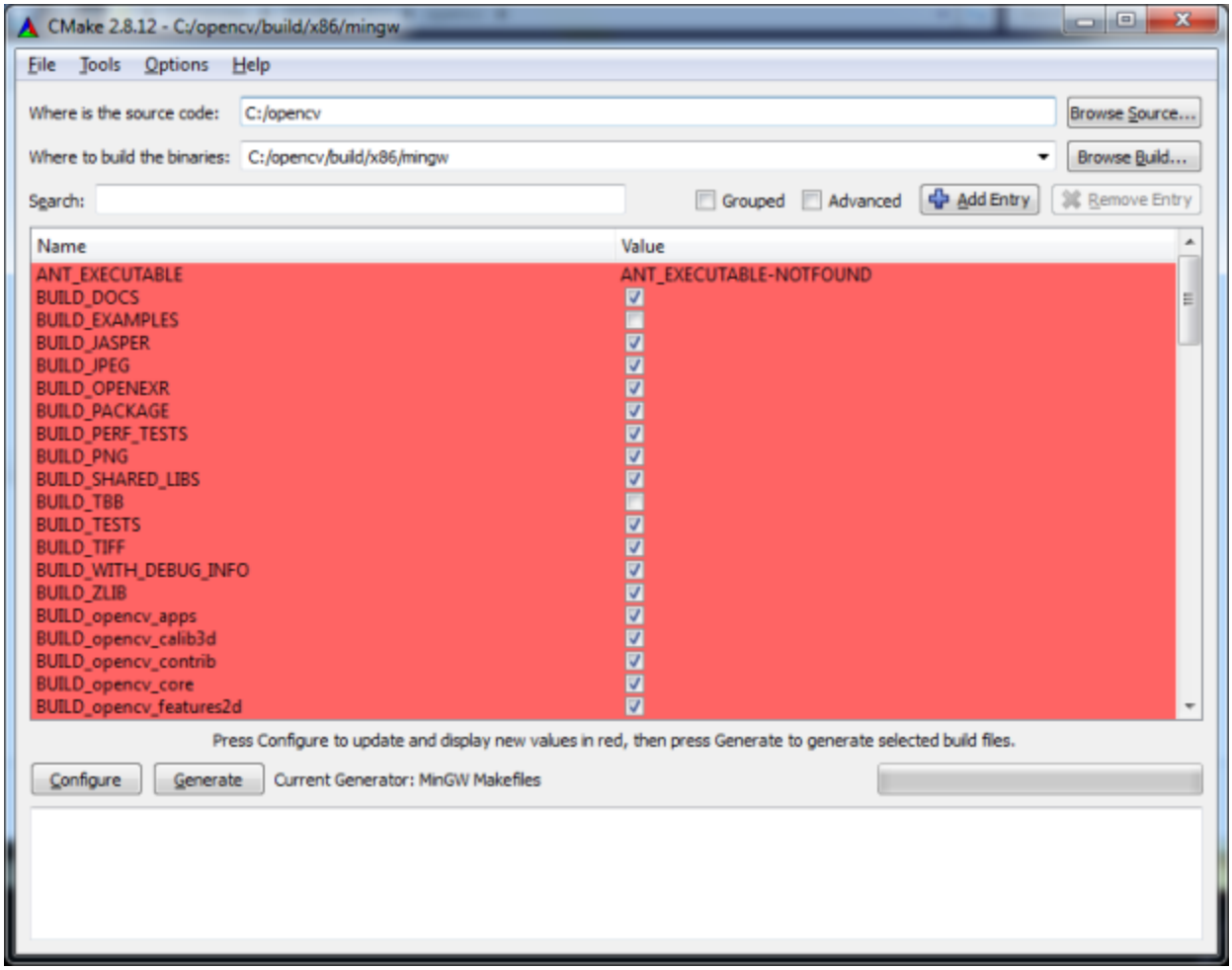
**** Update ****

If this is your first time through the tutorial doing a clean install then skip this step first and see if the supplied pre-built binaries will work for you, if you've already tried and had issues or if you really want to build your own then continue with this section.

First you'll need to download and install cmake

Open cmake and select C:\opencv as the source directory and C:\opencv\build\x86\mingw as the directory to build the binaries (you could select any directory but choosing this one will overwrite the pre-built OpenCV binaries and then the

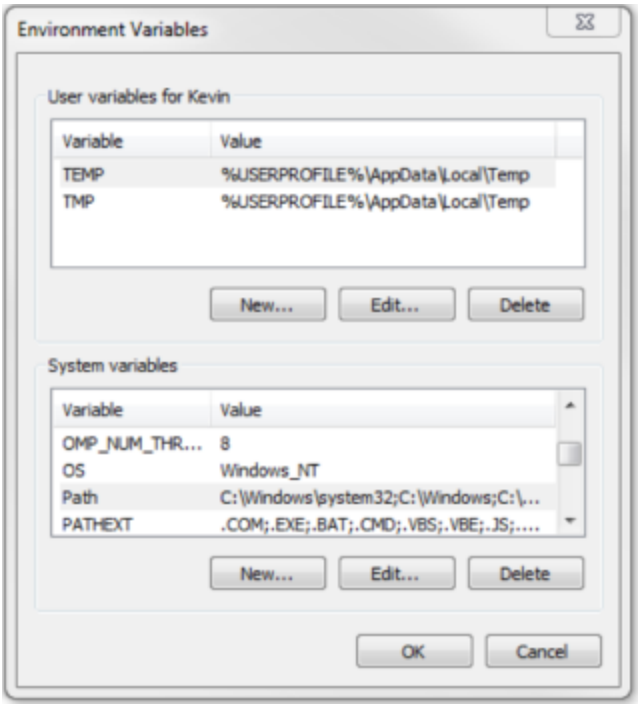
rest of the tutorial is the same. Click configure choose minGW makefiles wait and then click generate.

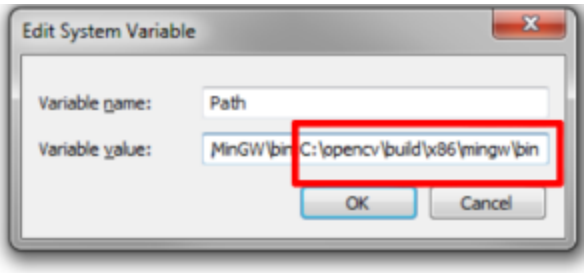


When cmake is done we need to open a command prompt in the build directory, so navigate to C:\opencv\build\x86\mingw then shift right click and choose open command window here then type "mingw32-make". Mingw will now start compiling OpenCV, this will take a bit so feel free to do something else, when you come back type "mingw32-make install" and continue with the rest of the tutorial as is.

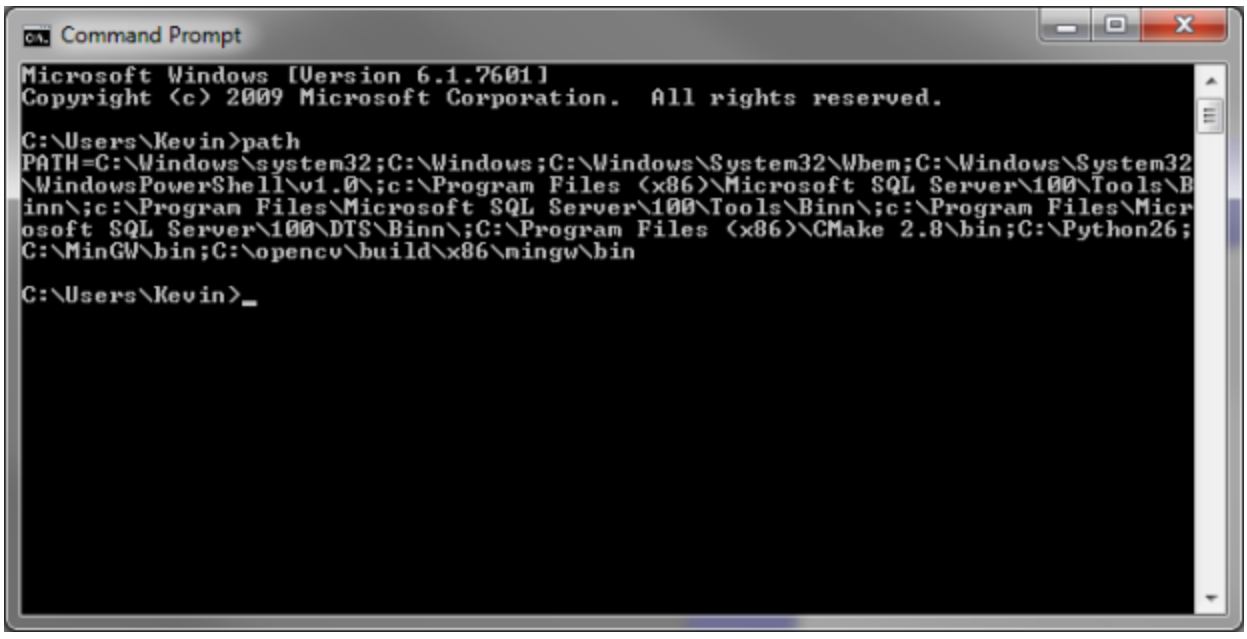
Step 5: Add OpenCV to the system path

C:\opencv\build\x86\mingw\bin (use the same process as above)





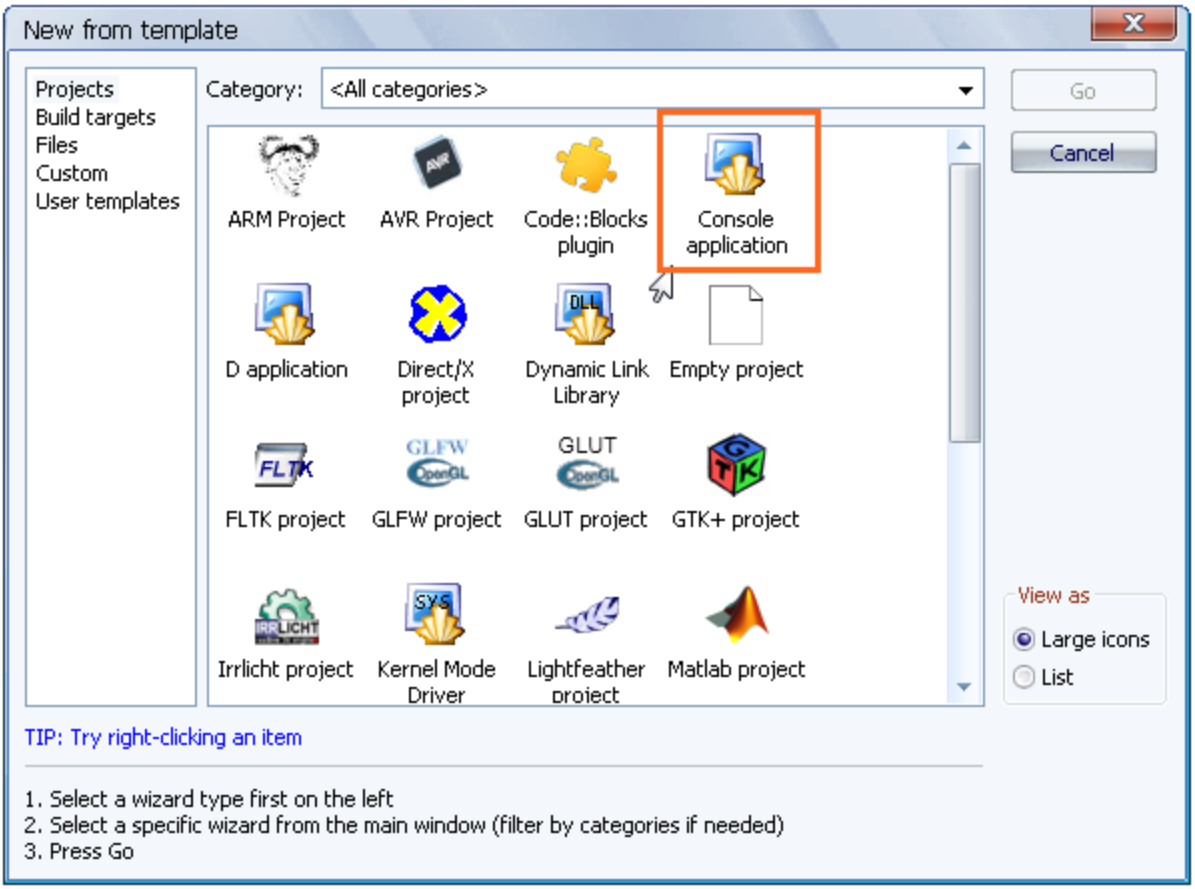
Note: Add x86 binaries regardless of your system type (32bit or 64bit) because minGW is 32bit.



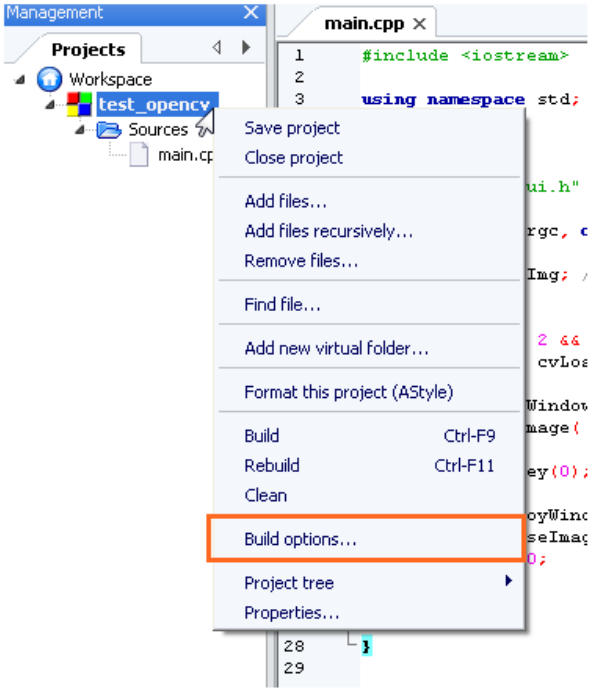
Verify that both MinGW and OpenCV are in your system path Make sure you restart Code::Blocks before continuing if you have it open.

Step 6: Configuring Code::Blocks with OpenCV

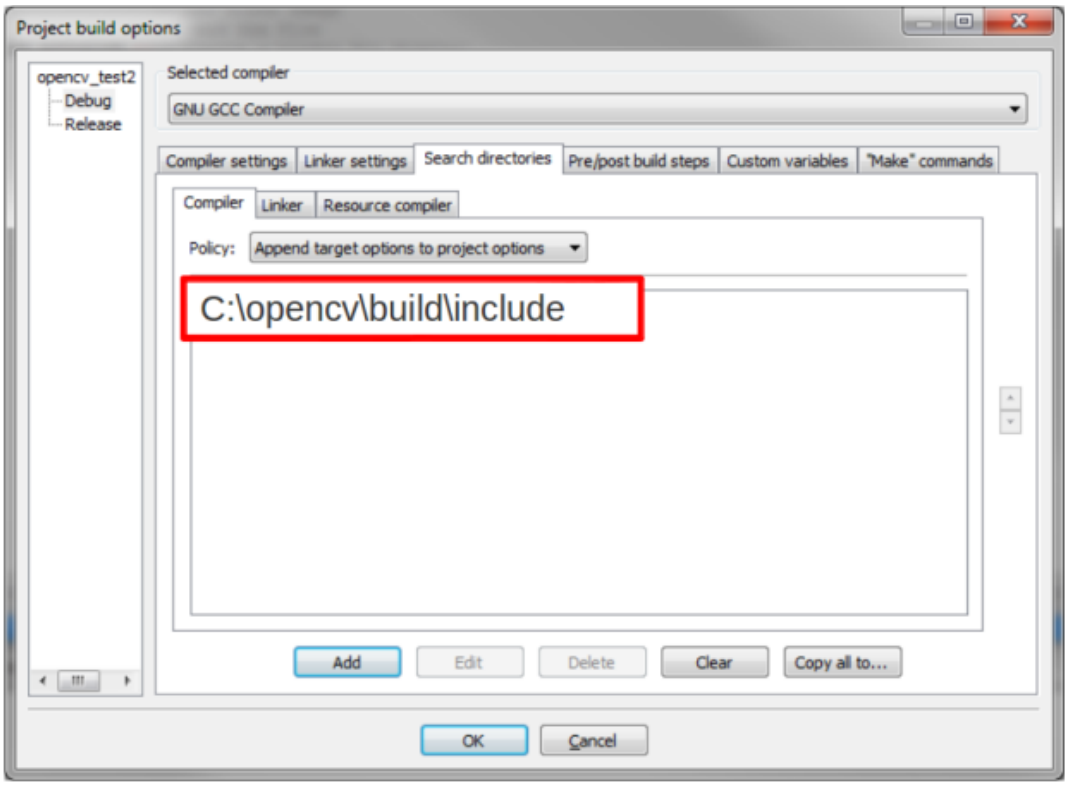
Make a new Code::Blocks Project:

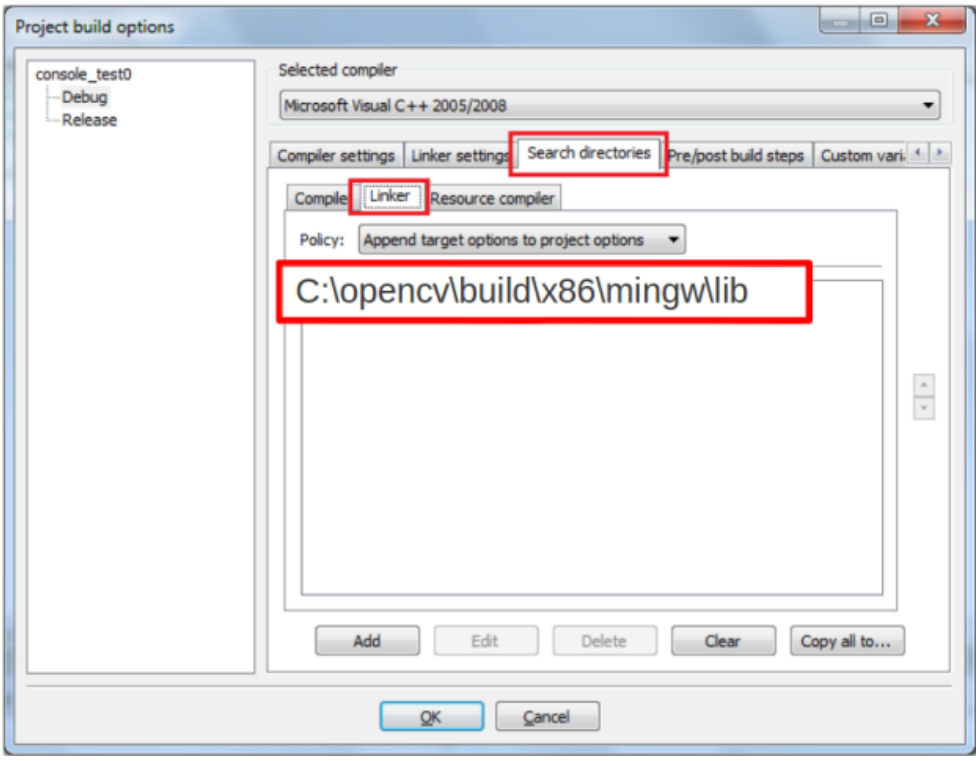


right click on your project and choose build options:

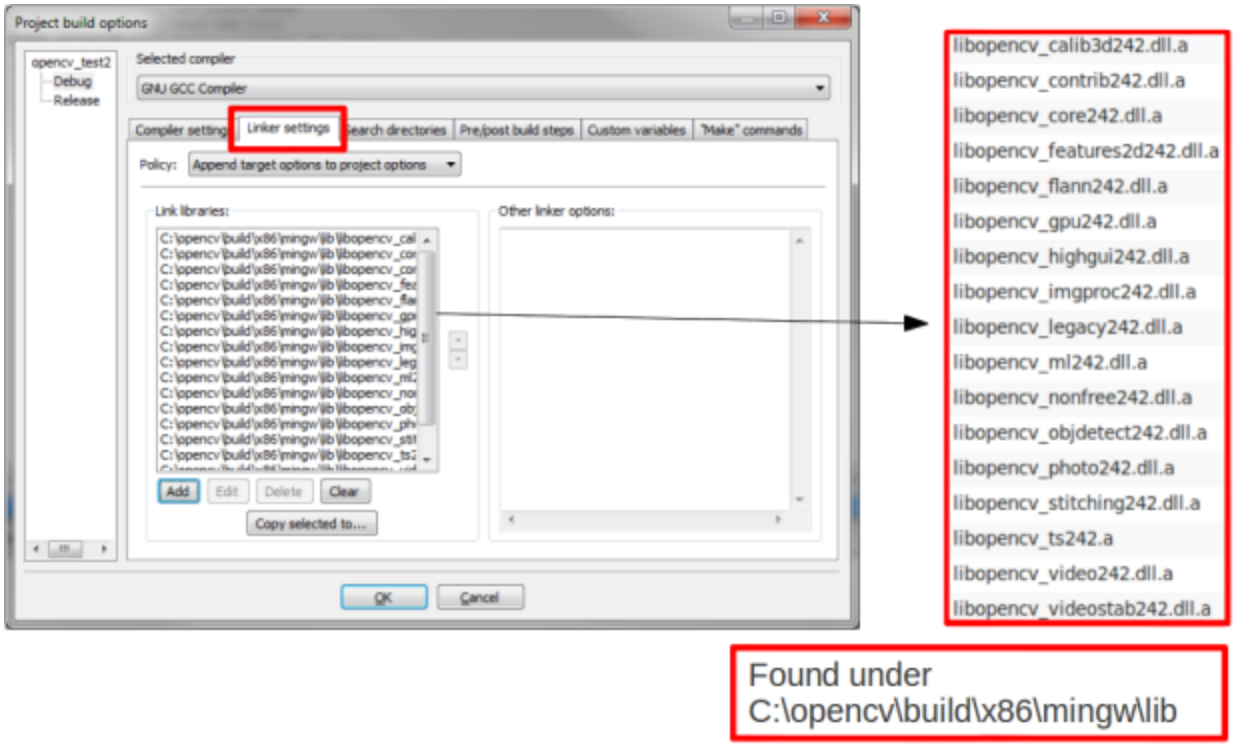


You can also change the global compiler settings from the menu bar at the top right.





Again Note – we are using 32-bit binaries even though the system is 64-bit because the compiler is 32-bit.



Now run this simple OpenCV “Hello World” program to test that the install has worked.

```
1     #include <opencv2/core/core.hpp>
2     #include <opencv2/highgui/highgui.hpp>
3
4     using namespace cv;
5
6     int main()
7     {
8         Mat image;// new blank image
9         image = cv::imread("test.png", 0);// read the file
10        namedWindow( "Display window", CV_WINDOW_AUTOSIZE );// create a window for
11    display.
12        imshow( "Display window", image );// show our image inside it.
13        waitKey(0);// wait for a keystroke in the window
14        return 0;
      }
```