Installation Cheat Sheet 1 - OpenCV 2.4.11 and C++
using Windows 7 + Visual Studio 2013 (Community Edition) + precompiled binaries
(should also work with Windows 8/8.1, not tested though)

download and install Visual Studio 2013 Community Edition (yes, its free) (choosing all default options will work fine)

download OpenCV 2.4.11

make a folder "C:\OpenCV-2.4.11" and extract OpenCV 2.4.11 to there

add the **bin** directory to the operating system PATH:
C:\OpenCV-2.4.11\opencv\build\x86\vc12\bin
pull up Command Prompt and verify bin directory is now in PATH, then reboot

from my MicrocontrollersAndMore GitHub page decide which example you are going to use:
CannyStill.cpp (uses a still image)
CannyWebcam.cpp (uses a webcam)
RedBallTracker.cpp (tracks a red ball, uses a webcam)

if you are going through this for the first time I suggest CannyStill.cpp

start Visual Studio 2013, make a new project
choose Visual C++, Win32 Console Application, name as you prefer, ex "SimpleCanny1"
set preferred location, uncheck "Create directory for solution" and "Add to source control", choose OK,
choose Next, uncheck "Precompiled Header" and "Security Development",
check "Empty Project" and verify "Console application" radio button is checked
choose Finish

right click in Solution Explorer, choose Add -> New Item
name C++ file as preferred, ex. "SimpleCanny1.cpp"
copy/paste the entire code from your chosen example into the .cpp file

if you are using an example with a still image (i.e. CannyStill.cpp), copy any JPEG image into the project directory and rename it "image.jpg"
you can use the "image.jpg" from my MicrocontrollersAndMore GitHub page if you would like to see the same results as in the video
(obviously if you are using a webcam example this step does not apply)

at this point Visual Studio will underline many of the lines of code with red because we have not yet informed Visual Studio as to the location of OpenCV

in VS go to:
Project -> Properties -> Configuration Properties -> VC++ Directories -> Include Directories
add the **include** directory: C:\OpenCV-2.4.11\opencv\build\include

in VS go to:
Project -> Properties -> Configuration Properties -> VC++ Directories -> Library Directories:

add the **library** directory: C:\OpenCV-2.4.11\opencv\build\x86\vc12\lib

in Windows Explorer (not within Visual Studio), navigate to the **lib** directory:
C:\OpenCV-2.4.11\opencv\build\x86\vc12\lib

if you currently do not have Windows 7 configured to allow viewing / editing of file extensions, go to:
Start -> Control Panel -> View by: Large icons -> Folder Options -> View tab -> uncheck "Hide extensions for known file types"

verify the libs listed are the same as this list:

opencv_calib3d2411d.lib
opencv_contrib2411d.lib
opencv_core2411d.lib
opencv_features2d2411d.lib
opencv_flann2411d.lib
opencv_gpu2411d.lib
opencv_highgui2411d.lib
opencv_imgproc2411d.lib
opencv_legacy2411d.lib
opencv_ml2411d.lib
opencv_nonfree2411d.lib
opencv_objdetect2411d.lib
opencv_ocl2411d.lib
opencv_photo2411d.lib
opencv_stitching2411d.lib
opencv_superres2411d.lib
opencv_ts2411d.lib
opencv_video2411d.lib
opencv_videostab2411d.lib

then in VS copy/paste this list of libs into:
Project -> Properties -> Configuration Properties -> Linker -> Input -> Additional Dependencies

next in the Visual Studio toolbar, verify that "Solution Configurations" and "Solution Platforms" are set to "Debug" and "Win32", respectively

run the program, either without debugging (choose Debug, then the hollow green arrow, or press Ctrl+F5)
or with debugging (solid green arrow or press F5)