

Installing OpenCV

Go to the OpenCV project website to download the OpenCV library:

<http://opencv.willowgarage.com>

For Windows, version 2.3 of OpenCV (the latest at the time of this writing) is available at:

<http://sourceforge.net/projects/opencvlibrary/files/opencv-win/2.3/>

Download the following file:

[OpenCV-2.3.0-win-superpack.exe](#)

When the download completes, run this file. Make the install location “C:\”. After the installer runs, the OpenCV files will be located at “C:\OpenCV2.3”.

Creating a Simple Project in Visual C++ 2010

In Visual C++ 2010, create a new empty project called “HelloOpenCV”. Add a file called main.cpp, where we will create a simple program to open and image in a window using the OpenCV libraries. Put this code in your main.cpp file:

```
#include <cv.h>
#include <highgui.h>
using namespace cv;

int main(int argc, char** argv) {

    Mat img = imread("cat.jpg");

    if (!img.data) {
        printf( " No image data \n " );
        return -1;
    }

    namedWindow("Example1", CV_WINDOW_AUTOSIZE);
    imshow("Example1", img);
    waitKey(0);

    return 1;
}
```

Add an image to your project folder. The example above uses “cat.jpg” from the sample images in the OpenCV files, found at:

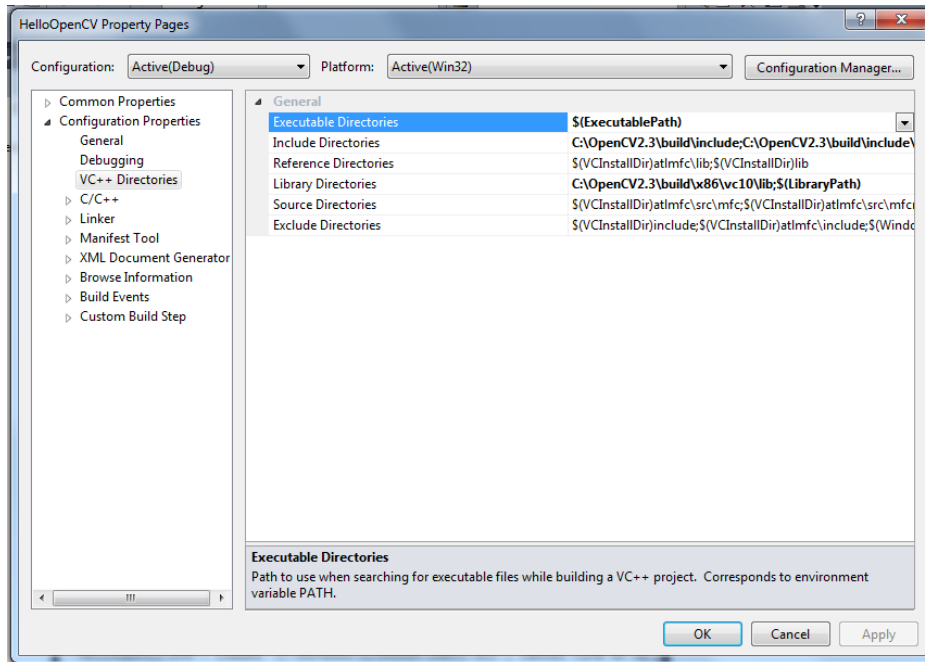
C:\OpenCV2.3\opencv\samples\cpp\tutorial_code\images\cat.jpg

Move the image to your project folder, in the same folder as your main.cpp file.

Setting Up Visual C++ Directories

Next we need to configure the project to be able to find the OpenCV header files and libraries.

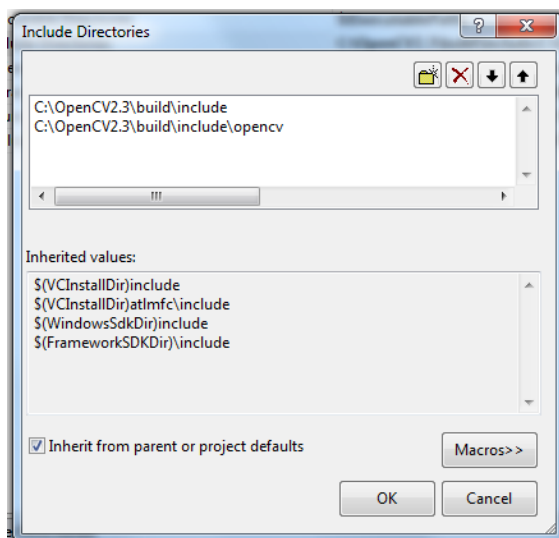
Go to Project > HelloOpenCV Properties. Then click on “VC++ Directories”.



Add the following paths to the Include Directories:

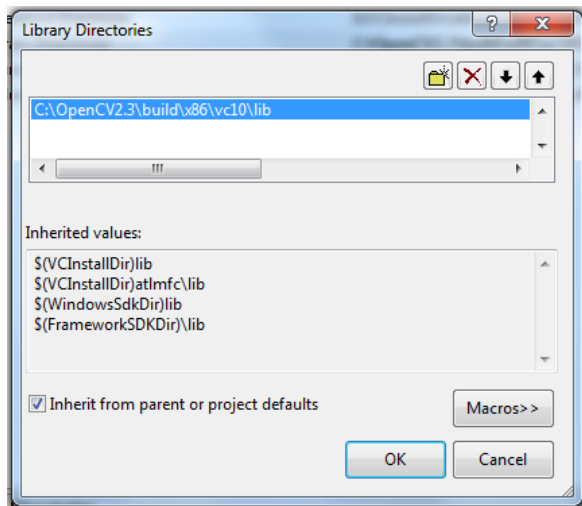
C:\OpenCV2.3\build\include

C:\OpenCV2.3\build\include\opencv



Add the following path to Library Directories:

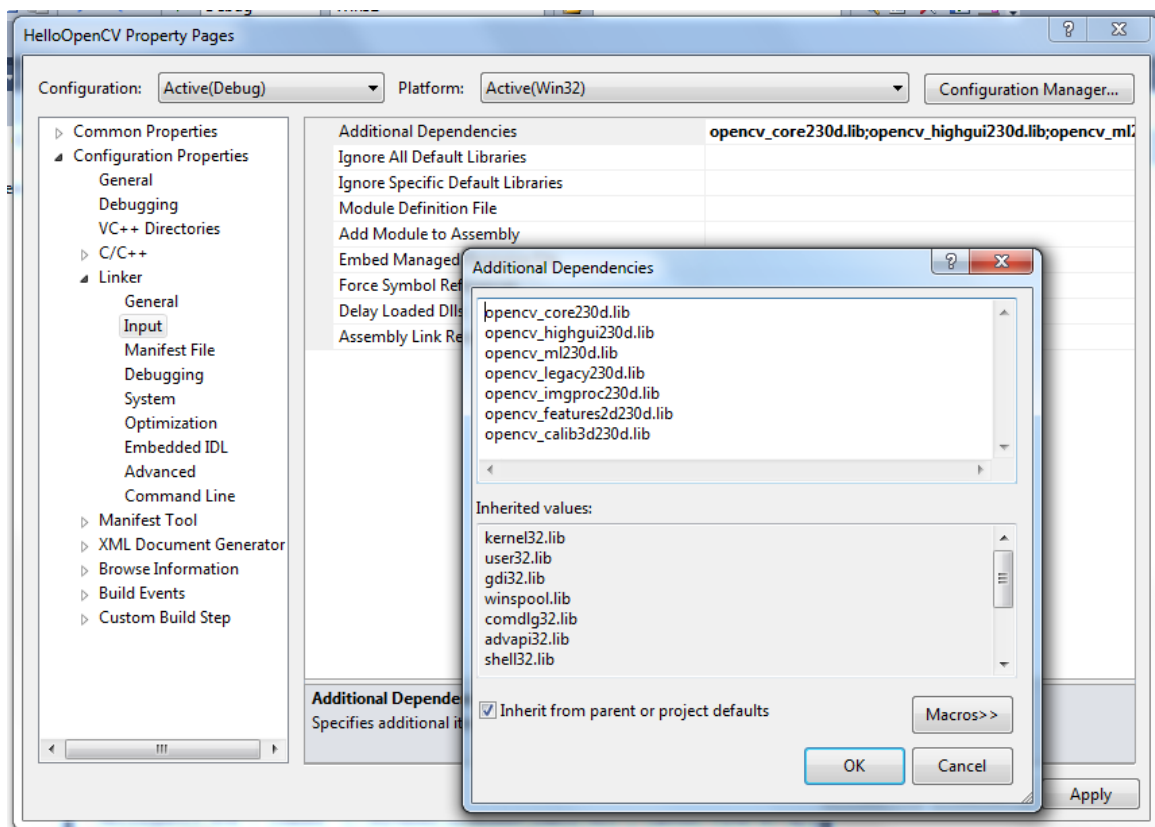
C:\OpenCV2.3\build\x86\vc10\lib



Setting Up Visual C++ Linking

Next, click on Linker > Input section in the properties. Add the following file names to the Additional Dependencies:

opencv_core230d.lib
opencv_highgui230d.lib
opencv_ml230d.lib
opencv_legacy230d.lib
opencv_imgproc230d.lib
opencv_features2d230d.lib
opencv_calib3d230d.lib

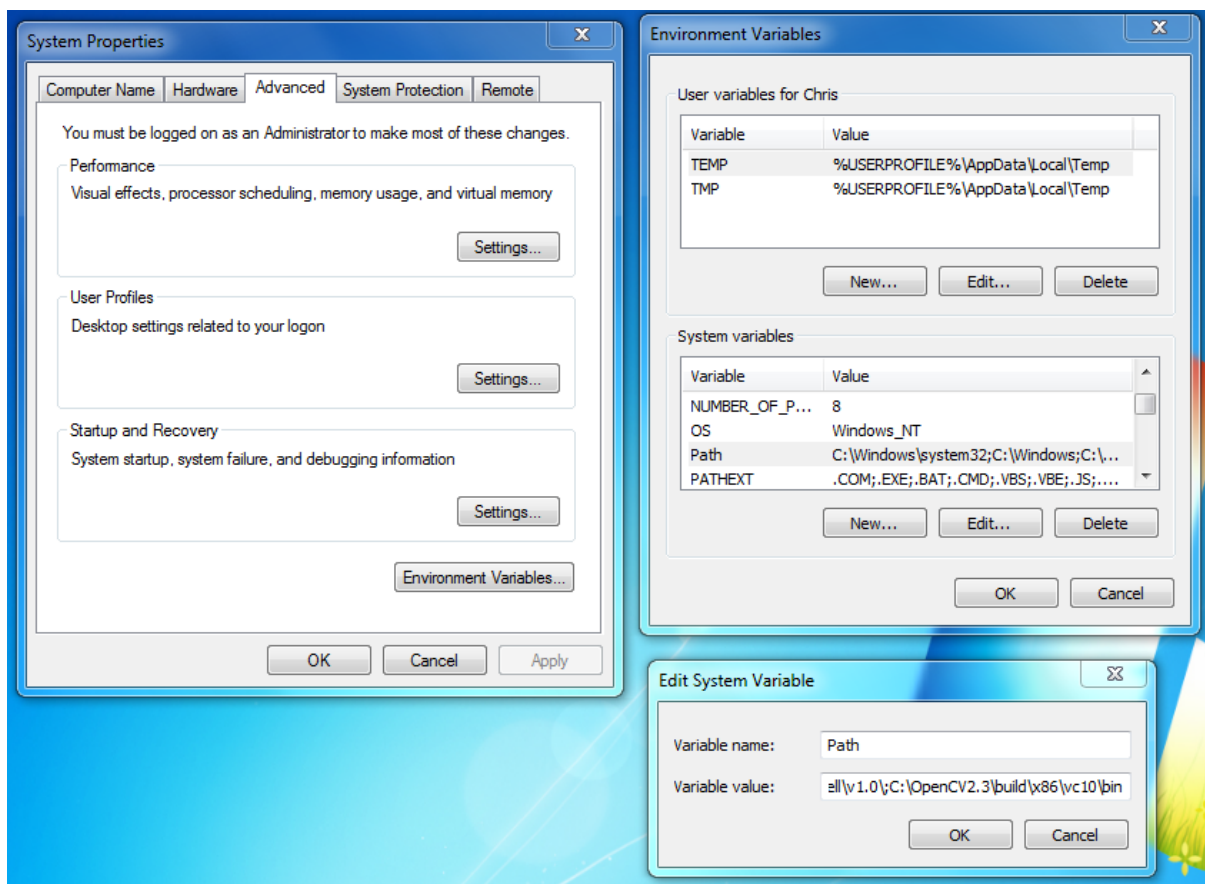


Updating the “Path” Environment Variable

Next, we need to set up Windows to be able to find the .dll files that are already built for the various modules in OpenCV.

Go to the Control Panel in Windows. Then select System, then Advanced System Settings. Click on the Advanced tab, then on the Environment Variables button. In the “System variables” section, select the line for the “Path” variable, and select the Edit button. In the “Variable value” box, add a semicolon and the following path:

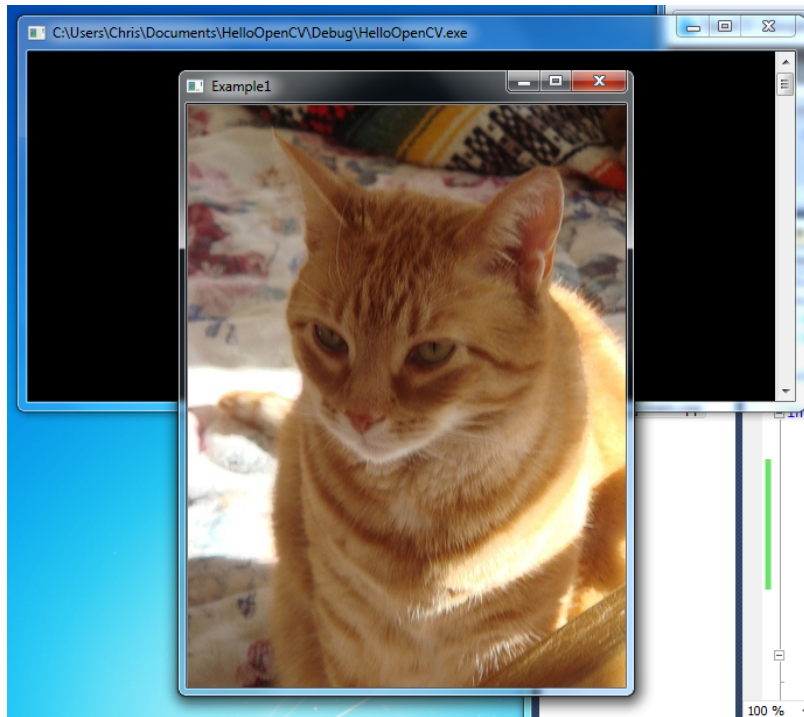
`C:\OpenCV2.3\build\x86\vc10\bin`



Next, restart Visual C++, otherwise the changes to the environment variables will not likely work (which results in an error like “can’t find foo.dll file”).

Building and Running the Program

You may now build the program and run it, which should create a window for the image, as well as a window for console output, similar to the following:



For documentation on the functions in this simple program, and for more tutorials, the official documentation is recommended reading:

<http://opencv.itseez.com/>

Also, there are several sample projects in the OpenCV folder now on your hard drive under C:\OpenCV2.3\opencv\samples.

Using the Properties for Other Projects

It is possible to import the settings above into other projects. You can create a properties file and import it into other project. These settings will then override the default settings.

Using a text editor, put the following XML into a file with a .props extension.

```
<?xml version="1.0" encoding="utf-8"?>
<Project ToolsVersion="4.0" xmlns="http://schemas.microsoft.com/developer/msbuild/2003">
  <PropertyGroup>
    <IncludePath>C:\OpenCV2.3\build\include;C:\OpenCV2.3\build\include\opencv;$(IncludePath)
  </IncludePath>
  </PropertyGroup>
  <PropertyGroup>
    <LibraryPath>C:\OpenCV2.3\build\x86\vc10\lib;$(LibraryPath)</LibraryPath>
  </PropertyGroup>
  <ItemDefinitionGroup>
    <Link>
      <AdditionalDependencies>opencv_core230d.lib;opencv_highgui230d.lib;opencv_ml230d.lib;opencv_legacy230d.lib;opencv_imgproc230d.lib;opencv_features2d230d.lib;opencv_calib3d230d.lib;%(AdditionalDependencies)</AdditionalDependencies>
    </Link>
  </ItemDefinitionGroup>
</Project>
```

In Visual C++ 2010 Express, you'll need to switch to the Expert Settings view, which will reveal more options from the menus. To turn on Expert Settings, go to Tools > Settings and select Expert Settings.

Then, to import the settings from the properties file, go to View > Property Manager. The projects in your solution will show in a list. Expand the project in the list and the configurations will appear, such configurations as Debug and Release. You'll apply the properties to each configuration as needed.

Under the project, right click on the configuration (such as Debug) where you want to import the settings and select "Add Existing Property Sheet". Navigate to where you saved the above .props file and select the file. These properties should now be applied to the configuration for the project.