Setting up OpenCV 2.0 on Netbeans 6.8 IDE on Windows Platform

Purpose
This tutorial steps you through the installation process of OpenCV 2.0 on Netbeans 6.8 IDE along with a working C++ example using the OpenCV library.

Author: Steve Hodkin © Copyright 2010 - All Rights Reserved.

It is highly recommend that you follow each step in order of appearance:

Step 1.

Install Cygwin Complier
http://www.cygwin.com/setup.exe
and follow installation guide.

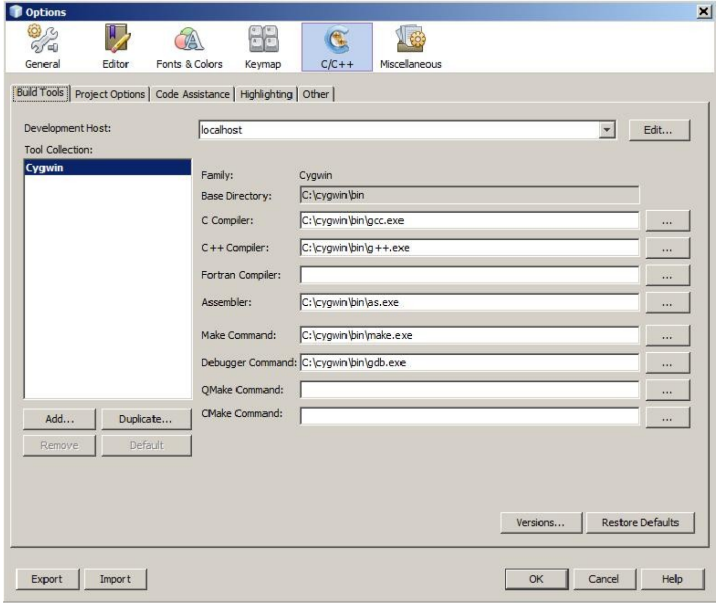
Step 2.

Download NetBeans 6.8 IDE
http://netbeans.org/index.html
and install.

Step 3.

NetBeans should now find Cygwin compiler and configure it automatically, if not don't worry do the following in Netbeans menu, but do check this is OK all the same...
Tools
Options
C/C++
Build Tools
and simply add the Cygwin in the **Tool Collection** and the paths point to the Cygwin directory.(fig 1.)

fig 1.



Development Host: localhost
Tool Collection: Cygwin
Family: Cygwin
Base Directory: C:\cygwin\bin
C Compiler: C:\cygwin\bin\gcc.exe
C++ Compiler: C:\cygwin\bin\g++.exe
Fortran Compiler:
Assembler: C:\cygwin\bin\as.exe
Make Command: C:\cygwin\bin\make.exe
Debugger Command: C:\cygwin\bin\gdb.exe
QMake Command:
CMake Command:

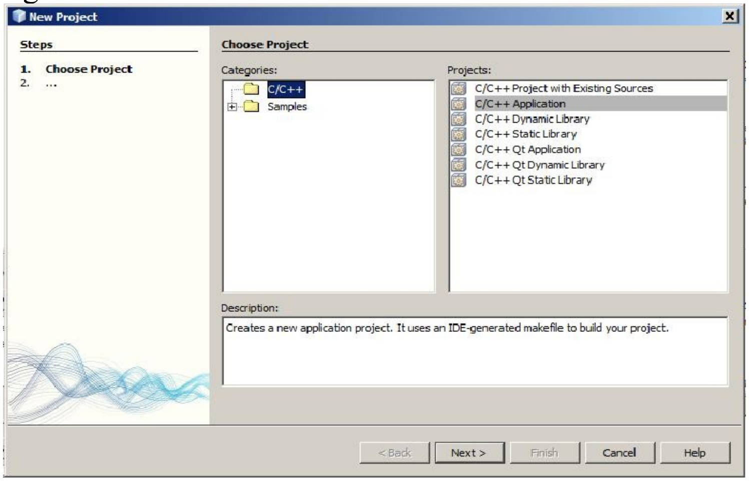
Finally click **OK**

Step 4.

Now we'll build a new project in Netbeans to test that we can create/compile & run a simply C/C++ program.

- First create a new folder on your desktop & call it **C++_Projects**
- Now in Netbeans **File>>New Project>>C/C++>>C/C++Application**(fig 2.)

fig 2.



1. Choose Project
2. ...

Categories: C/C++

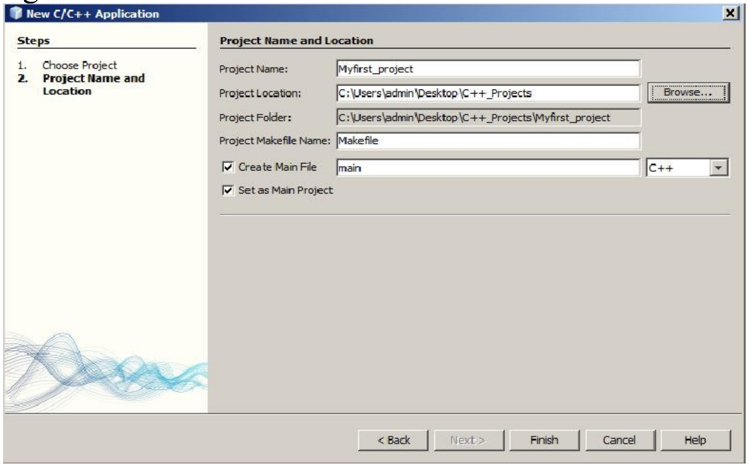
Projects: C/C++ Application

Description: Creates a new application project. It uses an IDE-generated makefile to build your project.

Click **Next**

In the next window do the following(fig 3.)
Project Name: **Myfirst_project**
Project Location: use the Browse button to point to the **C++_Project** folder you created on your desktop earlier, and check that the other settings match (fig 3).

fig 3.



1. Choose Project
2. **Project Name and Location**

Project Name: Myfirst_project
Project Location: C:\Users\admin\Desktop\C++_Projects
Project Folder: C:\Users\admin\Desktop\C++_Projects\Myfirst_project
Project Makefile Name: Makefile
☑ Create Main File: main, C++
☑ Set as Main Project

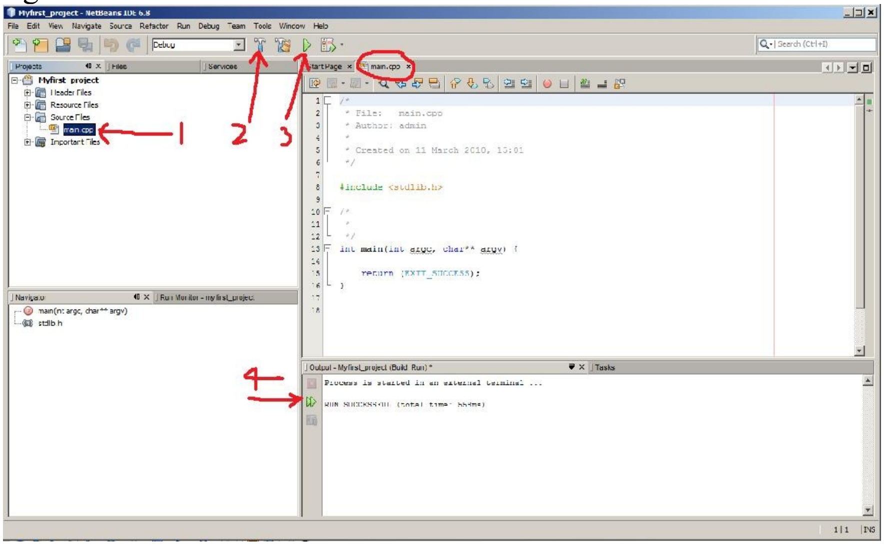
Finally click **Finish**

RECAP so far:

1. Installed Cygwin Compiler.
2. Installed Netbeans and configured Cygwin Compiler.
3. Created a folder on the desktop which contains our new C++ project named **Myfirst_project**

Build/Compile and Run our New Project

fig 4.



Myfirst_project - NetBeans IDE 6.8
File Edit View Navigate Source Refactor Run Debug Team Tools Window Help
Debug
Search (Ctrl+I)
Projects
Files
Services
Start Page
main.cpp
Myfirst project
Header Files
Resource Files
Source Files
main.cpp
Important Files
Navigator
Run Monitor - myfirst_project
main(int argc, char** argv)
stdlib.h
Output - Myfirst_project (Build, Run)
Tasks
1
2
3
4
1 | 1
INS

```
/*
 * File:   main.cpp
 * Author: admin
 *
 * Created on 11 March 2010, 13:01
 */

#include <stdlib.h>

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

    return (EXIT_SUCCESS);
}
```

```
Process is started in an external terminal ...

RUN SUCCESSFUL (total time: 558ms)
```

Steps

1. Double click on the main.cpp file under **Source Files** to load code in the right hand window(Item 1 fig 4.)
2. Click on the blue hammer (Item 2 fig 4.) to build the project.
3. Click on the green button (Item 3 fig 4.) to run the compiled exe.
4. If all goes well you should see RUN SUCCESSFUL (Item 4 fig 4.)

WARNING:

You should be aware that if you encounter any problems with any of the above steps or you can not build/compile or run your new project, then these issues need to be addressed before trying to install OpenCV, because it's probably not going to work.

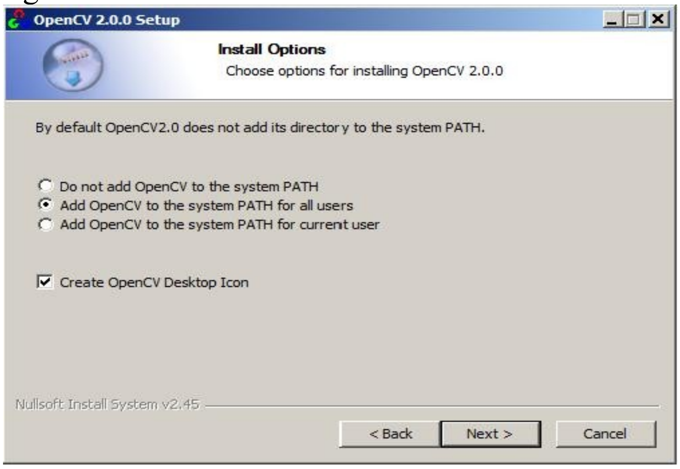
<u>OpenCV 2.0 Netbeans 6.8 IDE Installation guide</u>

Install OpenCV 2.0
http://sourceforge.net/projects/opencvlibrary/files/opencv-win/2.0/
and follow the installation guide.

Very Important Note:

OpenCV install options, add to **system Path** you need to select **ADD** very important(fig 5.)

fig 5.



Install Options

Choose options for installing OpenCV 2.0.0

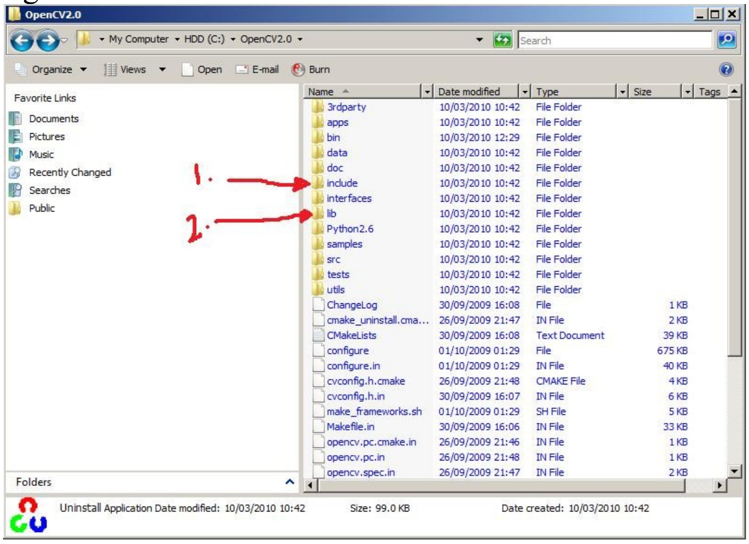
By default OpenCV2.0 does not add its directory to the system PATH.

○ Do not add OpenCV to the system PATH
● Add OpenCV to the system PATH for all users
○ Add OpenCV to the system PATH for current user

☑ Create OpenCV Desktop Icon

By default OpenCV install location is **C:\OpenCV2.0** so we are going to assume from here that this is the case.

Navigate to the OpenCV directory(fig 6.)
fig 6.



Name	Date modified	Type	Size	Tags
3rdparty	10/03/2010 10:42	File Folder		
apps	10/03/2010 10:42	File Folder		
bin	10/03/2010 12:29	File Folder		
data	10/03/2010 10:42	File Folder		
doc	10/03/2010 10:42	File Folder		
include	10/03/2010 10:42	File Folder		
interfaces	10/03/2010 10:42	File Folder		
lib	10/03/2010 10:42	File Folder		
Python2.6	10/03/2010 10:42	File Folder		
samples	10/03/2010 10:42	File Folder		
src	10/03/2010 10:42	File Folder		
tests	10/03/2010 10:42	File Folder		
utils	10/03/2010 10:42	File Folder		
ChangeLog	30/09/2009 16:08	File	1 KB	
cmake_uninstall.cma...	26/09/2009 21:47	IN File	2 KB	
CMakeLists	30/09/2009 16:08	Text Document	39 KB	
configure	01/10/2009 01:29	File	675 KB	
configure.in	01/10/2009 01:29	IN File	40 KB	
cvconfig.h.cmake	26/09/2009 21:48	CMAKE File	4 KB	
cvconfig.h.in	30/09/2009 16:07	IN File	6 KB	
make_frameworks.sh	01/10/2009 01:29	SH File	5 KB	
Makefile.in	30/09/2009 16:06	IN File	33 KB	
opencv.pc.cmake.in	26/09/2009 21:46	IN File	1 KB	
opencv.pc.in	26/09/2009 21:48	IN File	1 KB	
opencv.spec.in	26/09/2009 21:47	IN File	2 KB	

Uninstall Application Date modified: 10/03/2010 10:42 Size: 99.0 KB Date created: 10/03/2010 10:42

The folders we're interested in are the **include**(Item 1 fig 6.) and the **lib**(Item 2 fig 6.)

- include folder contains the header files for OpenCV
- lib folder contains the dlls/library files

The first thing we need to do is open the file **cxoperation.hpp** this is found in the **...include/opencv** folder and change lines 67-68

from this

```
67      #include <bits/atomicity.h>
68      #if __GNUC__ >= 4
```

to this

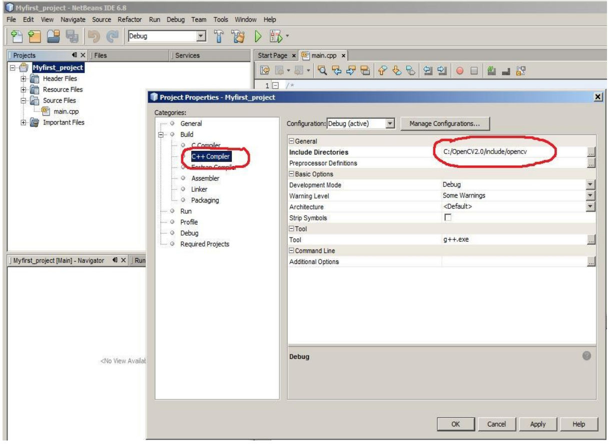
```
67      #include <bits/atomicity.h>
68      #if __GNUC__ >= 4 || __CYGWIN__
```

then **Save** the file.

Its best to use a text editor that can display line numbers to make this easy,VIM etc, you can use Netbeans IDE too of course to do this.

Now in Netbeans IDE, right click **Myfirst_project** and select **Properties**(fig 7.)

fig 7.



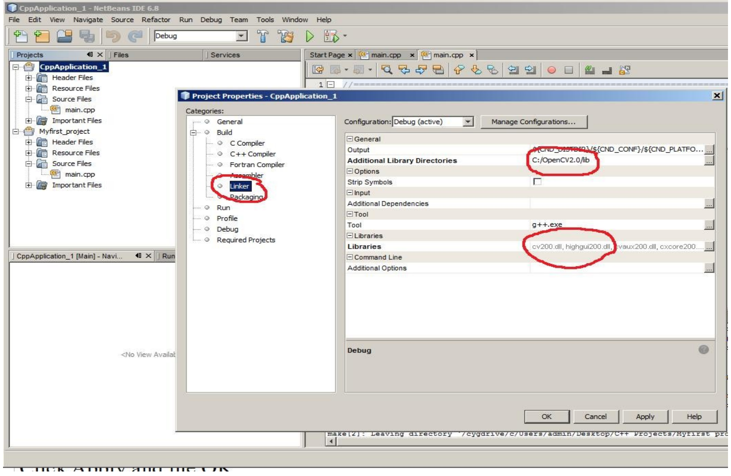
Configuration: Debug (active)

General
Include Directories: C:/OpenCV2.0/include/opencv
Preprocessor Definitions:
Basic Options
Development Mode: Debug
Warning Level: Some Warnings
Architecture: <Default>
Strip Symbols: □
Tool
Tool: g++.exe
Command Line
Additional Options:

Debug

- Click on **C++ Compiler** tab and change the **Include Directories** path so it points to **c:/OpenCV2.0/include/opencv**(fig 7.) this is where the header files are, for opencv projects.
- Next we need to tell Netbeans where to find the library files needed for opencv projects, this is under the Linker tab (fig 8.) the only two library files we need to install at this moment are **cv200.dll** and **highgui200.dll** the others dlls can be loaded later and also the **Additional Library Directories** points to **c:/OpenCV2.0/lib**

Fig 8.



Project Properties - CppApplication_1

Configuration: Debug (active)

General
Output: ${CND_DISTDIR}/${CND_CONF}/${CND_PLATFO...
Additional Library Directories: C:/OpenCV2.0/lib
Options
Strip Symbols: □
Input
Additional Dependencies:
Tool
Tool: g++.exe
Libraries
Libraries: cv200.dll, highgui200.dll, cvaux200.dll, cxcore200....
Command Line
Additional Options:

```
make[2]: Leaving directory `/cygdrive/c/Users/admin/Desktop/C++ Projects/Myfirst pro
```

Click **Apply** and the **OK** Buttons

That's all the settings done folks!

Next testing the OpenCV 2.0 installation

- Plug a web cam into a usb port
- Copy and Paste the entire code below into the **main.cpp** file(*overwrite any previous code*)
- **Build** project Blue Hammer
- **Run** Green Button

If all's well your web cam should light up and a demo window should appear with the output of your web cam(fig 9.)

fig 9.

I hope you enjoyed the tutorial...
Steve Hodkin

And one final bonus, Eclipse user's check out this link for installing Opencv2.0 on Eclipse IDE.
http://carrierfrequency.blogspot.com/2010/02/setting-up-opencv-20-in-eclipse-on.html

```
//============================================================================
// Name        : main.cpp
// Author      : Steve Hodkin
// Version     : 1.0
// Copyright   : Your copyright notice
// Description : opencv example in C++, Ansi-style
//============================================================================

#include <stdio.h>
#include "cv.h"
#include "highgui.h"

using namespace std;

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

        //create and setup a window called "Demo"
        cvNamedWindow("Demo", CV_WINDOW_AUTOSIZE);
        //setup camera capture process
        CvCapture* capture=cvCaptureFromCAM(0);

        //loop and keep displaying web cam image
        while( 1 ) {
                //exit when esc key pressed
                if (cvWaitKey(100)== 27) break;
                //get's image
                IplImage* frame = cvQueryFrame(capture);
                //display's image in the window
                cvShowImage("Demo", frame);
        }

  //clean up and destroy Demo window
  cvDestroyWindow("Demo");
  return 0;
}
```