2.8 Installation

The Mac minis in the Roberts 225 computer lab have all the software libraries installed that we will be using in this course. If you prefer to use the Eclipse IDE for programming, there is a PyDev plugin that provides syntax highlighting, code completion, and other development aides for Python programming. You’ll need to install your own version of Eclipse in your user directory instead of using the version provided on the base install—follow the directions for Eclipse and PyDev below.

If you want to install these packages on your computer, the task is much easier on a Linux box than a Mac. On Linux, search your package manager for the appropriate packages, and just install them. Let the dependencies work for you—you’ll get Qt, SIP, and PyQt4 by just installing PyQt4. If you don’t have the most up-to-date versions available (e.g. OpenCV 2.3.1), you can download and compile from source.

For Mac OS X, the MacPorts project provides the most reliable way to create a consistent installation of all the software. Follow the steps below in order. Before you begin, know that this can easily take the better part of a day to complete.

1. Xcode Tools

   This is on the Mac OS X Install DVD that came with your computer. I have a version of Xcode for Mac OS X 10.6 available. You may also be able to get this from the Mac Dev Center, which you should be able to join without paying, but Apple is making moves to close that door.

2. Install MacPorts from MacPorts.org

   This install will modify your .bash_profile script, so you may want to check this after installation. You will want the path /opt/local/bin/ to appear early in your PATH.

```
1 $ sudo port install python27 py27-numpy py27-scipy
2 $ sudo port install py27-matplotlib +qt4 +tkinter
3 $ sudo port install opencv +python27 +qt4
```

3. Option installs

   Again, using MacPorts.

```
1 $ sudo port install py27-pyqt4
2 $ sudo port install ImageMagick
3 $ sudo port install libdcl394
```

Note: PyQt4 is not required to get OpenCV working, but it will probably be pulled in if you install the +qt4 variant of matplotlib. Also, if you want
to be able to add sophisticated graphical user interface (GUI) controls to your applications, you will need SIP and PyQt4. (OpenCV uses the C++ API of Qt.)

ImageMagick provides a useful command-line image conversion program, convert, as well as other useful tools.


If you want to use the PyDev Python development tools for Eclipse, you probably need to install the previous version of Eclipse (Indigo, 3.7), using the Mac Cocoa 64-bit installer from here: http://www.eclipse.org/downloads/packages/release/indigo/r. I was unable to get PyDev to install with Eclipse version 3.7.1.

Download, unzip, and move the eclipse folder to the directory where you want it installed (/Applications on your machine, /Users/username in the lab).

Go to Help → Install New Software. Click the Add... button near the top of the Install dialog. On the Add Repository dialog, set Name to PyDev and Location to http://pydev.org/updates. Back on the Install dialog, select the PyDev for Eclipse and Mylyn (optional) plugins, then press Finish.

Go to Eclipse → Preferences. Find the Pydev / Interpreter - Python options. Under Python Interpreters, click New...; for Name use MacPorts, and for executable, browse to the Python executable, (/opt/local/bin/python2.7). Press OK, then Apply on the Preferences dialog (this will take a while), then OK. When you set up a new Python project, make sure it’s configured to use this interpreter—it’s best to only configure one interpreter. You may have to reconfigure the interpreter if you change workspaces.