

今天大部分时间都是在琢磨如何在Mac OS X上进行OpenCV项目的开发，尝试的开发工具有Xcode(版本是4.6.1)和Eclipse，使用的OpenCV版本是2.4.6。

如果只是需要OpenCV的相关头文件以及动态库，请直接执行 `brew install opencv` (如果安装了Homebrew的话)，如果不行，请看下面的OpenCV源码编译安装过程。

1.安装CMake

安装CMake可以使用MacPorts，也可以使用Homebrew，如果以前安装过两者中的任何一个就用那个进行安装吧，我用的是Homebrew，推荐使用Homebrew，真正的“佳酿”，命令如下：

```
sudo port install cmake //macports
sudo brew install cmake //homebrew
```

2.编译OpenCV

OpenCV下载地址：<http://sourceforge.net/projects/opencvlibrary/>

目前最新版本是2.4.8，我使用的是2.4.6，下载后解压，执行下面代码：

```
cd <path-to-opencv-source>
mkdir release
cd release
cmake -D CMAKE_BUILD_TYPE=RELEASE -D CMAKE_INSTALL_PREFIX=/usr/local ..
make
sudo make install
```

[完成之后在 `/usr/local/include` 目录下便有了 `opencv` 和 `opencv2` 两个目录，在 `/usr/local/lib` 目录下有很多的 `opencv` 相关的动态库，例如 `libopencv_core.dylib` 等等，还有几个其他的文件，它们都存放在 `/usr/local` 目录下]

[注1:如果不需要了，想要卸载 OpenCV的话，可以回到 `release` 目录，执行 `sudo make uninstall`，然后手动删除一些 `/usr/local` 下与OpenCV有关的目录和文件]

[注2:如果不想把OpenCV安装在默认的 `/usr/local/` 目录下的话，例如为了防止Homebrew中对opencv部分的报错，而又无法使用Homebrew正常安装opencv的情况下，可以考虑将opencv安装到其他的位置，修改 `CMAKE_INSTALL_PREFIX=/usr/local` 即可，但是在Eclipse中的项目中可能会出现一些问题，详情看后面]

其他参考内容：

[Building OpenCV from Source Using CMake, Using the Command Line](#)

[Installing OpenCV](#)

3.使用Xcode进行OpenCV项目开发

1.Open Xcode, choose `New -> New Project -> Command Line Tool`

2.Name it and select `C++` for type

3.Click on your project from the left menu. Click the `build settings` tab from the top. Filter all. Scroll to `Search Paths`. Under `header search paths`, for debug and release, set the path to `/usr/local/include`. Under `library search paths`, set the path to `$(PROJECT_DIR)`. Finally, check if `C++ standard library` is `libstdc++` or not, if not, change it to this!

4. Click on your project from the left menu. `File->New->New Group`, Name the group `OpenCV Frameworks`.

5. Select the folder (group) you just labeled, `OpenCV Frameworks` in the left menu. Go to `File -> add Files`, Type `/`, which will allow you to manually go to a folder. Go to -> `/usr/local/lib`

6. Select both of these files, `libopencv_core.dylib`, `libopencv_highgui.dylib`, and click `Add`. (you may need to add other library files from this folder to run other code.)

7. You must include this line of code in the beginning of your main.cpp file:

```
#include <opencv2/opencv.hpp>
```

可以修改main.cpp，代码如下，运行结果就是显示一张指定的图片。

```
#include <opencv2/opencv.hpp>using namespace cv;
int main(int argc, char** argv) {
    Mat image;
    image = imread("/Users/hujiawei/Pictures/others/other_naicha/naicha.jpg", 1);
    namedWindow("Display Image", WINDOW_AUTOSIZE);
    imshow("Display Image", image);
    waitKey(0);
    return 0;
}
```

其他参考内容：

[C++ linking error after upgrading to Mac OS X 10.9 / Xcode 5.0.1](#)

[MathLink linking error after OS X 10.9 \(Mavericks\) upgrade](#)

4.使用Eclipse进行OpenCV项目开发

如果使用Eclipse开发的话按照下面的步骤进行：

1. 按照正常的步骤，使用Eclipse建立一个 `Mac C++` 工程，包含一个cpp文件

2. 右击工程名，选择 `Properties`，在属性配置页中选择，点击 `C/C++ Build`，在下拉选项中选择 `Settings`。在右边的选项卡中选择 `Tool Settings`。

3. 在 `GCC C++ Compiler` 选项列表中选择 `Includes`，在 `Include paths(-I)` 中添加安装好的opencv的头文件存放目录：`/usr/local/include/` [存放opencv头文件的目录，自行看情况而定]

4. 在 `MacOS X C++Linker` 选项列表中选择 `Library`，在 `Library search path (-L)` 中添加安装好的opencv dylib文件存放目录：`/usr/local/lib/` [经过我的测试只能是这个目录！其他目录甚至是它的子目录都不行！如果在其他路径中，复制过来也行！]

5. 在 `MacOS X C++Linker` 选项列表中选择 `Library`，在 `Libraries(-l)` 中依次点击 `+` 号，添加需要使用的lib文件(通常情况下，使用前三个，注意不要包括前缀 `lib`，可以添加版本号)：

opencv_core opencv_imgproc opencv_highgui opencv_ml opencv_video opencv_features2d opencv_calib3d opencv_objdetect
opencv_contrib opencv_legacy opencv_flann

6. 重新build项目即可。

如果遇到问题 `ld: symbol(s) not found for architecture x86_64`，先检查代码中是否需要包含还没有添加的库文件，再检查是否是其他问题。如果是Mac平台，下面还有一个关于问题 `ld: symbol(s) not found for architecture x86_64` 的解释可供参考：

There are two implementations of the standard C++ library available on OS X: libstdc++ and libc++. They are On 10.8 and earlier libstdc++ is chosen by default, on 10.9 libc++ is chosen by default. To ensure compatibility To do this, add -stdlib=libstdc++ to the linking command.

更多相关内容参考：

<http://blog.sciencenet.cn/blog-702148-657754.html>

5. 阅读开源项目

阅读开源项目 [Mastering OpenCV with Practical Computer Vision Projects](#) 中的代码，以第8章 Face Recognition using Eigenfaces or Fisherfaces 为例

编写一个 shell，内容如下(修改自 `README.txt`)，其中的 `OpenCV_DIR` 为 OpenCV 源码编译后得到的文件夹(如上面的 release 目录)，执行这个 shell 便可以得到 Xcode 项目，当然打开这个项目之后还要修改相应的配置。

```
export OpenCV_DIR="/Volumes/hujiawei/Users/hujiawei/Android/opencv-2.4.6.1/build"
mkdir build
cd build
cp $OpenCV_DIR/./data/lbpcascades/lbpcascade_frontalface.xml .
cp $OpenCV_DIR/./data/haarcascades/haarcascade_eye.xml .
cp $OpenCV_DIR/./data/haarcascades/haarcascade_eye_tree_eyeglasses.xml .
cmake -G Xcode -D OpenCV_DIR=$OpenCV_DIR ..
```

Eclipse 使用 opencv

1. 建立一个 Mac C++ 工程，包含一个 cpp 文件

2. 右击工程名，选择“Properties”，在属性配置页中选择，点击 C/C++ Build，在下拉选项中选择 Settings。在右边的选项卡中选择 Tool Settings。

3. 在 GCC C++ Compiler 选项列表中选择 Includes，在 Include paths(-I) 中添加安装好的 opencv 的头文件存放目录：

/usr/local/include

/usr/local/include/opencv

/usr/local/include/opencv2

