在Mac OS X下编译安装OpenCV3.1.0（Python2.7）

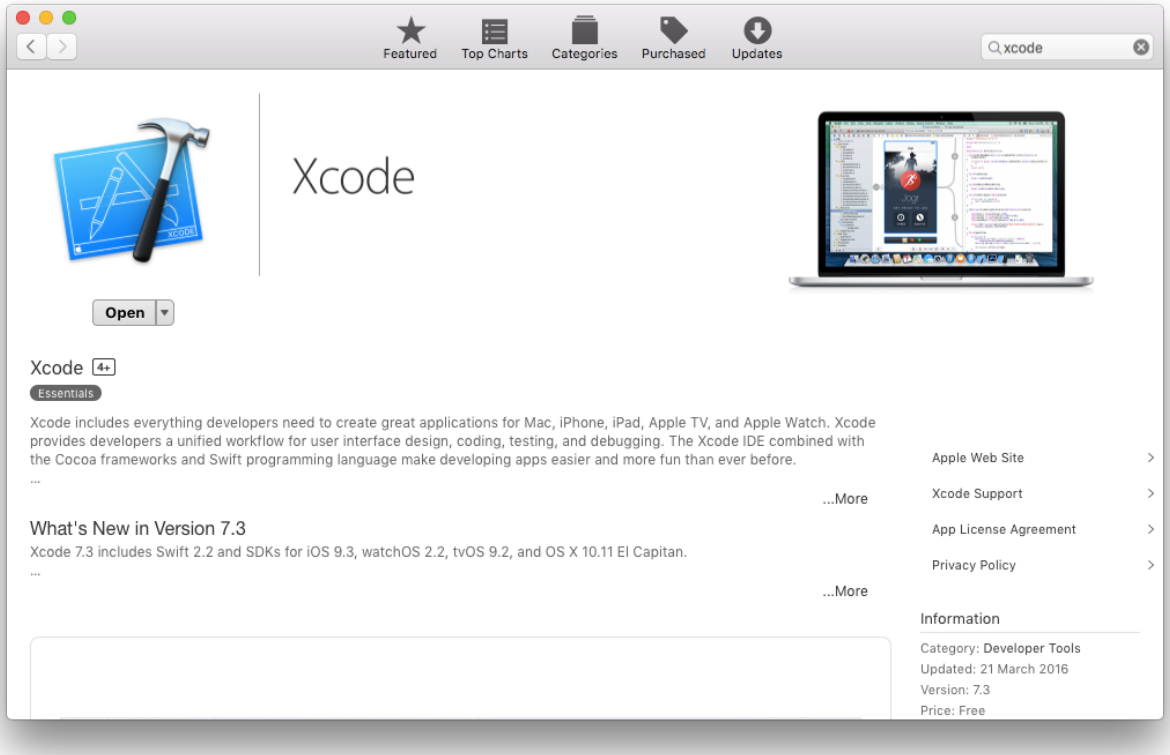
1、安装Homebrew

Homebrew是OS X下一个非常优秀的轻量级包管理工具，其官网为：http://brew.sh/
首先在终端输入以下命令：

```
$ cd ~
$ ruby -e "$(curl -fsSL https://raw.githubusercontent.com/Homebrew/install/master/install)"
$ brew update
```

ruby脚本命令会自动将Homebrew安装至/usr/local目录中，这样在以后任何时候安装新的包的时候都不需要输入sudo命令。同时，在输入brew install <package-name>命令的时候，脚本会自动检测当前已安装的包和新安装包的依赖情况，根据依赖情况下载并安装依赖包

2、安装Xcode



在App Store中搜索Xcode，然后点击安装按钮，需下载大约2G左右的内容，可选Xcode Command Line Tools，建议一并安装Xcode Command Line Tools

3、设置Python

Mac OS X系统自带有Python，版本为2.7，需要为这个版本的Python配置一些依赖库
首先为Python安装pip包管理器，在终端输入以下命令：

```
$ sudo easy_install pip
```

有的系统可能在自带Python的同时一并安装好了pip，这个时候可以对pip进行更新，在终端输入以下命令：

$ sudo pip install --upgrade pip

在更新好pip之后，就开始为Python安装一些OpenCV的依赖库，在终端输入以下命令：

$ sudo pip install --ignore-installed numpy scipy matplotlib

这三个库都是有关科学计算的，其中numpy是必须安装的库，其余两个为可选库，—ignore-installed选项是因为系统自带的Python中有安装过一些其它的库，在用pip安装numpy时，pip检测到某个依赖库的版本过旧会自动卸载，但是OS X的系统机制会对某些目录加以保护，使得使用sudo命令也无法修改目录下的内容

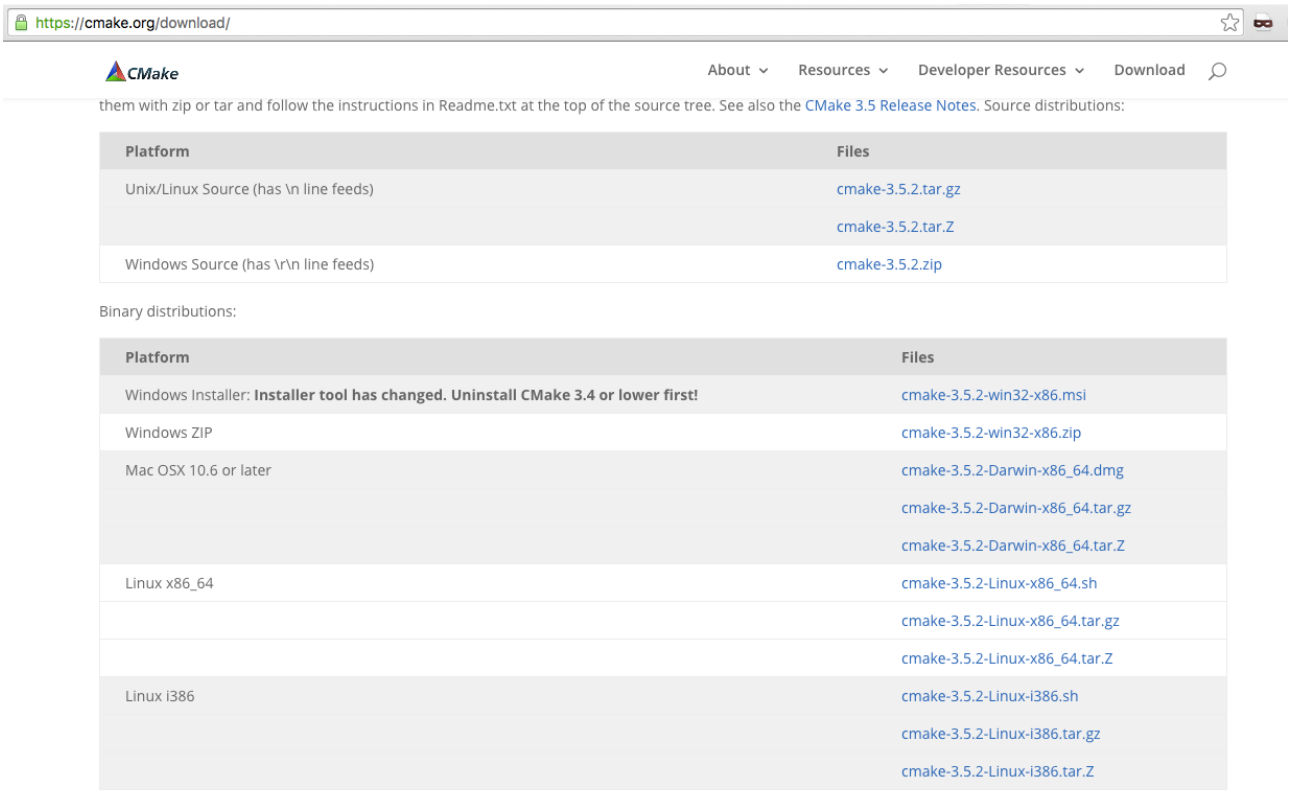
4、安装依赖库

在正式编译安装OpenCV之前，还需要安装一些依赖包和图像处理包，在终端输入以下命令：

$ brew install cmake pkg-config ffmpeg jpeg libpng libtiff eigen tbb gcc git openni libgphoto2 jasper webp

其中cmake包是要用来进行对OpenCV源码编译，其它的都是图像和视频处理库

5、安装Cmake.app（可选）



在cmake官网中的download页面中（https://cmake.org/download/）可以找到cmake.app的安装包下载地址，如上图所示，选择cmake-3.5.2-Darwin-x86_64.dmg或最新版本的dmg文件，下载完后双击运行，将cmake.app拖入到Applications文件中即可

6、下载OpenCV源码

进入OpenCV官网（http://opencv.org/），在右侧Latest Downloads列表里面选择OpenCV for Linux/Mac下载源码，或在终端中输入以下命令：

```
$ cd ~
$ git clone https://github.com/Itseez/opencv.git
```

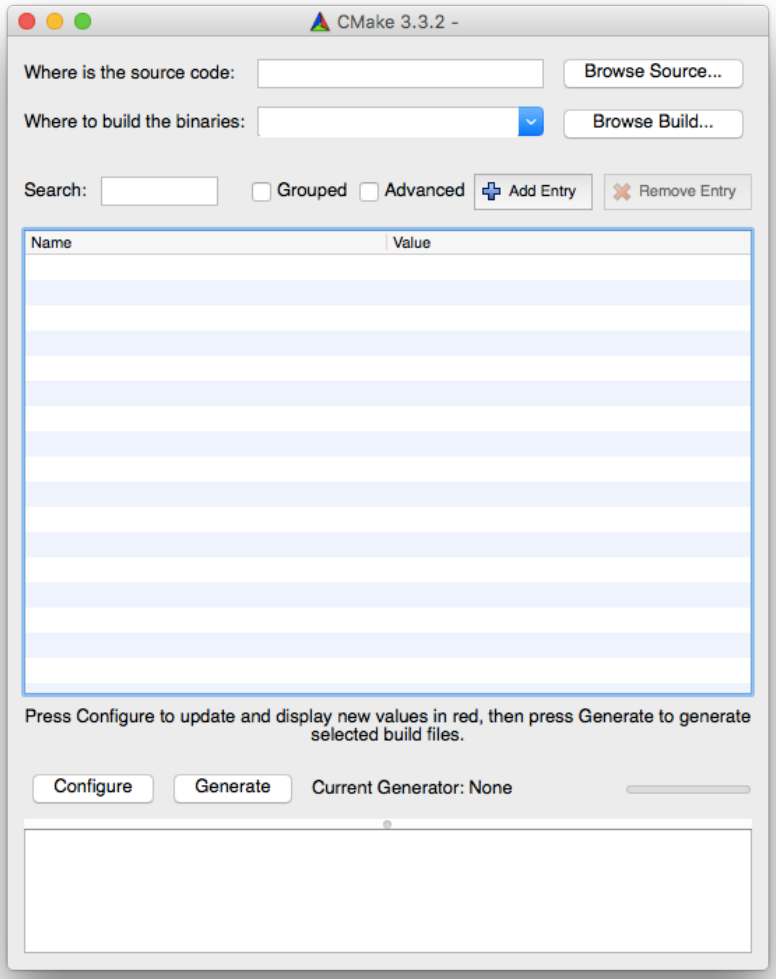
OpenCV3.0以后的版本加入了一系列额外库，在终端输入以下命令下载额外库源码：

```
$ cd ~
$ git clone https://github.com/Itseez/opencv_contrib
```

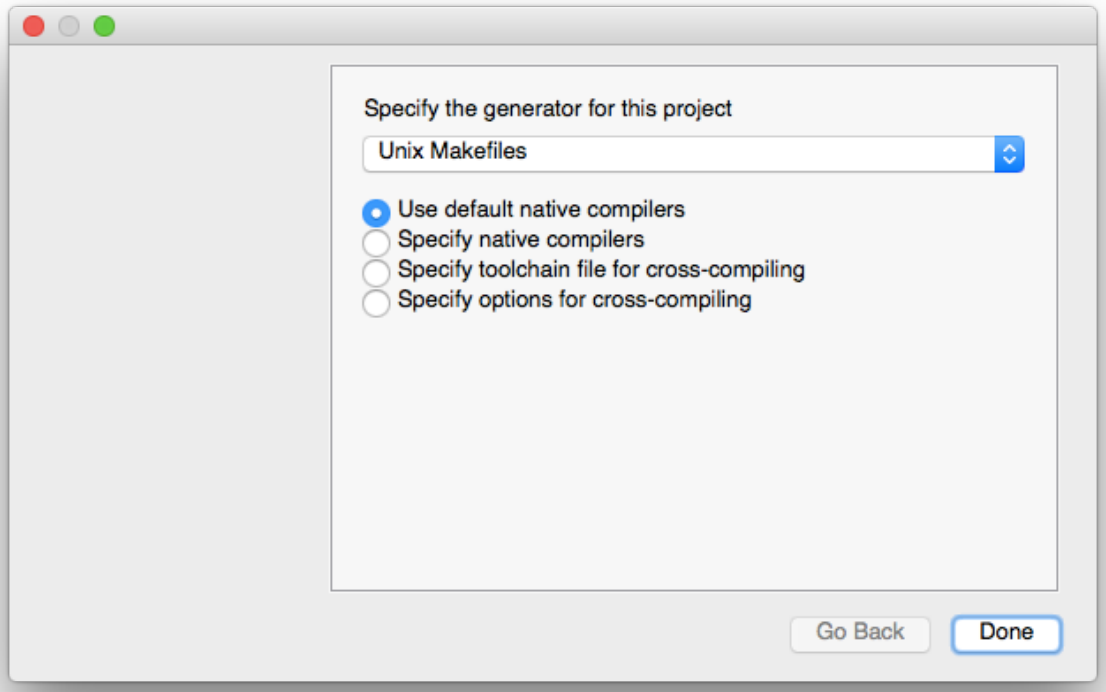
7、编译安装

首先在终端输入以下命令在OpenCV源码所在文件夹内建立一个build目录

```
$ cd ~/opencv
$ mkdir build
```



进入cmake.app，可以看到如上图所示界面。首先点击Browse Source按钮选择OpenCV源码所在文件夹（使用git命令下载的源码是不需要解压的，在官网下载的文件是需要解压的，这里的路径是解压过后的路径），点击Browse Build按钮选择刚才已经建好的build文件夹



点击Configure按钮会弹出如上图所示弹窗，选择Unix Makefiles后点击Done，等待Configure完成后，会显示一系列的选项内容，在这里面做一些细微调整
首先找到OPENCV_EXTRA_MODULES_PATH选项，在里面填入下载解压好的OpenCV额外库的地址，例如：/Users/username/opencv_contrib-master/modules/，再点击Configure按钮，等待Configure完成后，会多出一系列新增选项，这时候再点一遍Configure按钮，让所有标红的选项消失，点击Generate按钮，当下面的显示框内显示Generating done后，就可以关闭cmake.app了

在终端中输入以下命令开始编译安装OpenCV源码：

```
$ cd ~/opencv/build
$ make -j8
$ sudo make install
```

如果没有安装cmake.app，可以在终端中输入以下命令来编译安装：

```
$ cd ~/opencv/build
$ cmake -D CMAKE_BUILD_TYPE=RELEASE -D CMAKE_INSTALL_PREFIX=/usr/local -D
OPENCV_EXTRA_MODULES_PATH=~/opencv_contrib-master/modules ..
```

其中CMAKE_INSTALL_PREFIX是编译之后的安装位置，OPENCV_EXTRA_MODULES_PATH是OpenCV额外库的路径，cmake会自己查找相关依赖库和图像及视频处理接口路径，最终会像下图所示

最后在终端输入以下命令完成安装：

$ make -j8
$ sudo make install

8、修复Python导入cv2报错

当编译安装完OpenCV之后，在Python中导入cv2会报以下错误信息：

Traceback (most recent call last):
File "<stdin>", line 1, in <module>
ImportError: dlopen(/usr/local/lib/python2.7/site-packages/cv2.so, 2): Library not loaded: lib/libopencv_shape.3.1.dylib
Referenced from: /usr/local/lib/python2.7/site-packages/cv2.so
Reason: unsafe use of relative rpath lib/libopencv_shape.3.1.dylib in /usr/local/lib/python2.7/site-packages/cv2.so with restricted binary

这个错误是说dylib中使用了不安全的相对路径，需要手动把dylib中的相对路径改为绝对路径，用下面这个python脚本可以修复，需使用sudo命令：

from os import system

lib_list = ['libopencv_reg.3.1.dylib','libopencv_surface_matching.3.1.dylib','libopencv_dnn.3.1.dylib','libopencv_superres.3.1.dylib','libopencv_xobjdetect.3.1.dylib','libopencv_xphoto.3.1.dylib','libopencv_bgsegm.3.1.dylib','libopencv_bioinspired.3.1.dylib','libopencv_dpm.3.1.dylib','libopencv_line_descriptor.3.1.dylib','libopencv_saliency.3.1.dylib','libopencv_ccalib.3.1.dylib','libopencv_rgbd.3.1.dylib','libopencv_tracking.3.1.dylib','libopencv_videostab.3.1.dylib','libopencv_aruco.3.1.dylib','libopencv_optflow.3.1.dylib','libopencv_stitching.3.1.dylib','libopencv_datasets.3.1.dylib','libopencv_face.3.1.dylib','libopencv_text.3.1.dylib','libopencv_photo.3.1.dylib','libopencv_ximgproc.3.1.dylib','libopencv_objdetect.3.1.dylib','libopencv_xfeatures2d.3.1.dylib','libopencv_shape.3.1.dylib','libopencv_video.3.1.dylib','libopencv_calib3d.3.1.dylib','libopencv_features2d.3.1.dylib','libopencv_flann.

```
3.1.dylib','libopencv_ml.3.1.dylib','libopencv_highgui.3.1.dylib','libopencv_videoio.
3.1.dylib','libopencv_imgcodecs.3.1.dylib','libopencv_imgproc.3.1.dylib','libopencv_core.3.1.dylib']
libs_list = ['/Library/Python/2.7/site-packages/cv2.so','/usr/local/lib/libopencv_reg.3.1.dylib','/usr/local/lib/
libopencv_surface_matching.3.1.dylib','/usr/local/lib/libopencv_dnn.3.1.dylib','/usr/local/lib/
libopencv_superres.3.1.dylib','/usr/local/lib/libopencv_xobjdetect.3.1.dylib','/usr/local/lib/libopencv_xphoto.
3.1.dylib','/usr/local/lib/libopencv_bgsegm.3.1.dylib','/usr/local/lib/libopencv_bioinspired.3.1.dylib','/usr/
local/lib/libopencv_dpm.3.1.dylib','/usr/local/lib/libopencv_line_descriptor.3.1.dylib','/usr/local/lib/
libopencv_saliency.3.1.dylib','/usr/local/lib/libopencv_ccalib.3.1.dylib','/usr/local/lib/libopencv_rgbd.
3.1.dylib','/usr/local/lib/libopencv_tracking.3.1.dylib','/usr/local/lib/libopencv_videostab.3.1.dylib','/usr/local/
lib/libopencv_aruco.3.1.dylib','/usr/local/lib/libopencv_optflow.3.1.dylib','/usr/local/lib/libopencv_stitching.
3.1.dylib','/usr/local/lib/libopencv_datasets.3.1.dylib','/usr/local/lib/libopencv_face.3.1.dylib','/usr/local/lib/
libopencv_text.3.1.dylib','/usr/local/lib/libopencv_photo.3.1.dylib','/usr/local/lib/libopencv_ximgproc.
3.1.dylib','/usr/local/lib/libopencv_objdetect.3.1.dylib','/usr/local/lib/libopencv_xfeatures2d.3.1.dylib','/usr/
local/lib/libopencv_shape.3.1.dylib','/usr/local/lib/libopencv_video.3.1.dylib','/usr/local/lib/
libopencv_calib3d.3.1.dylib','/usr/local/lib/libopencv_features2d.3.1.dylib','/usr/local/lib/libopencv_flann.
3.1.dylib','/usr/local/lib/libopencv_ml.3.1.dylib','/usr/local/lib/libopencv_highgui.3.1.dylib','/usr/local/lib/
libopencv_videoio.3.1.dylib','/usr/local/lib/libopencv_imgcodecs.3.1.dylib','/usr/local/lib/libopencv_imgproc.
3.1.dylib','/usr/local/lib/libopencv_core.3.1.dylib']
cmd = 'install_name_tool -change '
path = '/usr/local/lib/'

for item in lib_list:
        for lib in libs_list:
                system(cmd + 'lib/' + item + ' ' + path + item + ' ' + lib)
```

这样就可以在Python中使用OpenCV了，到此OpenCV就完全安装好了