

Mac OS X 10.10 Yosemite 在 MATLAB R2014b 上配置openCV

1.安装command line tools。Xcode-select。可以先在terminal中输入

`xcode-select -p`，如果没报错则可跳过这一步。

(1) 在terminal输入：

`xcode-select -install`

(2) Click “Install” to download and install Xcode Command Line Tools

(3) 确认安装成功，在terminal输入：

`xcode-select -p`

出现/Library/Developer/CommandLineTools即可。

(4) 确认gcc已被安装，在terminal输入：

`gcc --version`

出现类似以下信息即可。

Configured with: --prefix=/Library/Developer/CommandLineTools/usr --with-gxx-include-dir=/usr/include/c++/4.2.1

Apple LLVM version 6.0 (clang-600.0.54) (based on LLVM 3.5svn)

Target: x86_64-apple-darwin14.0.0

Thread model: posix

2.安装好pkg-config。

在terminal检查是否装有pkg-config：`which pkg-config`

如果没有在terminal中输入：

```
curl http://pkgconfig.freedesktop.org/releases/pkg-config-0.28.tar.gz -o  
pkgconfig.tgz
```

```
tar -zxf pkgconfig.tgz && cd pkg-config-0.28
```

```
./configure && make install
```

可能会碰到这个错误：

configure: error: Either a previously installed pkg-config or "glib-2.0 >= 2.16" could not be found. Please set GLIB_CFLAGS and GLIB_LIBS to the correct values or pass --with-internal-glib to configure to use the bundled copy.

再在terminal中输入：

```
./configure --with-internal-glib && make install
```

再检查一下确认即可。

3.安装opencv, 三种方法, 任选一种都行。第一种最简单, 不行就试下第三种, 别用第二种。

(1) 下载homebrew并且用homebrew安装opencv。在terminal输入以下命令行：

```
brew tap homebrew/science
```

```
brew install opencv
```

找到opencv的安装路径确认在/usr/local/Cellar/opencv/2.4.11_1/include中有opencv和opencv2两个文件夹。找到opencv和opencv2继续。这个时候用xcode已经可以直接使用opencv。

如果想要测试可以按照以下步骤尝试：

1)在xcode中新建工程，同时将测试程序复制到main.cpp中。测试程序附在最后。

2)将/usr/local/Cellar/opencv/2.4.11_1/lib中的所有后缀为dylib的文件加入到工程中。

3)将build settings中的Header Search Paths修改为/usr/local/Cellar/opencv/2.4.11_1/
include /usr/local/Cellar/opencv/2.4.11_1/include/opencv

4)将下面的Library Search Paths 修改为/usr/local/Cellar/opencv/2.4.11_1/lib

别用这种！(2)直接下载cmake安装。 (<http://www.cmake.org/download/>)

进入opencv-2.4.11路径

```
cd ~/Downloads/OpenCV-2.4.11
```

- mkdir release

直接打开cmake。

(记得改用户名):

Where is the source code: `/Users/yueyangzou/Downloads/OpenCV-2.4.11`

Where to build the binaries: /Users/yueyangzou/Downloads/OpenCV-2.4.11/release

- click 'Configure' [and choose 'Unix Makefiles' and 'Use default native compilers']

- cmake screen will be red

- add 'BUILD EXAMPLES'

```
add 'INSTALL C EXAMPLES'
```

```
delete 'WITH CUDA'
```

- click 'Configure' again

- cmake screen should now be clear:

if not, read diagnostics and delete appropriate packages

- click 'Generate'

- exit cmake

cd release

make

```
sudo make install
```

这种方法opencv安装路径我没找到，可能是没安装成功。所以我最后还是用了第一种。

(3)在terminal中输入：

```
brew install cmake
```

安装成功后在terminal中输入：

```
cd downloads
cd opencv-2.4.11
mkdir release
cd release
cmake -G "Unix Makefiles" ..
make -j8
sudo make install
```

用这种方法opencv的路径在usr/local/lib和usr/local/include下面，用这种方法用xcode跑测试的时候需要改下路径，其他没什么变化。

4.用新建opencv.pc的方式保证pkg-config可以找到opencv的路径。（参考<http://prateekvjoshi.com/2013/10/18/package-opencv-not-found-lets-find-it/>）

前往 /usr/local/lib/pkgconfig文件夹，创建opencv.pc，内容如下：

```
prefix=/usr
exec_prefix=${prefix}
includedir=${prefix}/include
libdir=${exec_prefix}/lib

Name: opencv
Description: The opencv library
Version: 2.x.x
Cflags: -I${includedir}/opencv -I${includedir}/opencv2
Libs: -L${libdir} -lopencv_calib3d -lopencv_imgproc -lopencv_contrib -lopencv_legacy -l
```

在terminal中输入：`pkg-config --cflags --libs opencv`。

返回opencv的正确路径/usr/local/Cellar/opencv/2.4.11_1，说明opencv安装成功，而且pkg-config可以找到opencv的路径。

5.在github下载mexopencv。2.4版本的，不要下错3.0的。网址是<https://github.com/kyamagu/mexopencv/tree/v2.4>。

6.前往/Applications/MATLAB_R2014b.app/bin/maci64。新建libopencv文件夹。将maci64中文件名为libopencv_的文件剪切到libopencv文件夹中。保证matlab不会调用它自己的opencv函数。

7.跟着eric cristofalo网站修改/Applications/MATLAB_R2014b.app/toolbox/local中的startupsav.m, 改名为startup.m, 删掉matlab.mat, 添加下面两条路径。
`addpath(genpath('/Users/yueyangzou/downloads/mexopencv-2.4'));` //username记得改。
`setenv('PATH', [getenv('PATH') ':/usr/local/bin']);` //这是添加pkg-config的路径, 如果有问题可以去/usr/local/bin确认下。

8.修改mexopencv.sh。

(1)生成配置文件。打开matlab, 输入:

```
mex -setup
```

(2)修改配置文件。

在matlab中输入:

```
edit ~/.matlab/R2014b/mexopts.sh
```

或者在matlab中输入:

```
cd (matlabroot)
```

```
cd bin
```

```
edit mexopts.sh
```

或者直接前往/Applications/MATLAB_R2014b.app/bin用xcode打开mexopts.sh进行如下编辑:

将所有 10.8 替换成 10.10

将所有 -lstdc++ 替换成 -lc++

将 MW_SDKROOT_TMP="findxcode-select -print-path-name MacOSX10.7.sdk" 改成 MW_SDKROOT_TMP="/Applications/Xcode.app/Contents/Developer/Platforms/MacOSX.platform/Developer/SDKs/MacOSX10.10.sdk/"

在 CFLAGS="\$CFLAGS -fexceptions" 后一行添加 CFLAGS="\$CFLAGS -Dchar16_t=uint16_t"

在 CXXFLAGS="-fno-common -fexceptions -arch \$ARCHS -isysroot \$MW_SDKROOT -mmacosx-version-min=\$MACOSX_DEPLOYMENT_TARGET" 后一行添加 CXXFLAGS="\$CXXFLAGS -std=c++11 -stdlib=libc++"

将 `MW_SDKROOT='$MW_SDKROOT_TMP'` 改成 `MW_SDKROOT='/Applications/Xcode.app/Contents/Developer/Platforms/MacOSX.platform/Developer/SDKs/MacOSX10.10.sdk/'`

将 `MLIBS="-L$TMW_ROOT/bin/$Arch -lmx -lmex -lmat"` 改成 `MLIBS="$TMW_ROOT/bin/$Arch/libmx.dylib $TMW_ROOT/bin/$Arch/libmex.dylib $TMW_ROOT/bin/$Arch/libmat.dylib"`

如果是在 `/Applications/MATLAB_R2014b.app/bin` 中编辑的（后两种方法），编辑完之后把这个文件复制到 `Users/yueyangzou/.matlab/R2014b/` 中（这里记得改Username）。

(3) 再在matlab中输入：

```
mex -setup
```

依次输入 '1' 和 'y'。

9. 编译出mexfile。在terminal进入mexopencv文件夹 (`cd downloads/mexopencv-2.4`)，输入：

```
make MATLABDIR=/Applications/MATLAB_R2014b.app clean
```

```
make MATLABDIR=/Applications/MATLAB_R2014b.app
```

这个时候查看一下mexopencv中的+cv，是否生成了mex文件。生成即编译成功。

10. 重启matlab，测试mexopencv的样本程序，会发现如下错误：

```
Invalid MEX-file '/Users/dgolden/software/cpp/mexopencv/+cv/Canny.mexmaci64':
dlopen(/Users/dgolden/software/cpp/mexopencv/+cv/Canny.mexmaci64, 6): Library
not loaded:
```

```
/opt/local/lib/libtiff.5.dylib
```

```
Referenced from: /opt/local/lib/libopencv_highgui.2.4.dylib
```

```
Reason: Incompatible library version: libopencv_highgui.2.4.dylib requires version
8.0.0 or later, but libtiff.5.dylib provides version 6.0.0
```

需要在matlab中修改invalid mexfile的错误。两种方法，任选一种。

(1) 找到 `/Applications/MATLAB_R2014b.app/bin/maci64` 中的冲突文件，改名字或者移走。我把libtiff.5.dylib修改为libtiff.5.bak.dylib in Applications/MATLAB_R2014b.app/bin/maci64。不过可能会有其他问题如果其他matlab函数用了这个，但是方便。

(2) 关闭matlab，然后在terminal中打开matlab，在terminal中输入matlab的路径直接打开：

/Applications/MATLAB_R2014b.app/bin/matlab

打开之后输入：

DYLD_INSERT_LIBRARIES=/opt/local/lib/libtiff.5.dylib /Applications/MATLAB_R2014b.app/bin/matlab & (对照error中libtiff.5.dylib, 修改libtiff.5.dylib之前的路径即可)

Xcode测试程序：

```
#include <iostream>

#include <opencv2/opencv.hpp>
#include <opencv2/highgui/highgui.hpp>
#include <opencv/cvaux.hpp>
#include <fstream>
using namespace std;

#define BYTE unsigned char

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

{
    // insert code here...
    #if 1
        //get the image from the directed path
        IplImage* img = cvLoadImage("/Users/yueyangzou/1.jpg", 1); //这里记得改用户名。
        NSLog(img);
        //create a window to display the image
        cvNamedWindow("picture", 1);
        //show the image in the window
        cvShowImage("picture", img);
        //wait for the user to hit a key
        cvWaitKey(0);
        //delete the image and window
        cvReleaseImage(&img);
        cvDestroyWindow("picture");
    #endif
    //return
    return 0;
}
```