**摘自网站：**[**http://blog.csdn.net/surgewong/article/details/39078251**](http://blog.csdn.net/surgewong/article/details/39078251)

**Opencv 2.4.9在Ubuntu下的配置与安装**

[surgewong@gmail.com](mailto:surgewong@gmail.com)

<http://blog.csdn.net/surgewong>

      每次学习新东西或者换新的电脑，最让人头疼的就是环境的安装了。网上有一大堆教程，不过总地来讲，比较混乱，有些甚至有误导性，当然也不乏一些有质量的博客。为了方便自己以后重复安装以及能够帮助到有需要的朋友，在此将相关有用的博客整理一下。

      这里主要讲一些如何在Ubuntu（12.04）下安装Opencv（2.4.9）。Opencv的源码可以在其[官网](http://opencv.org/)下载，也可以在[百度网盘](http://pan.baidu.com/s/1mgJXiNe)下载。 Opencv的最近的一些版本都是使用cmake来编译的，其配置过程基本上是一样的。

**一、依赖包安装**

       Ubuntu下安装软件或多或少都需要安装一些**依赖包（软件）**。opencv也不例外。

       1.  build-essential 软件包，为编译程序提供必需软件包的列表信息，这样软件包才知道头文件、库函数在哪里。还会下载依赖的软件包，安装gcc/g++/gdb/make 等基本编程工具，最后才组成一个开发环境。

**[plain]** [view plaincopy](http://blog.csdn.net/surgewong/article/details/39078251)

1. sudo apt-get install build-essential

       2. 下面依赖关系主要是为了能够支持读写图片以及视频等。

**[plain]** [view plaincopy](http://blog.csdn.net/surgewong/article/details/39078251)

1. sudo apt-get install  libgtk2.0-dev libavcodec-dev libavformat-dev libjpeg62-dev libtiff4-dev  libswscale-dev libjasper-dev

       3. 安装cmake，可以[官网](http://www.cmake.org/)上下载源码编译，也可以在[百度网盘](http://pan.baidu.com/s/1bns5gRl)下载2.8.12版本，当然可以直接使用apt-get方法。

**[plain]** [view plaincopy](http://blog.csdn.net/surgewong/article/details/39078251)

1. sudo apt-get install cmake

      4. 安装pkg-config，它是一个提供从源代码中编译软件时查询已安装的库时使用的统一接口的计算机软件。

**[plain]** [view plaincopy](http://blog.csdn.net/surgewong/article/details/39078251)

1. sudo apt-get install pkg-config

**二、编译安装过程**

       1. 使用cmake来检测编译环境以及生成makefile以及其他project信息，在使用cmake过程可以设置一些参数。

         /\*  没用到 在本目录下使用cmake，注意后面的点表示在当前目录下寻找CMakeLists.txt。

**[plain]** [view plaincopy](http://blog.csdn.net/surgewong/article/details/39078251)

1. cmake .  \*/

            也可以在可以在另外建一个文件夹，最常见的就是子目录。当然在构建的时候也可以配置一些参数：

**[plain]** [view plaincopy](http://blog.csdn.net/surgewong/article/details/39078251)

1. cd opencv-2.4.9
2. mkdir release
3. cd release
4. cmake -D CMAKE\_BUILD\_TYPE=RELEASE -D CMAKE\_INSTALL\_PREFIX=/usr/local ..

       2. 使用make编译

**[plain]** [view plaincopy](http://blog.csdn.net/surgewong/article/details/39078251)

1. make

       3.  使用make install 安装

**[plain]** [view plaincopy](http://blog.csdn.net/surgewong/article/details/39078251)

1. sudo make install

       4.  上面的过程基本上完成了opencv的安装。但是并不能够马上用它，opencv还包含很多动态链接库。你可以在文件/etc/ld.so.conf中添加 /usr/local/lib（这个跟安装目录有关， {CMAKE\_INSTALL\_PREFIX}/lib），也可以在/etc/ld.so.conf.d 目录下增加一个conf文件（可以命名为 opencv.conf），同样添加 /usr/local/lib

http://img.blog.csdn.net/20140905125942990?watermark/2/text/aHR0cDovL2Jsb2cuY3Nkbi5uZXQvc3VyZ2V3b25n/font/5a6L5L2T/fontsize/400/fill/I0JBQkFCMA==/dissolve/70/gravity/SouthEast

创建opencv.conf的方法是在命令行输入 vi opencv.conf

I 输入图片程序

输入完成后按ESC取消光标

：wq 保存并退出

           使用下面的动态库管理命令ldconfig，让opencv的相关链接库被系统共享

**[plain]** [view plaincopy](http://blog.csdn.net/surgewong/article/details/39078251)

1. sudo ldconfig -v

      5.  完成了动态库的共享，还不能使用opencv来编程，还需要为程序指定openvc的头文件位置。这里使用pkg-config命令来完成。首先在 /etc/profile 中添加

在ETC文件夹中找到profile文件，然后输入vi profile 进行修改，添加

**[plain]** [view plaincopy](http://blog.csdn.net/surgewong/article/details/39078251)

1. export  PKG\_CONFIG\_PATH=$PKG\_CONFIG\_PATH:/usr/local/lib/pkgconfig

http://img.blog.csdn.net/20140905131540015?watermark/2/text/aHR0cDovL2Jsb2cuY3Nkbi5uZXQvc3VyZ2V3b25n/font/5a6L5L2T/fontsize/400/fill/I0JBQkFCMA==/dissolve/70/gravity/SouthEast

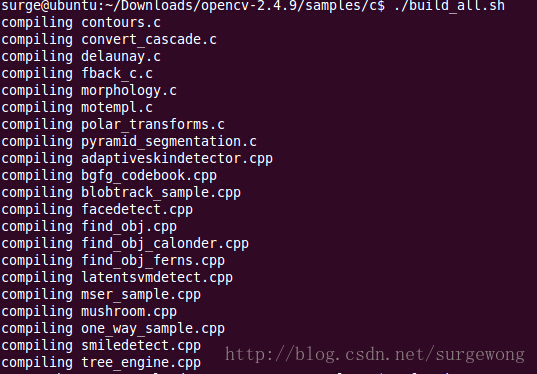
三、测试

       1. 在解压的opencv目录下找到 samples/c/build\_all.sh，运行该文件

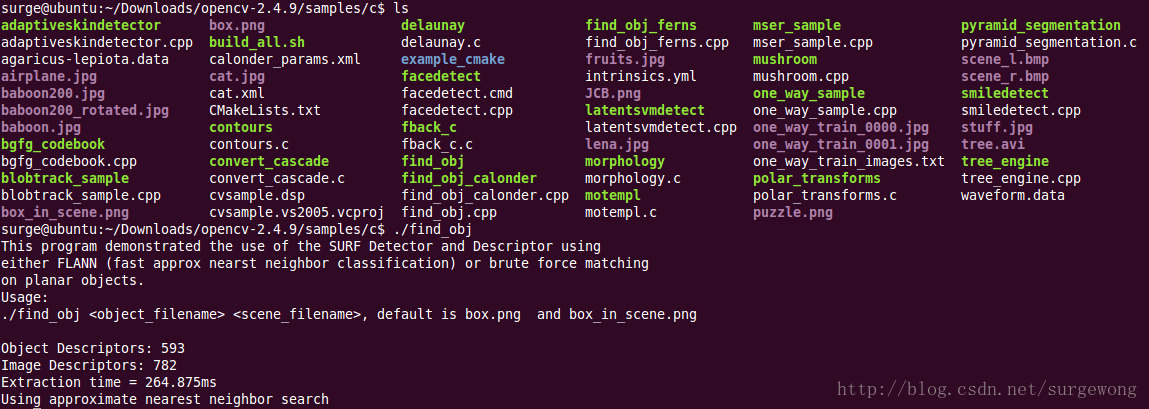
**[plain]** [view plaincopy](http://blog.csdn.net/surgewong/article/details/39078251)

1. ./build\_all.sh

注意：\*\*\*\*\*\*在命令行输入 sh ./build\_all.sh而不是像上面说的



          2. 找其中一个来运行看一下效果，例如 find\_obj程序：





       至此，关于opencv在Ubuntu下的安装成功了，尽情地遨游在计算机视觉的世界里吧！

# 在Ubuntu上安装qt

在Ubuntu的浏览器中输入qt的网站：

[www.qt-project.org](http://www.qt-project.org)

下载最新版本的qt-opensource-Linux-x86-5.3.2.run（471M）



在命令行中输入：

Chmod +x ./qt-opensource-linux-x86-5.3.2.run

Sudo ./ qt-opensource-linux-x86-5.3.2.run

安装以后就可以运行QT了

[复制代码](javascript:void(0);)

#include <QCoreApplication>

#include <opencv2/core/core.hpp>

#include <opencv2/highgui/highgui.hpp>

#include <opencv2/imgproc/imgproc.hpp>

using namespace cv;

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

{

// QCoreApplication a(argc, argv);

namedWindow( "src", WINDOW\_AUTOSIZE );

Mat src1 = imread( "lena.jpg" );

while(1)

{

imshow( "src", src1 );

char c = waitKey(30);

if( 27==c )

return 0;

}

// return a.exec();

}

[复制代码](javascript:void(0);)

注意：

1.  因为这里使用的是Qt控制台程序，所以main函数的第一句 ：

QCoreApplication a(argc, argv);

和最后一句：

return a.exec();

不能使用，否则运行时会报错。

2.  另外需要注意一点的是，需要将opencv的头文件和库文件与Qt关联起来，所以在双击QtCreator工程下到\*\*\*.pro文件，在其后面添加下列代码：

[复制代码](javascript:void(0);)

INCLUDEPATH += /usr/local/include \

/usr/local/include/opencv \

/usr/local/include/opencv2

LIBS += /usr/local/lib/libopencv\_highgui.so \

/usr/local/lib/libopencv\_core.so \

/usr/local/lib/libopencv\_imgproc.so

[复制代码](javascript:void(0);)

即此时对应的整个\*\*\*.pro文件内容如下：

[复制代码](javascript:void(0);)

#-------------------------------------------------

#

# Project created by QtCreator 2012-07-09T22:01:34

#

#-------------------------------------------------

QT += core

QT -= gui

TARGET = test1

CONFIG += console

CONFIG -= app\_bundle

TEMPLATE = app

SOURCES += main.cpp

INCLUDEPATH += /usr/local/include \

/usr/local/include/opencv \

/usr/local/include/opencv2

LIBS += /usr/local/lib/libopencv\_highgui.so \

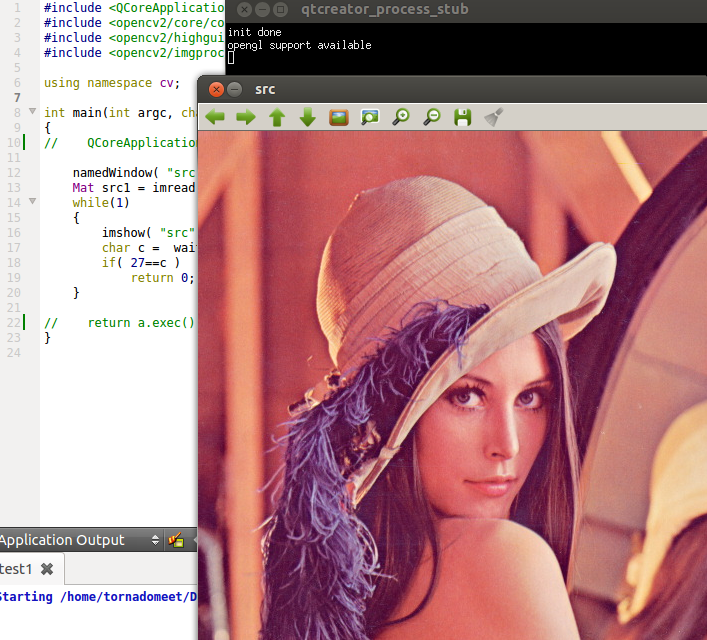
/usr/local/lib/libopencv\_core.so \

/usr/local/lib/libopencv\_imgproc.so

[复制代码](javascript:void(0);)

3.  第一个实验需要提前把图片，我这里用的是lena.jpg放入\*\*\*-build-desktop-Qt\_4\_8\_2\_in\_PATH\_\_Qt-4\_8\_2\_\_Release文件夹中，因为我的工程建立在桌面上，所以其默认的名字比较长。不放入这个文件夹或者放错文件夹则运行时会出现错误的。

其运行结果如下：



最后的最后附上两个参考的网站，以上教程均引自上面的网站：

<http://www.cnblogs.com/tornadomeet/archive/2012/07/10/2584702.html>

这是linux下opencv+qt的

<http://blog.csdn.net/surgewong/article/details/39078251>

这是在Ubuntu中配置opencv的