





```
[plain]
1. >>> import cv2
2. >>> cv2.namedWindow('test',cv2.WINDOW_AUTOSIZE)
3. >>>
```

如果没有报任何错误就代表gtk也是正常的

如如果出现下面提示

```
[plain]
1. OpenCV Error: Unspecified error (The function is not implemented. Rebuild the library with Wind
2. GTK+ 2.x or Carbon support. If you are on Ubuntu or Debian, install libgtk2.0-dev and pkg-
config,
3. then re-run cmake or configure script) in cvNamedWindow,
```

那就代表opencv不支持gtk

如果上面两个测试都是好使的你可以跳过本步骤

首先我发现我不论如何重新编译opencv发现python一直都是有问题，  
最後发现是因为不知道什么时候已经装过 python-opencv的包了  
可以用命令

```
[plain]
1. $ pip list |grep opencv
```

如果发现有任何跟opencv有关的包都可以利用 pip uninstall xxx 来移除  
假设看到 pythom-opencv的包那就 pip uninstall opencv-python 来移除

还有检查dpkg -i |grep opencv 如果跟python的opencv有关的包也得移除

```
[plain]
1. $ sudo apt-get remove python-opencv
```

然後到你之前安装opencv3.2的那个源码目录，

```
[plain]
1. $ cd build
2. $ make uninstall
```

这样就会卸载之前安装的opencv

再来V4L的头文件已经改名了，但是opencv会默认使用linux/videodev.h所以要做个软链接

```
[plain]
1. $ ln -s /usr/include/libv4l1-videodev.h /usr/include/linux/videodev.h
```

opencv安装过程中会自动的检测相关的包，以及一些依赖，  
先列出我安装的包，但是因为环境多少有点不同，下面会教大家如何看缺少的包

```
[plain]
1. $ sudo apt-get install libgphoto2-dev v4l2ucp libv4l-dev dv4l libwebcam0-dev libgtkglext1-
dev libunicap2-dev
```



1

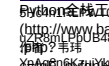


固态硬盘快多少

在线课程



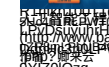
(http://www.baidu.com/cb.php?c=lgF\_pyfqHmknjsnjD0IZ0qnfK9ujYzP1mznWR10Aw-



(http://www.baidu.com/cb.php?c=lgF\_pyfqHmknjsnjD0IZ0qnfK9ujYzP1mznWR10Aw-



(http://www.baidu.com/cb.php?c=lgF\_pyfqHmknjsnjD0IZ0qnfK9ujYzP1mznWR10Aw-



(http://www.baidu.com/cb.php?c=lgF\_pyfqHmknjsnjD0IZ0qnfK9ujYzP1mznWR10Aw-



(http://www.baidu.com/cb.php?c=lgF\_pyfqHmknjsnjD0IZ0qnfK9ujYzP1mznWR10Aw-



(http://www.baidu.com/cb.php?c=lgF\_pyfqHmknjsnjD0IZ0qnfK9ujYzP1mznWR10Aw-



(http://www.baidu.com/cb.php?c=lgF\_pyfqHmknjsnjD0IZ0qnfK9ujYzP1mznWR10Aw-



比固态硬盘还快

weixin\_36441117/article/details/7281119

6)

735

IIC专题(三)-smart210裸机透过IIC操作板  
载AT24LC04 (http://blog.csdn.net/weixin\_36441117/article/details/52870087)

685

安装NVIDIA显卡驱动+CUDA+CUDNN+N  
VCAFFE的问题总结 (http://blog.csdn.net/weixin\_36441117/article/details/7686448

1)

655

再来执行cmake

```
[plain]
1. $ cmake -D CMAKE_BUILD_TYPE=Release -D CMAKE_INSTALL_PREFIX=/usr/local \
2. -D WITH_TBB=ON -D WITH_V4L=ON -D WITH_GTK=ON -D WITH_GTK_2_X -D WITH_OPENGL=ON \
3. -D WITH_CUDA=ON -D WITH_CUBLAS=ON -D BUILD_PYTHON_SUPPORT=ON \
4. -D OPENCV_EXTRA_MODULES_PATH=../opencv_contrib-3.2.0/modules/ ../opencv-3.2.0
```

其中如果V4L2使用share库也可以把-D WITH\_V4L: = ON 换成

```
[plain]
1. -D WITH_LIBV4L=ON
```

在cmake的过程中会有类似如下的提示

```
[plain]
1. Detected version of GNU GCC: 54 (504)
2. FP16: Feature disabled

3. Found OpenEXR: /usr/lib/x86_64-linux-gnu/libIlmImf.so
4. Checking for module 'libcil'
5. No package 'libcil' found
6. Looking for linux/videodev.h
7. Looking for linux/videodev.h - found
8. Looking for linux/videodev2.h
9. Looking for linux/videodev2.h - found
10. Looking for sys/videoio.h
11. Looking for sys/videoio.h - not found
12. Checking for module 'libavresample'
13. No package 'libavresample' found
14. Found TBB: build
15. found IPP (ICV version): 9.0.1 [9.0.1]
16. at: /mnt/sdb/ubuntu/install/opencv/build/3rdparty/ippicv/ippicv_linux
17. CUDA detected: 8.0
18. CUDA NVCC target flags: -gencode;arch=compute_20,code=sm_20;-
gencode;arch=compute_30,code=sm_30;-gencode;arch=compute_35,code=sm_35;-
gencode;arch=compute_37,code=sm_37;-gencode;arch=compute_50,code=sm_50;-
gencode;arch=compute_52,code=sm_52;-gencode;arch=compute_60,code=sm_60;-
gencode;arch=compute_61,code=sm_61;-D_FORCE_INLINES
19. LAPACK_IMPL: Atlas, LAPACK_LIBRARIES: /usr/lib/liblapack.so;/usr/lib/libcblas.so;/usr/lib/libat
20. Could NOT find JNI (missing: JAVA_AWT_LIBRARY JAVA_JVM_LIBRARY JAVA_INCLUDE_PATH JAVA_INCLUDE_L
21. Could NOT find Matlab (missing: MATLAB_MEX_SCRIPT MATLAB_INCLUDE_DIRS MATLAB_ROOT_DIR MATLAB_L
22. VTK is not found. Please set -
D_VTK_DIR in CMake to VTK build directory, or to VTK install subdirectory with VTKConfig.cmake f
23.
24. General configuration for OpenCV 3.2.0 =====
25. Version control: unknown
26.
27. Platform:
28. Timestamp: 2017-08-14T00:31:55Z
29. Host: Linux 4.10.0-30-generic x86_64
30. CMake: 3.5.1
31. CMake generator: Unix Makefiles
32. CMake build tool: /usr/bin/make
33. Configuration: Release
34.
35. C/C++:
36. Built as dynamic libs?: YES
37. C++ Compiler: /usr/bin/c++ (ver 5.4.0)
38. C++ flags (Release): -fsigned-char -W -Wall -Werror=return-type -Werror=non-
virtual-dtor -Werror=address -Werror=sequence-point -Wformat -Werror=format-security -
Wmissing-declarations -Wundef -Winit-self -Wpointer-arith -Wshadow -Wsign-promo -Wno-
narrowing -Wno-delete-non-virtual-dtor -Wno-comment -fdiagnostics-show-option -Wno-long-long -
pthread -fomit-frame-pointer -msse -msse2 -mno-avx -msse3 -mno-ssse3 -mno-sse4.1 -mno-sse4.2 -
```



内容举报

返回顶部

内容举报

```
ffunction-sections -fvisibility=hidden -fvisibility-inlines-hidden -O3 -DNDEBUG -DNDEBUG
39. C++ flags (Debug): -fsigned-char -W -Wall -Werror=return-type -Werror=non-
virtual-dtor -Werror=address -Werror=sequence-point -Wformat -Werror=format-security -
Wmissing-declarations -Wundef -Winit-self -Wpointer-arith -Wshadow -Wsign-promo -Wno-
narrowing -Wno-delete-non-virtual-dtor -Wno-comment -fdiagnostics-show-option -Wno-long-long -
pthread -fomit-frame-pointer -msse -msse2 -mno-avx -msse3 -mno-ssse3 -mno-sse4.1 -mno-sse4.2 -
ffunction-sections -fvisibility=hidden -fvisibility-inlines-hidden -g -O0 -DDEBUG -D_DEBUG
40. C Compiler: /usr/bin/cc
41. C flags (Release): -fsigned-char -W -Wall -Werror=return-type -Werror=non-
virtual-dtor -Werror=address -Werror=sequence-point -Wformat -Werror=format-security -
Wmissing-declarations -Wmissing-prototypes -Wstrict-prototypes -Wundef -Winit-self -Wpointer-
arith -Wshadow -Wno-narrowing -Wno-comment -fdiagnostics-show-option -Wno-long-long -pthread -
fomit-frame-pointer -msse -msse2 -mno-avx -msse3 -mno-ssse3 -mno-sse4.1 -mno-sse4.2 -
ffunction-sections -fvisibility=hidden -O3 -DNDEBUG -DNDEBUG
42. C flags (Debug): -fsigned-char -W -Wall -Werror=return-type -Werror=non-
virtual-dtor -Werror=address -Werror=sequence-point -Wformat -Werror=format-security -
Wmissing-declarations -Wmissing-prototypes -Wstrict-prototypes -Wundef -Winit-self -Wpointer-
arith -Wshadow -Wno-narrowing -Wno-comment -fdiagnostics-show-option -Wno-long-long -pthread -
fomit-frame-pointer -msse -msse2 -mno-avx -msse3 -mno-ssse3 -mno-sse4.1 -mno-sse4.2 -
ffunction-sections -fvisibility=hidden -g -O0 -DDEBUG -D_DEBUG
43. Linker flags (Release):
44. Linker flags (Debug):
45. ccache: NO
46. Precompiled headers: YES
47. Extra dependencies: /usr/lib/x86_64-linux-gnu/libpng.so /usr/lib/x86_64-linux-
gnu/libtiff.so /usr/lib/x86_64-linux-gnu/libjasper.so /usr/lib/x86_64-linux-
gnu/liblmath.so /usr/lib/x86_64-linux-gnu/liblImf.so /usr/lib/x86_64-linux-
gnu/libIex.so /usr/lib/x86_64-linux-gnu/libHalf.so /usr/lib/x86_64-linux-
gnu/libIlmThread.so gtk-x11-2.0 gdk-x11-2.0 pangocairo-1.0 atk-1.0 cairo gdk_pixbuf-2.0 gio-
2.0 pangot2-1.0 pango-1.0 gobject-2.0 glib-2.0 fontconfig freetype gthread-2.0 gtkglext-x11-
1.0 gdkglext-x11-1.0 GLU GL Xmu Xt SM ICE pangox-1.0 X11 gmodule-2.0 gstreamer-
1.0 gstvideo-1.0 gstapp-1.0 gstriff-1.0 gstreamer-1.0 unicap dc1394 avcodec-ffmpeg avformat-
ffmpeg avutil-ffmpeg swscale-ffmpeg gphoto2 gphoto2_port exif dl m pthread rt /usr/lib/x86_64-
linux-gnu/libGLU.so /usr/lib/x86_64-linux-
gnu/libGL.so /usr/lib/liblapack.so /usr/lib/libblas.so /usr/lib/libatlas.so cudart nppc nppi n
L/usr/local/cuda-8.0/lib64
48. 3rdparty dependencies: libwebp zlib libjpeg tbb
49.
50. OpenCV modules:
51. To be built: cudev core cudaarithm flann imgproc ml video cudabgsegm cudaf1
52. Disabled: world
53. Disabled by dependency: -
54. Unavailable: java python3 viz
55.
56. GUI:
57. QT: NO
58. GTK+ 2.x: YES (ver 2.24.30)
59. GThread : YES (ver 2.48.2)
60. GtkGLExt: YES (ver 1.2.0)
61. OpenGL support: YES (/usr/lib/x86_64-linux-gnu/libGLU.so /usr/lib/x86_64-
linux-gnu/libGL.so)
62. VTK support: NO
63.
64. Media I/O:
65. ZLib: build (ver 1.2.8)
66. JPEG: libjpeg (ver 90)
67. WEBP: build (ver 0.3.1)
68. PNG: /usr/lib/x86_64-linux-gnu/libpng.so (ver 1.2.54)
69. TIFF: /usr/lib/x86_64-linux-gnu/libtiff.so (ver 42 - 4.0.6)
70. JPEG 2000: /usr/lib/x86_64-linux-gnu/libjasper.so (ver 1.900.1)

71. OpenEXR: /usr/lib/x86_64-linux-gnu/liblmath.so /usr/lib/x86_64-linux-
gnu/liblImf.so /usr/lib/x86_64-linux-gnu/libIex.so /usr/lib/x86_64-linux-
gnu/libHalf.so /usr/lib/x86_64-linux-gnu/libIlmThread.so (ver 2.2.0)
72. GDAL: NO
73. GDCM: NO
74.
```

返回顶部



内容举报

返回顶部





1



1



```

75. Video I/O:
76.   DC1394 1.x: NO
77.   DC1394 2.x: YES (ver 2.2.4)
78.   FFmpeg: YES
79.     avcodec: YES (ver 56.60.100)
80.     avformat: YES (ver 56.40.101)
81.     avutil: YES (ver 54.31.100)
82.     swscale: YES (ver 3.1.101)
83.     avresample: NO
84.   GStreamer:
85.     base: YES (ver 1.8.3)
86.     video: YES (ver 1.8.3)
87.     app: YES (ver 1.8.3)
88.     riff: YES (ver 1.8.3)
89.     pbutils: YES (ver 1.8.3)
90.   OpenNI: NO
91.   OpenNI PrimeSensor Modules: NO
92.   OpenNI2: NO
93.   PvaAPI: NO
94.   GigE Vision SDK: NO
95.   Aravis SDK: NO
96.   UniCap: NO
97.   UniCap ucil: NO
98.   V4L/V4L2: YES/YES
99.   XIMEA: NO
100.  Xine: NO
101.  gPhoto2: YES
102.
103. Parallel framework: TBB (ver 4.4 interface 9002)
104.
105. Other third-party libraries:
106.   Use IPP: 9.0.1 [9.0.1]
107.     at: /mnt/sdb/ubuntu/install/opencv/build/3rdparty/ippicv/ippicv_in
108.   Use IPP Async: NO
109.   Use VA: NO
110.   Use Intel VA-API/OpenCL: NO
111.   Use Lapack: YES (/usr/lib/liblapack.so /usr/lib/libcblas.so /usr/lib/libat
112.   Use Eigen: NO
113.   Use Cuda: YES (ver 8.0)
114.   Use OpenCL: YES
115.   Use OpenVX: NO
116.   Use custom HAL: NO
117.
118. NVIDIA CUDA
119.   Use CUFFT: YES
120.   Use CUBLAS: YES
121.   USE NVCUVID: NO
122.
123.   NVIDIA GPU arch: 20 30 35 37 50 52 60 61
124.   NVIDIA PTX archs:
125.   Use fast math: NO
126.
127.   OpenCL: <Dynamic loading of OpenCL library>
128.   Include path: /mnt/sdb/ubuntu/install/opencv/opencv-
3.2.0/3rdparty/include/opencv/1.2
129.   Use AMDFFT: NO
130.   Use AMDBLAS: NO
131.
132. Python 2:
133.   Interpreter: /usr/bin/python2.7 (ver 2.7.12)
134.   Libraries: /usr/lib/x86_64-linux-gnu/libpython2.7.so (ver 2.7.12)
135.   numpy: /usr/local/lib/python2.7/dist-
packages/numpy/core/include (ver 1.13.1)
136.   packages path: lib/python2.7/dist-packages
137.
138. Python 3:
139.   Interpreter: /usr/bin/python3 (ver 3.5.2)
140.
141. Python (for build): /usr/bin/python2.7

```



内容举报

返回顶部



```
141.
142.   Java:
143.     ant:                      NO
144.     JNI:                      NO
145.     Java wrappers:           NO
146.     Java tests:              NO
147.
148.   Matlab:                     Matlab not found or implicitly disabled
149.
150.   Documentation:
151.     Doxygen:                  /usr/bin/doxygen (ver 1.8.11)
152.
153.   Tests and samples:
154.     Tests:                    YES
155.     Performance tests:       YES
156.     C/C++ Examples:          NO
157.
158.   Install path:               /usr/local
159.
160.   cvconfig.h is in:           /mnt/sdb/ubuntu/install/opencv/build
161.   -----
162.
163.   Configuring done
```

开始的部份会检查一堆依赖，请仔细看如可有缺包，可以百度一下怎么安装，因为我已经装完了就不提示了，  
不过有些包我也找不到我就没解决了如上面的

```
[plain]
1.  Checking for module 'libucil'
2.  No package 'libucil' found
```

这个我还真没找到怎么解决，不过反正问题不大  
再来下面的部份一定要注意几个部分一定要有

```
[plain]
1.  GUI:
2.  GTK+ 2.x:                      YES (ver 2.24.30)

[plain]
1.  Video I/O:
2.    V4L/V4L2:                    YES/YES
3.    FFMPEG:                      YES
4.  Python 2:
5.    Interpreter:                 /usr/bin/python2.7 (ver 2.7.12)
6.    Libraries:                   /usr/lib/x86_64-linux-gnu/libpython2.7.so (ver 2.7.12)
7.    numpy:                       /usr/local/lib/python2.7/dist-
packages/numpy/core/include (ver 1.13.1)
8.    packages path:               lib/python2.7/dist-packages
```

主要就是上面几个个非常重要，必须要装上  
再来就正常 make

```
[plain]
1.  $ make -j8
2.  $ make install
```

 内容举报

 返回顶部



比固态硬盘还快

设置nccl的ld环境

```
[plain]
1. $ vi /etc/ld.so.conf.d/nccl.conf
```

加上下面目录

```
[plain]
1. /usr/local/nccl/lib/
```

然后执行

```
[plain]
1. $ sudo ldconfig
```

安装完后回python按步骤一再次检查是否环境都好使了

二、撰写Python测试程序

```
[python]
1. # -*- coding:utf-8 -*-
2. # 用于模型的单张图像分类操作
3. import os
4. os.environ['GLOG_minloglevel'] = '2' # 将caffe的输出log信息不显示，必须放到import caffe前
5. import caffe # caffe 模块
6. from caffe.proto import caffe_pb2
7. from google.protobuf import text_format
8. import numpy as np
9. import cv2
10. import matplotlib.pyplot as plt
11. import time
12. import skimage.io
13.
14. global num
15. num = 0
16.
17.
18.
19. def detect(image1,net):
20.     # 传进来的image1的dtype为uint8
21.     # print image1.shape
22.     # print image1.dtype
23.     # print image1.size
24.
25.     # image = np.array(image1, dtype=np.float32)
26.     # image = caffe.io.resize_image(image1, (480, 640))
27.     image = skimage.img_as_float(image1).astype(np.float32)
28.     # image = caffe.io.resize_image(image2, (300, 300))
29.
30.     # skimage.io.imsave("photo.png", image)
31.     # cv2.imwrite("photo.png", image)
32.     # image = caffe.io.load_image(caffe_root + 'examples/images/bird.jpg')
33.     # 以下方式读取的imaged的dtype为float32
34.     # image = caffe.io.load_image(caffe_root + 'examples/images/bird.jpg')
```



内容举报

返回顶部

内容举报

返回顶部



1



```

34. # image = caffe.io.load_image(caffe_root + 'photo.png')
35. # image = caffe.io.load_image(image1)
36.
37. # 改变dtype
38. # image.dtype = 'float32'
39. # print 'mode:'+image.mode
40. # print image.shape
41. # print image.dtype
42. # print image.size
43.
44. # plt.imshow(image)
45.
46. # * Run the net and examine the top_k results
47. # In[5]:
48. global num
49. num += 1
50. print 'image num:' + str(num)
51.
52. transformed_image = transformer.preprocess('data', image)
53. net.blobs['data'].data[...] = transformed_image
54.
55. time_start=time.time()
56. # Forward pass.
57. net.forward()
58.
59. time_end=time.time()
60. print 'time:' + str(time_end-time_start) + ' s'
61.
62.
63.
64. loc = net.blobs['bbox-list'].data[0]
65. print(loc)
66. #查看了结构文件发现在CAFFE一开始图像输入的时候就已经将图片缩小了，宽度1248高度384
67. #然后我们在net.blobs['bbox-list'].data得到的是检测到的目标座标，但是是相对于1248*384的
68. #所以我们要把座标转换回相对原大小的位置，下面im.shape是保存在原尺寸的宽高，
69. for l in range(len(loc)):
70.     xmin = int(loc[l][0] * image.shape[1] / 1248)
71.     ymin = int(loc[l][1] * image.shape[0] / 384)
72.     xmax = int(loc[l][2] * image.shape[1] / 1248)
73.     ymax = int(loc[l][3] * image.shape[0] / 384)
74.     #在该座标位置画一个方框
75.     cv2.rectangle(image, (xmin, ymin), (xmax, ymax), (55 / 255.0, 255 / 255.0, 155 / 255.0))
76. # 显示结果
77.
78. #plt.imshow(image, 'brg')
79. #plt.show()
80. cv2.imshow('img', image)
81.
82.
83.
84. def show_info(cam):
85.     print 'POS_FRAMES:'+str(cam.get(1))
86.     print 'FRAME_COUNT:'+str(cam.get(7))
87.     print 'FORMAT:'+str(cam.get(8))
88.     print 'MODE:'+str(cam.get(9))
89.     print 'SATURATION:'+str(cam.get(12))
90.     print 'FPS:'+str(cam.get(5))
91.
92. #CPU或GPU模型转换
93. caffe.set_mode_gpu()
94. #caffe.set_mode_cpu()
95. #caffe.set_device(0)
96.
97. caffe_root = '/var/smb/work/mycode/'
98. # 网络参数（权重）文件
99. caffemodel = caffe_root + 'module/detectnet/snapshot_iter_2391.caffemodel'
100. # 网络实施结构配置文件
101. deploy = caffe_root + 'module/detectnet/deploy.prototxt'

```



### 内容举报

[返回顶部](#)

1



```
102.
103.
104. img_root = caffe_root + 'data/'
105.
106. # 网络实施分类
107. net = caffe.Net(deploy, # 定义模型结构
108.                 caffemodel, # 包含了模型的训练权值
109.                 caffe.TEST) # 使用测试模式(不执行dropout)
110.
111. # 加载ImageNet图像均值 (随着Caffe一起发布的)
112. print(os.environ['PYTHONPATH'])
113. #mu = np.load(os.environ['PYTHONPATH'] + '/caffe/imagenet/ilsvrc_2012_mean.npy')
114. #mu = mu.mean(1).mean(1) # 对所有像素值取平均以此获取BGR的均值像素值
115.
116. # 图像预处理
117. transformer = caffe.io.Transformer({'data': net.blobs['data'].data.shape})
118. transformer.set_transpose('data', (2,0,1))
119. #transformer.set_mean('data', mu)
120. transformer.set_raw_scale('data', 255)
121. transformer.set_channel_swap('data', (2,1,0))
122.
123. # 处理图像
124. cam = cv2.VideoCapture(0)
125. if cam.isOpened():
126.     cam.set(3, 400)
127.     cam.set(4, 300)
128.     cam.set(5, 3)
129.     time.sleep(6)
130.     cam.set(15, -8.0)
131.     size = (int(cam.get(3)), int(cam.get(4)))
132.     print 'size:'
133.     print size
134.
135. cv2.namedWindow('img', cv2.WINDOW_NORMAL)
136.
137. # cnt=2
138. # while cnt:
139. #     cnt -= 1
140. while cam.isOpened():
141.     ret, img = cam.read()
142.     if ret:
143.         #show_info(cam)
144.         detect(img, net)
145.
146.     if 0xFF == ord('q') & cv2.waitKey(5) == 27:
147.         break
148.     # time.sleep(0.033)
149. cam.release()
150. cv2.destroyAllWindows()
```

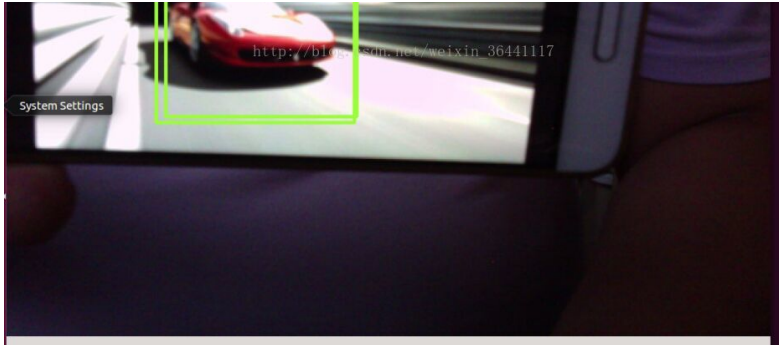
介面上会打印bbox也就是侦测到的目标在图像的座标，另外请自行修改python代码里的相关目录，



内容举报

返回顶部





我用自己训练的KITTI数据集，用于侦测车辆，因为拍不到车子拿手机欺骗一下，好使

版权声明：本文为博主原创文章，未经博主允许不得转载。



发表你的评论

([http://my.csdn.net/weixin\\_35068028](http://my.csdn.net/weixin_35068028))



hu3350261 (/hu3350261) 2017-08-21 21:12

1楼

(/hu3350261)

回复

### 相关文章推荐

**程 | 深度学习 + OpenCV，Python 实现实时视频目标检测 机器之心 09-21** (<http://blog.csdn.net/...>)

教程 | 深度学习 + OpenCV，Python 实现实时视频目标检测 机器之心 09-21 选自 PyImageSearch 机器之心编译 参与：路雪、李泽南 使用 ...



weixin\_40355324 ([http://blog.csdn.net/weixin\\_40355324](http://blog.csdn.net/weixin_40355324)) 2017年12月04日 21:29 9

**Ubuntu 14.04 内核升级 详细步骤** (<http://blog.csdn.net/u011774239/article/details/49361055>)

1. 下载源码 [www.kernel.org/pub/linux/kernel/v3.x/](http://www.kernel.org/pub/linux/kernel/v3.x/)，比如下载到linux-3.14.tar.gz，大约100+M 2. 拷贝到/usr/src目录下，用 ...



u011774239 (<http://blog.csdn.net/u011774239>) 2015年10月23日 14:31 5549




广告

### AI 专业人才缺口上百万，年薪 80 万远超同行？

就目前来看，国内 AI 人才缺乏且经验不足，为争抢优秀人才，企业背后的暗战早已打响。作为正在谋求一份好工作我，又该如何抉择...

([http://www.baidu.com/cb.php?c=lgF\\_pyfqHmknjnvPjn0IZ0qnfK9ujYzP1ndPWb10Aw-](http://www.baidu.com/cb.php?c=lgF_pyfqHmknjnvPjn0IZ0qnfK9ujYzP1ndPWb10Aw-))

 内容举报

 返回顶部



比固态硬盘还快

5Hc3rHnYnHb0TAq15HfLPWRznpj0T1YLP1mYnWmdPHPBrjm3n1T30AwY5HDdnHc3rj0LPWR0IgF\_5y9Y1ZOIQzq-uZR8mLPbUB48ugfEIAqspynEmybz5LNYUNq1ULNzmvRqmhkEu1Ds0ZFb5HD0mhYqn0KsTWYs0ZNGujYKPHYn1mk0AqGujYKnWb3rjDY0APGujYLnWm4n1c0ULI85H00TZbqnV0S0APzm1YdnWTL)

【Python+OpenCV】目标跟踪-实现基本的运动检测 (http://blog.csdn.net/lwplwf/article/deta...

目标跟踪是对摄像头视频中的移动目标进行定位的过程，有着非常广泛的应用。实时目标跟踪是许多计算机视觉应用的重要任务，如监控、基于感知的用户界面、增强现实、基于对象的视频压缩以及辅助驾驶等。...

lwplwf (http://blog.csdn.net/lwplwf) 2017年06月21日 09:26 3792

opencv3-python目标跟踪—基本的运动检测 (http://blog.csdn.net/qq\_16463925/article/deta...

目标跟踪是对摄像头视频中的移动目标进行定位的过程，一种简单的方法就是计算帧与帧之间的差异。本文利用“背景”zhen与其他帧之间的差异，来跟踪视频中的目标。#coding=utf-8 import cv...

qq\_16463925 (http://blog.csdn.net/qq\_16463925) 2017年06月26日 10:56 1017

计算机视觉 | Python OpenCV 3 使用背景减除进行目标检测 (http://blog.csdn.net/leadai/artic...

北京 | 高性能计算之GPU CUDA课程11月24-26日3天密集学习 快速带你晋级阅读全文> 背景减除(Background Subtraction)是许多基于计算机视觉的任务中的主要预...

leadai (http://blog.csdn.net/leadai) 2017年11月15日 00:00 110



人人都能看懂的 AI 入门课

本课程将讲述人工智能的现状、应用场景和入门方法，并通过运用 TensorFlow，使得受众能清晰了解人工智能的运作方式。

(http://www.baidu.com/cb.php?c=lgF\_pyfqHmknjtzrc0IZ0qnfK9ujYzP1f4Pjn10Aw-5Hc4nj6vPjm0TAq15Hf4rjn1n1b0T1dBuHDzuW6kuAf3rARvmHcY0AwY5HDdnHc3rj0LPWR0IgF\_5y9Y1ZOIQzqMpgwBUvqoQhP8QvIGIAPCmgfEmvq\_ljd8Q1R4uWI-n16kPWKW6rHnvnHRvnnNBuYD4PHqdAdxTvqdThP-5HDknWFWmHkEusKzujYk0AFV5H00TZcqn0KdpyfqHRLPjnvnfKEpyfqHnsnj0YnsKWpyfqP1cvrHnz0AqLUWYs0ZK45HcsP6KWThnqPWcvPHb)

python opencv检测目标颜色 (http://blog.csdn.net/Lingdongtianxia/article/details/75194950)

# -\*- coding:utf-8 -\*- \_\_author\_\_ = 'kingking' \_\_version\_\_ = '1.0' \_\_date\_\_ = '14/07/2017' import cv...

Lingdongtianxia (http://blog.csdn.net/Lingdongtianxia) 2017年07月15日 23:56 986

《OpenCV 3计算机视觉：Python语言实现》学习笔记——目标跟踪中基本运动检测的思考 (ht...

刚刚接触Python以及OpenCV，通过敲本书中的代码来学习计算机视觉。在按照书中代码进行实现时，发现有些部分总是没有作者实现的好(代码是完全一些样的)。经过反复琢磨对比试验后，明白了其中的原因：作...

dongshengweidsw (http://blog.csdn.net/dongshengweidsw) 2017年05月13日 14:54 598

【用Python学习Caffe】2. 使用Caffe完成图像目标检测 (http://blog.csdn.net/tostq/article/de...

2. 使用Caffe完成图像目标检测本节将以一个快速的图像目标检测网络SSD作为例子，通过Python Caffe来进行图像目标检测。必须安装windows-ssd版本的Caffe，或者自行在c...



内容举报





返回顶部





内容举报



返回顶部



 [tostq \(http://blog.csdn.net/tostq\)](http://blog.csdn.net/tostq) 2017年06月22日 22:08  1856



**目标检测算法SSD运行代码caffe** (<http://download.csdn.net/download/maji...>)  
<http://download.csdn.net/download/maji...> 2017年09月27日 09:13 12.45MB [下载](#)



**QT使用Opencv库及网络摄像头做的视频监控系统带人脸检测功能** (<http://do...>)  
<http://do...> 2017年09月26日 19:30 2.34MB [下载](#)

**使用caffe训练的深度学习做目标检测(车辆检测)** (<http://blog.csdn.net/u013289254/article/det...>)  

```
#include "opencv2/core/core.hpp" #include "opencv2/imgproc/imgproc.hpp" #include "opencv2/highgui/hi...
```

 [u013289254 \(http://blog.csdn.net/u013289254\)](http://blog.csdn.net/u013289254) 2017年04月22日 15:09  2241



**利用python和opencv做的摄像头颜色提取** ([http://blog.csdn.net/qq\\_24693837/article/details...](http://blog.csdn.net/qq_24693837/article/details...))  
最近导师给了个项目，做个智能捡拾乒乓球机器人，其中一部分涉及摄像头动态捕捉黄色乒乓球。这对于初涉机器视觉开发的大三学生来说是个难题，于是我把任务分解，先完成黄色的颜色提取，再往后做后续的任务。 于是...  
 [qq\\_24693837 \(http://blog.csdn.net/qq\\_24693837\)](http://blog.csdn.net/qq_24693837) 2016年09月18日 21:46  2324

**基于win10，GPU的Tensorflow Object Detection API部署及USB摄像头目标检测** (<http://blog...>)  
TensorFlow安装 下载 TensorFlow models 配置依赖库 程序配置和运行1.TensorFlow安装安装教程在CSDN上有很多文章，但最好依据官方教程，因为TensorFlow...  
 [asukasmallriver \(http://blog.csdn.net/asukasmallriver\)](http://blog.csdn.net/asukasmallriver) 2017年12月02日 15:46  90



**目标检测matlab代码 matlab控制摄像头自动定时拍摄-gui源码** (<http://downl...>)  
<http://downl...> 2009年05月06日 09:32 12KB [下载](#)

**TensorFlow usb摄像头视频目标检测代码** (<http://download.csdn.net/down...>)  
 2017年12月02日 15:50 6KB [下载](#)  
<http://download.csdn.net/down...>

 [benqyanluo1542 \(http://blog.csdn.net/bengyanluo1542\)](http://blog.csdn.net/bengyanluo1542) 2017年07月15日 21:13  308

**使用Windows Caffe 建立Qt工程 运行SSD(Single Shot MultiBox Detector)目标检测算法** (<http://download.csdn.net/download/bengyanluo1542/5444444>)  
使用Windows Caffe 建立Qt工程 运行SSD(Single Shot MultiBox Detector)目标检测算法

**DL开源框架Caffe | 目标检测Faster-rcnn问题全解析** (<http://blog.csdn.net/u010402786/articl...>)  
目标检测框架Faster-RCNN训练以及应用时常见问题



 内容举报

 返回顶部



u010402786 (<http://blog.csdn.net/u010402786>) 2017年05月24日 14:06 1877

## DL开源框架Caffe | 目标检测Faster-rcnn训练自己数据问题整理 (<http://blog.csdn.net/u010402786>)

目标检测Faster-rcnn训练自己数据问题整理



u010402786 (<http://blog.csdn.net/u010402786>) 2017年05月23日 17:03 2038

## Linux 编译SSD Caffe目标检测代码 (<http://blog.csdn.net/hqh45/article/details/78734983>)

引言SSD: Single Shot MultiBox Detector 是ECCV 2016上发表的目标检测的论文 但是编译的时候确出现了 Type name declared error等问题,我...



hqh45 (<http://blog.csdn.net/hqh45>) 2017年12月06日 23:55 8

## SSD+caffe | Single Shot MultiBox Detector 目标检测 (一) ([http://blog.csdn.net/sinat\\_26917383](http://blog.csdn.net/sinat_26917383))

作者的思路就是Faster R-CNN+YOLO, 利用YOLO的思路和Faster R-CNN的anchor box的思想。 .0 导读 ( 本节来源于BOT 大赛嘉宾问答环节 ) SSD 这里的设计就...



sinat\_26917383 ([http://blog.csdn.net/sinat\\_26917383](http://blog.csdn.net/sinat_26917383)) 2017年03月28日 17:13 6405



内容举报



返回顶部