## 10.YOLOv3 environment building and camera real-time detection

Official Link:https://pjreddie.com/darknet/yolo/

In this course, we use the tiny version of YOLOv3. The official version and the tiny version are installed in the same way, except that the configuration file and the weight file are different.

**1. Install CUDA, OpenCV, cuDNN**
Please refer to tutorial 1 for details.

**2. Download**
git clone https://github.com/AlexeyAB/darknet.git

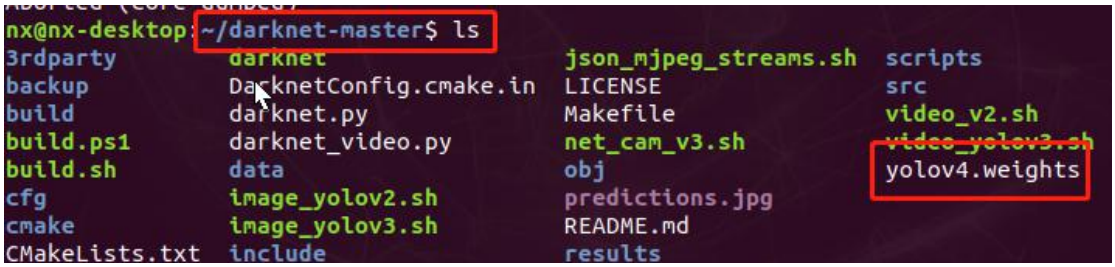
Note:
If this prompt appears “Error: RPC failed; curl 56 GnuTLS recv error (-54): Error in the pull function.”
This is due to insufficient git default cache size. Use the following command to increase the cache size
git config --global http.postBuffer 5242880000
If it still doesn’t work, it may be that the network speed is slow, configure the minimum speed and minimum speed time of git
git config --global http.lowSpeedLimit 0
git config --global http.lowSpeedTime 999999
If it still doesn’t work, it’s a network problem from my own experience.
You can also download the ZIP file to your computer by going to https://github.com/AlexeyAB/darknet.git, and then unzip it.

**3. Configuration**
**cd darknet**
**sudo vim Makefile** #Modify Makefile

**4. Modify the first three lines of the Makefile**
GPU=1
CUDNN=1
OPENCV=1

**5. Compile**
make -j4

**6. Download the weight file**
We can directly copy yolov4.weights to the darknet directory. As shown below.




**7. Testing**

**Picture detection**

./darknet detect cfg/yolov4.cfg yolov4.weights data/dog.jpg  # Simplified version

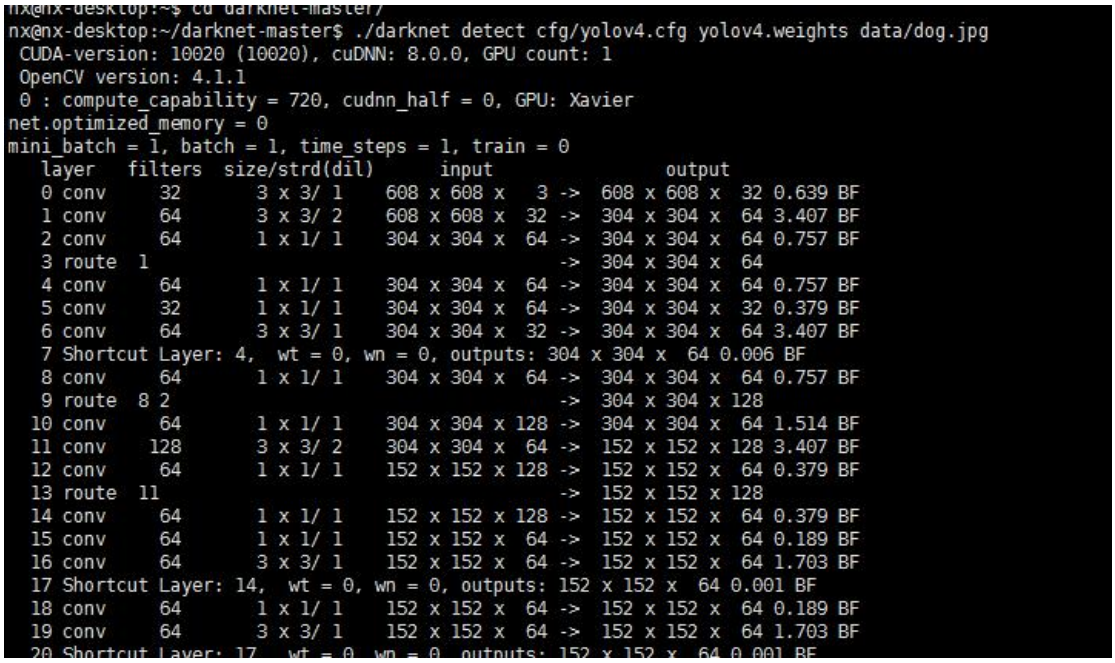
./darknet detector test cfg/coco.data cfg/yolov4.cfg yolov4.weights data/dog.jpg

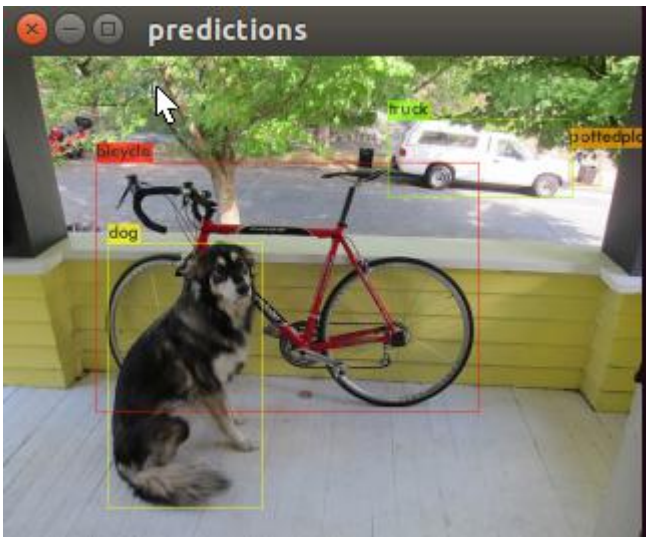
#Full version

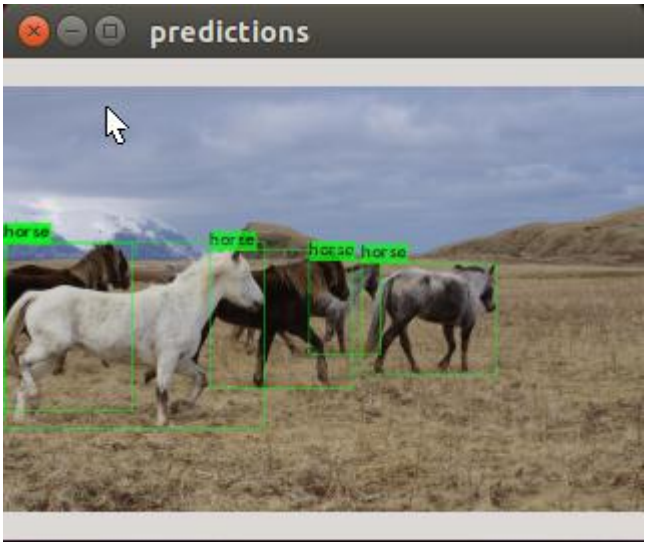
# Change the detection threshold

# By default, YOLO only displays objects detected with a confidence of .25 or higher. You can change this setting by passing the -thresh <val> flag to the yolo command. For example, to display all detections, you can set the threshold to 0.1:

./darknet detect cfg/yolov4.cfg yolov4.weights data/dog.jpg -thresh 0.1



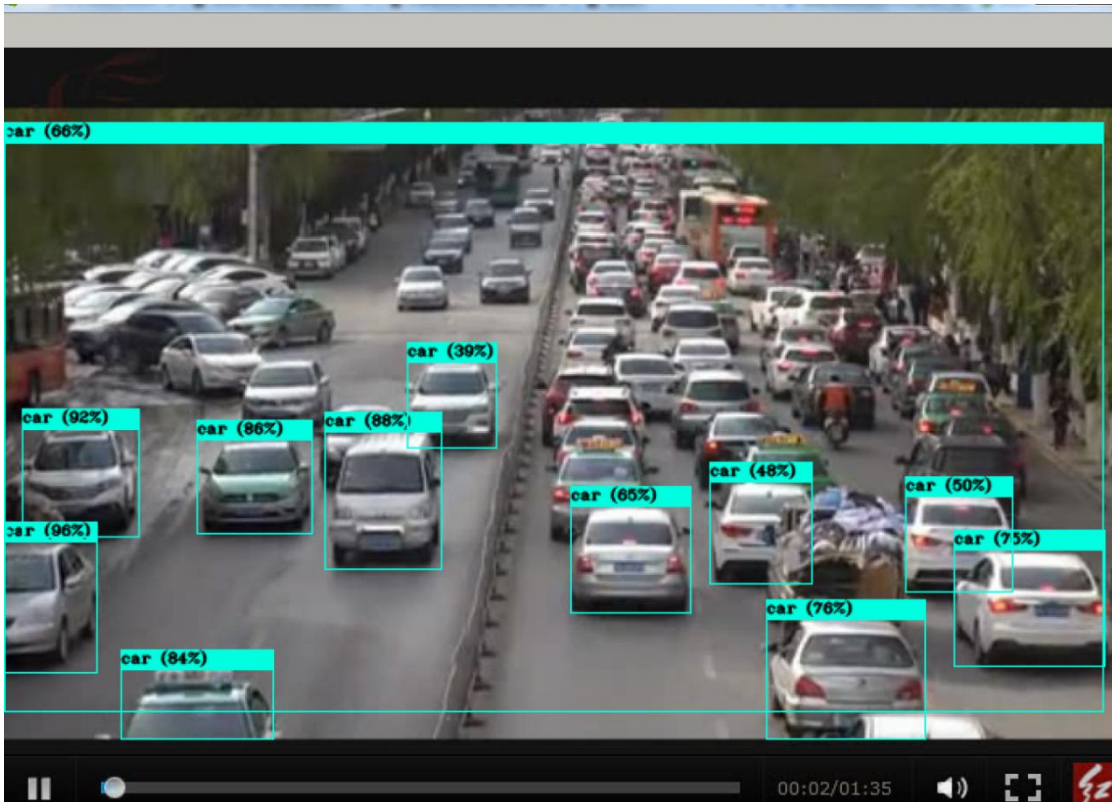



**Video detection**

./darknet detector demo cfg/coco.data cfg/yolov4.cfg yolov4.weights data/123.mp4

```
nx@nx-desktop:~/darknet-master$ ./darknet detector demo cfg/coco.data cfg/yolov4.cfg yolov4.weights data/123.mp4
 CUDA-version: 10020 (10020), cuDNN: 8.0.0, GPU count: 1
 OpenCV version: 4.1.1
Demo
 0 : compute_capability = 720, cudnn_half = 0, GPU: Xavier
net.optimized_memory = 0
mini_batch = 1, batch = 1, time_steps = 1, train = 0
   layer   filters  size/strd(dil)      input                output
   0 conv     32       3 x 3/ 1    608 x 608 x   3 ->  608 x 608 x  32 0.639 BF
   1 conv     64       3 x 3/ 2    608 x 608 x  32 ->  304 x 304 x  64 3.407 BF
   2 conv     64       1 x 1/ 1    304 x 304 x  64 ->  304 x 304 x  64 0.757 BF
   3 route  1                                  ->  304 x 304 x  64
   4 conv     64       1 x 1/ 1    304 x 304 x  64 ->  304 x 304 x  64 0.757 BF
   5 conv     32       1 x 1/ 1    304 x 304 x  64 ->  304 x 304 x  32 0.379 BF
   6 conv     64       3 x 3/ 1    304 x 304 x  32 ->  304 x 304 x  64 3.407 BF
```

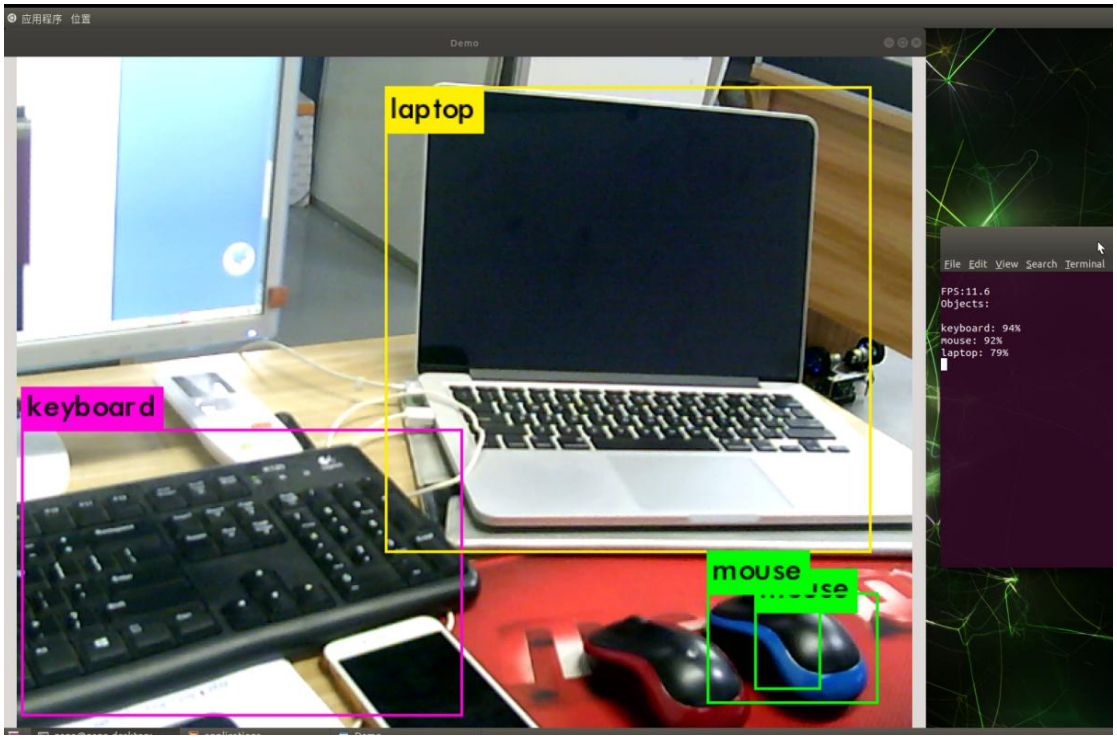


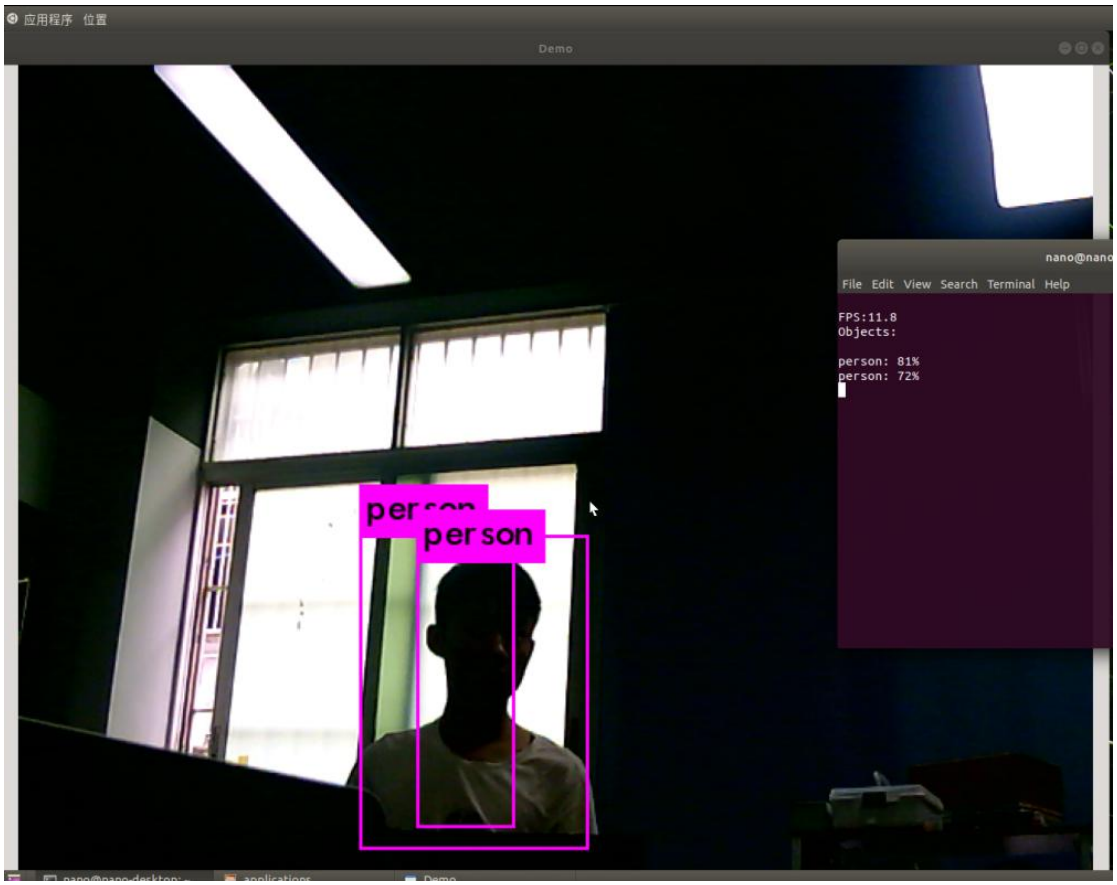

**Camera real-time detect method**

./darknet detector demo cfg/coco.data cfg/yolov4.cfg yolov4.weights /dev/video1

!Note:

Video needs to select the number corresponding to the current USB camera, such as video0, video1


应用程序 位置
Demo
laptop
keyboard
mouse
File Edit View Search Terminal
FPS:11.6
Objects:
keyboard: 94%
mouse: 92%
laptop: 79%


应用程序 位置
Demo
File Edit View Search Terminal Help
FPS:11.8
Objects:
person: 81%
person: 72%
person


192.168.1.182:5900 (nano@nano-desktop) - VNC Viewer
Demo
keyboard (26%)
keyboard (35%)
mouse (29%)
mouse (28%)
mouse (29%)
cell phone (39%)
nano@nano-desktop: ~/dartnetnew
File Edit View Search Terminal Help
laptop: 59%
FPS:12.0
Objects:
keyboard: 51%
keyboard: 31%
mouse: 44%
mouse: 38%
mouse: 30%
laptop: 63%
FPS:15.4
Objects:
cell phone: 36%
keyboard: 45%
mouse: 69%
mouse: 48%
mouse: 36%
laptop: 58%
FPS:13.8
applications
Demo