

## 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/pjreddie/darknet.git
```

### 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, download the tiny version of the weight file directly here.

```
wget https://pjreddie.com/media/files/yolov3-tiny.weights
```

### 7. Test

```
./darknet detect cfg/yolov3-tiny.cfg yolov3-tiny.weights data/dog.jpg
```

| layer        | filters | size      | input          | output            |
|--------------|---------|-----------|----------------|-------------------|
| 0 conv       | 16      | 3 x 3 / 1 | 416 x 416 x 3  | -> 416 x 416 x 16 |
| 0.150 BFLOPs |         |           |                |                   |
| 1 max        |         | 2 x 2 / 2 | 416 x 416 x 16 | -> 208 x 208 x 16 |
| 2 conv       | 32      | 3 x 3 / 1 | 208 x 208 x 16 | -> 208 x 208 x 32 |
| 0.399 BFLOPs |         |           |                |                   |
| 3 max        |         | 2 x 2 / 2 | 208 x 208 x 32 | -> 104 x 104 x 32 |
| 4 conv       | 64      | 3 x 3 / 1 | 104 x 104 x 32 | -> 104 x 104 x 64 |
| 0.399 BFLOPs |         |           |                |                   |
| 5 max        |         | 2 x 2 / 2 | 104 x 104 x 64 | -> 52 x 52 x 64   |
| 6 conv       | 128     | 3 x 3 / 1 | 52 x 52 x 64   | -> 52 x 52 x 128  |
| 0.399 BFLOPs |         |           |                |                   |

|              |      |           |      |           |    |      |           |
|--------------|------|-----------|------|-----------|----|------|-----------|
| 7 max        |      | 2 x 2 / 2 | 52 x | 52 x 128  | -> | 26 x | 26 x 128  |
| 8 conv       | 256  | 3 x 3 / 1 | 26 x | 26 x 128  | -> | 26 x | 26 x 256  |
| 0.399 BFLOPs |      |           |      |           |    |      |           |
| 9 max        |      | 2 x 2 / 2 | 26 x | 26 x 256  | -> | 13 x | 13 x 256  |
| 10 conv      | 512  | 3 x 3 / 1 | 13 x | 13 x 256  | -> | 13 x | 13 x 512  |
| 0.399 BFLOPs |      |           |      |           |    |      |           |
| 11 max       |      | 2 x 2 / 1 | 13 x | 13 x 512  | -> | 13 x | 13 x 512  |
| 12 conv      | 1024 | 3 x 3 / 1 | 13 x | 13 x 512  | -> | 13 x | 13 x 1024 |
| 1.595 BFLOPs |      |           |      |           |    |      |           |
| 13 conv      | 256  | 1 x 1 / 1 | 13 x | 13 x 1024 | -> | 13 x | 13 x 256  |
| 0.089 BFLOPs |      |           |      |           |    |      |           |
| 14 conv      | 512  | 3 x 3 / 1 | 13 x | 13 x 256  | -> | 13 x | 13 x 512  |
| 0.399 BFLOPs |      |           |      |           |    |      |           |
| 15 conv      | 255  | 1 x 1 / 1 | 13 x | 13 x 512  | -> | 13 x | 13 x 255  |
| 0.044 BFLOPs |      |           |      |           |    |      |           |
| 16 yolo      |      |           |      |           |    |      |           |
| 17 route     | 13   |           |      |           |    |      |           |
| 18 conv      | 128  | 1 x 1 / 1 | 13 x | 13 x 256  | -> | 13 x | 13 x 128  |
| 0.011 BFLOPs |      |           |      |           |    |      |           |
| 19 upsample  |      | 2x        | 13 x | 13 x 128  | -> | 26 x | 26 x 128  |
| 20 route     | 19 8 |           |      |           |    |      |           |
| 21 conv      | 256  | 3 x 3 / 1 | 26 x | 26 x 384  | -> | 26 x | 26 x 256  |
| 1.196 BFLOPs |      |           |      |           |    |      |           |
| 22 conv      | 255  | 1 x 1 / 1 | 26 x | 26 x 256  | -> | 26 x | 26 x 255  |
| 0.088 BFLOPs |      |           |      |           |    |      |           |
| 23 yolo      |      |           |      |           |    |      |           |

Loading weights from yolov3-tiny.weights...Done!

data/dog.jpg: Predicted in 0.239507 seconds.

dog: 56%

car: 52%

truck: 56%

car: 62%

bicycle: 58%

```
./darknet detect cfg/yolov3.cfg yolov3.weights data/dog.jpg # Short version
```

```
./darknet detector test cfg/coco.data cfg/yolov3.cfg yolov3.weights
```

```
data/dog.jpg # full version
```

```
# Change detection threshold
```

# By default, YOLO only displays objects with a detected confidence level of .25 or higher. You can change this setting by passing the -thresh <val> flag to the yolo command.

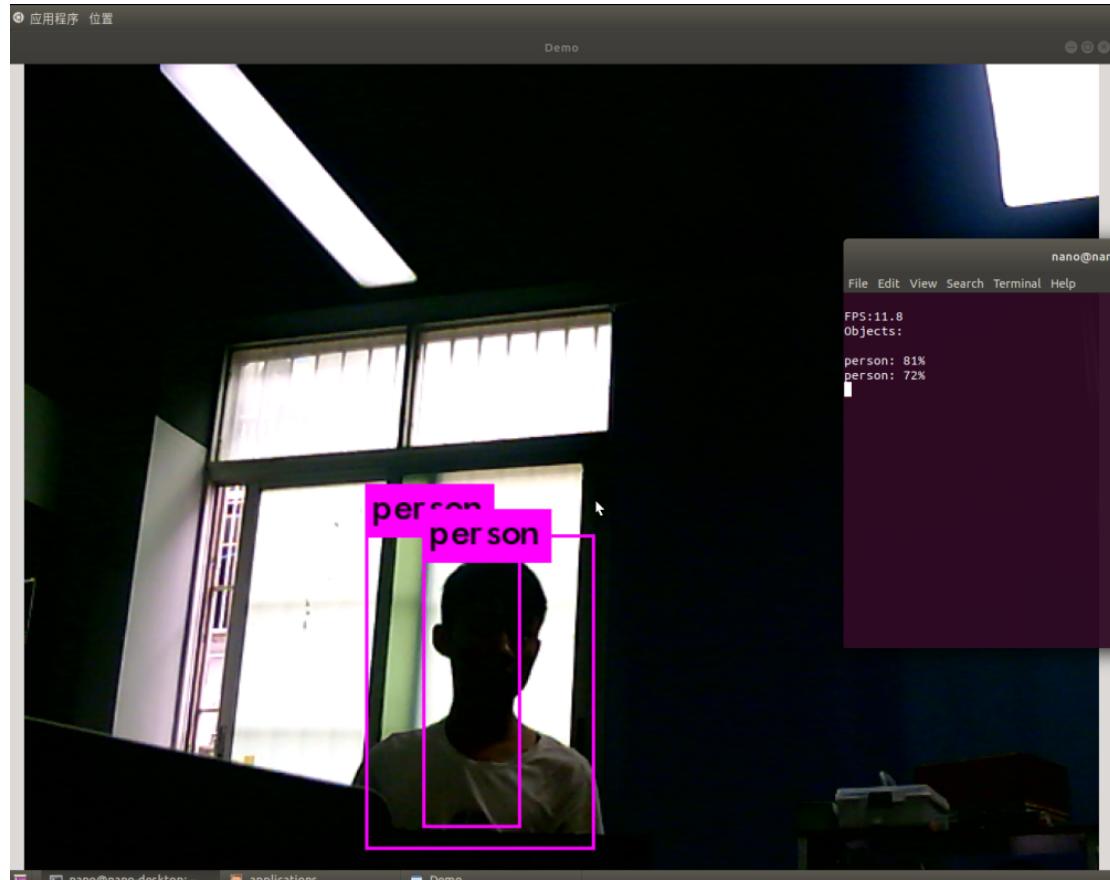
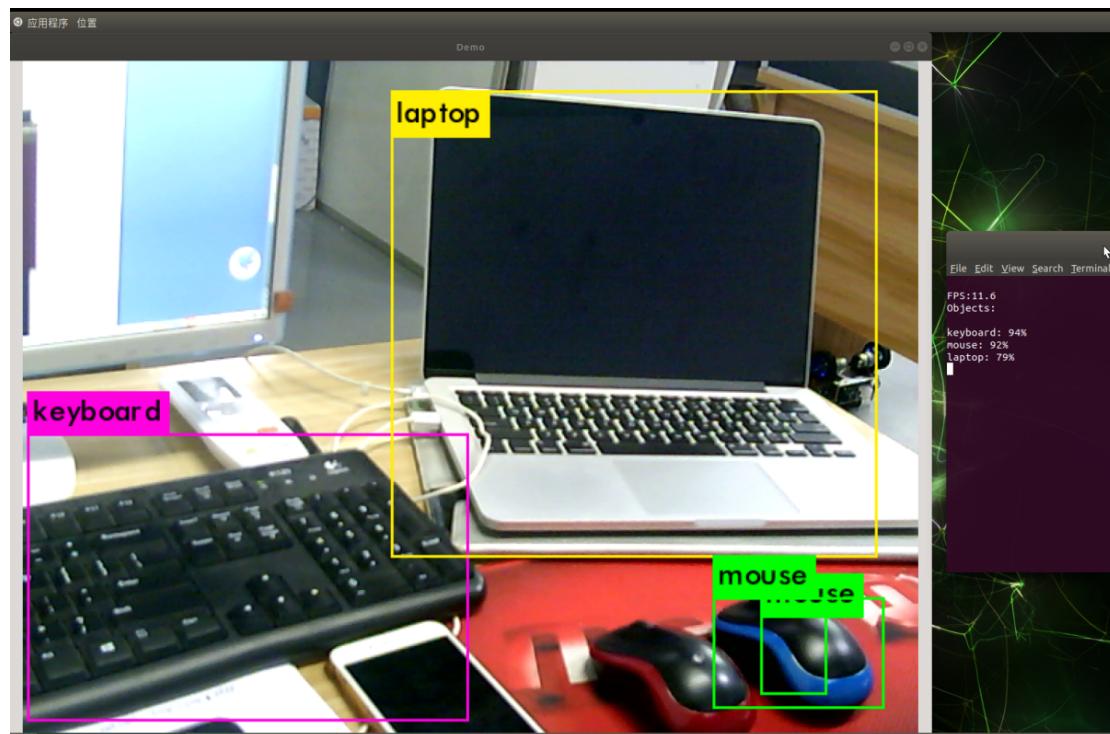
For example, you can set the threshold to 0 to display all tests, :

```
./darknet detect cfg/yolov3.cfg yolov3.weights data/dog.jpg -thresh 0
```

### Camera real-time detect method:

./darknet detector demo cfg/coco.data cfg/yolov3-tiny.cfg yolov3-tiny.weights

**!!! Note: The power supply must be 5V 4A.**



## 8.Coping plan when resources are insufficient

Memory of Nano is too small, sometimes you need to expand the memory by swap. After the nano is restarted, the swap space is automatically reclaimed.

```
# Disable previous
sudo swapoff /swapfile
```

```
# Modify the size of the swap space to 4G
sudo dd if=/dev/zero of=/swapfile bs=1M count=4096
```

```
# Set the file to the "swap file" type
sudo mkswap /swapfile
```

```
# Start swapfile
sudo swapon /swapfile
```

### Exapnd course:

If you feel that the experiment is not working well, you can use the following github address version.

Re-create the directory and execute the following command:

```
git clone https://github.com/AlexeyAB/darknet
```

Results as shown below:

