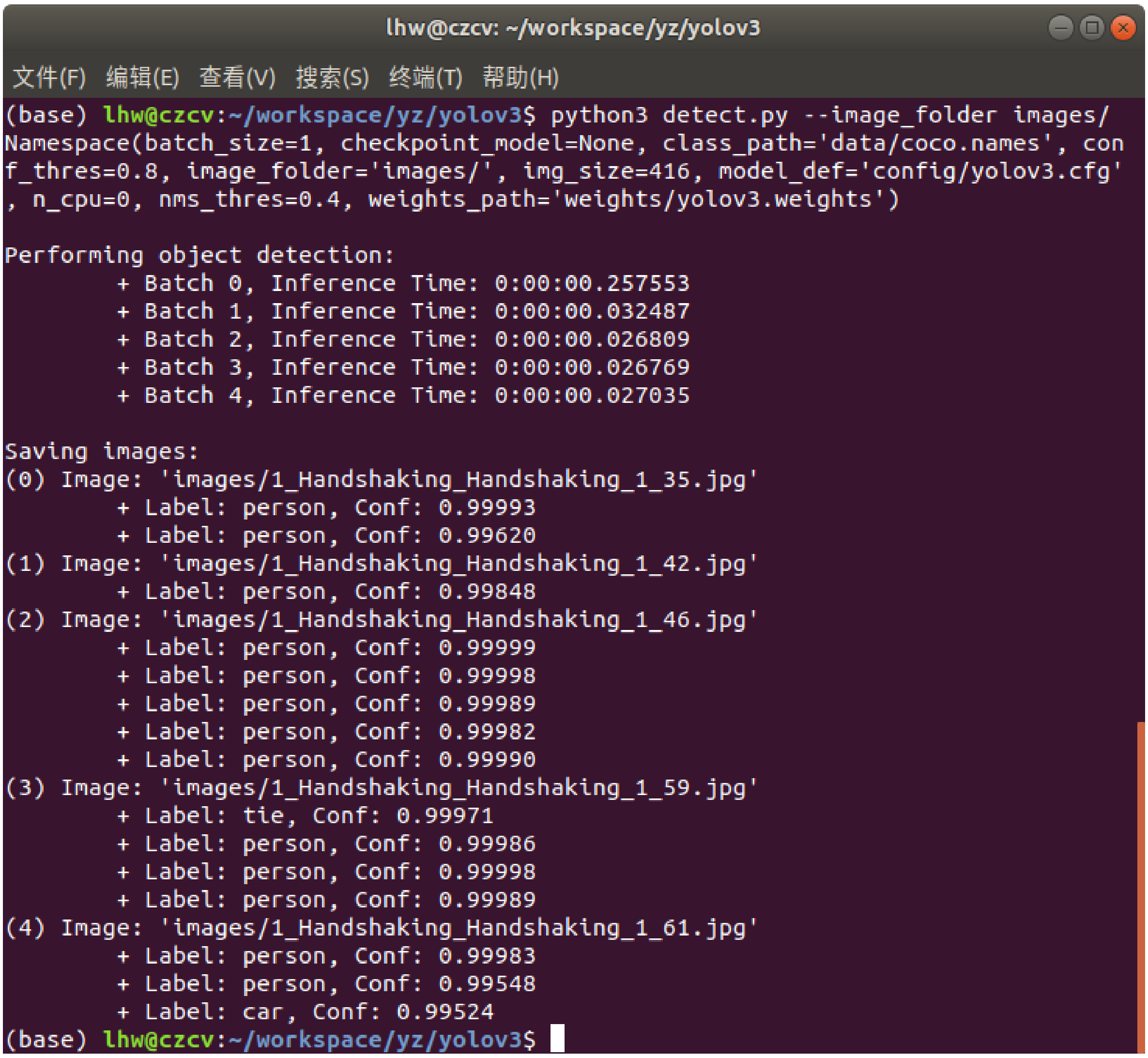
预训练模型推理和测试

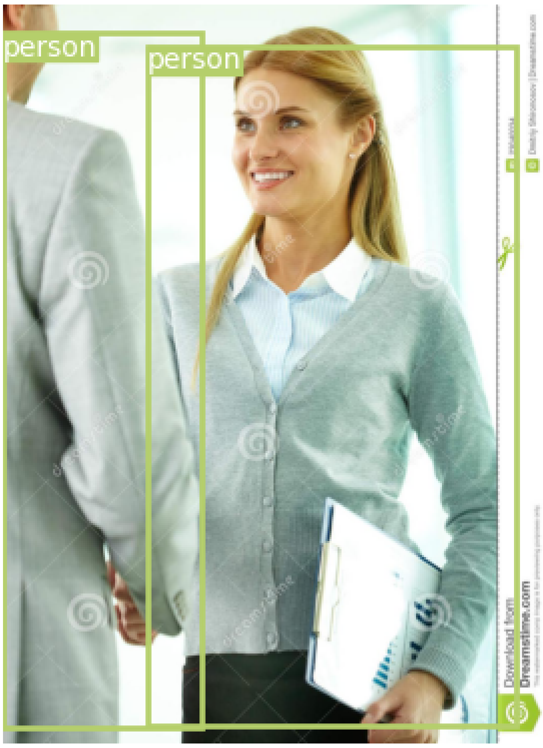
推理

利用pretrained model进行推理，结果生成在output文件夹

```
python3 detect.py --image_folder images/
```

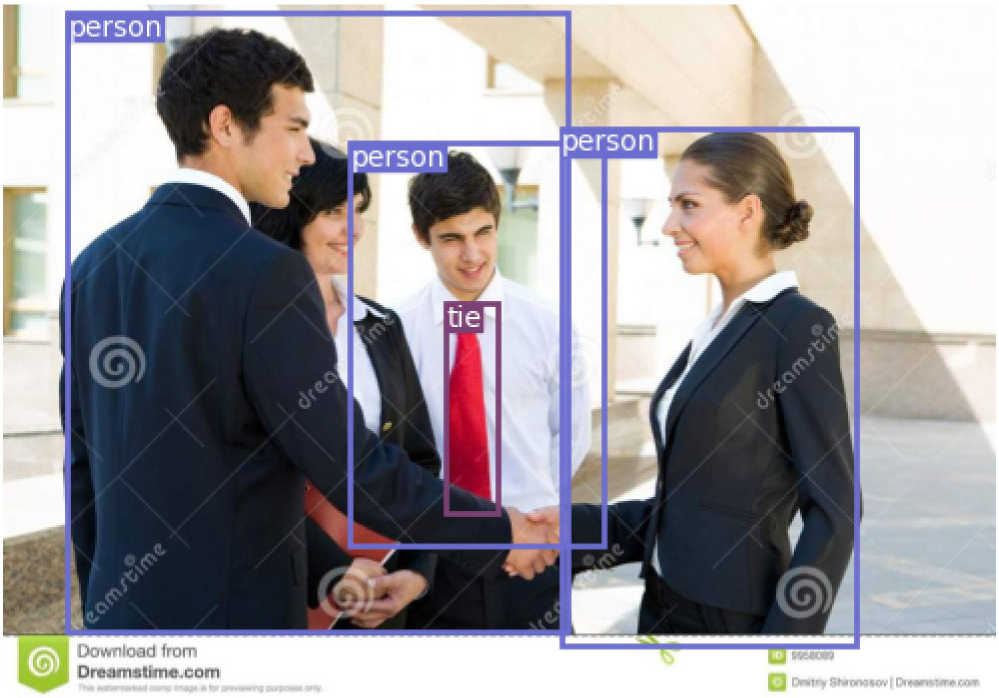


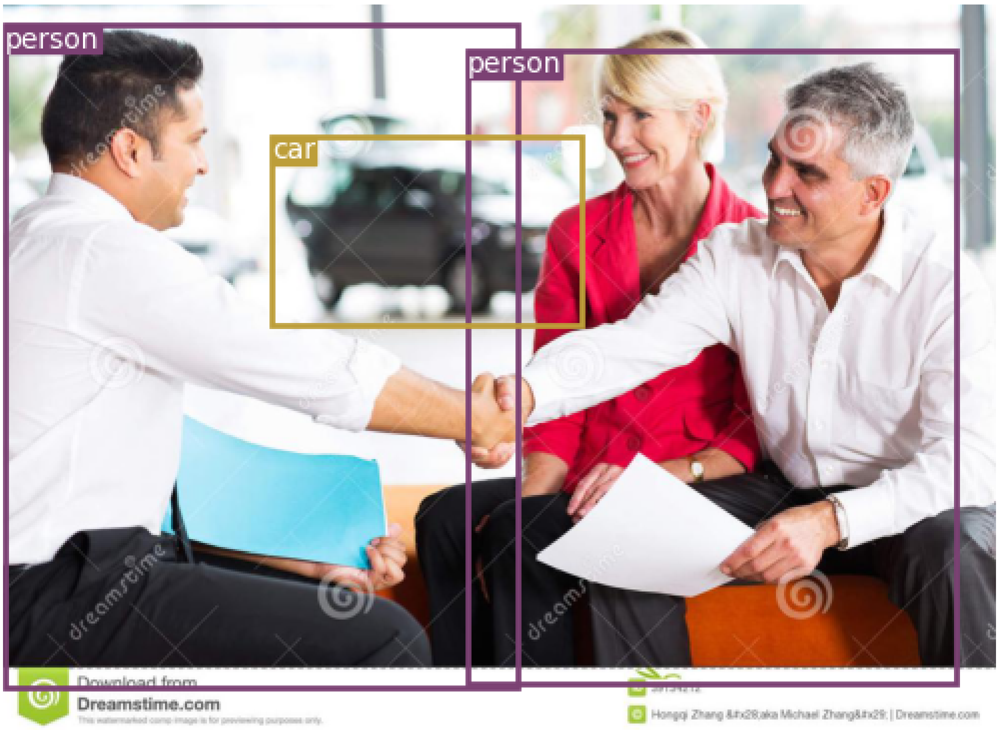
```
(base) lhw@czcv:~/workspace/yz/yolov3$ python3 detect.py --image_folder images/
Namespace(batch_size=1, checkpoint_model=None, class_path='data/coco.names', con
f_thres=0.8, image_folder='images/', img_size=416, model_def='config/yolov3.cfg'
, n_cpu=0, nms_thres=0.4, weights_path='weights/yolov3.weights')

Performing object detection:
        + Batch 0, Inference Time: 0:00:00.257553
        + Batch 1, Inference Time: 0:00:00.032487
        + Batch 2, Inference Time: 0:00:00.026809
        + Batch 3, Inference Time: 0:00:00.026769
        + Batch 4, Inference Time: 0:00:00.027035

Saving images:
(0) Image: 'images/1_Handshaking_Handshaking_1_35.jpg'
        + Label: person, Conf: 0.99993
        + Label: person, Conf: 0.99620
(1) Image: 'images/1_Handshaking_Handshaking_1_42.jpg'
        + Label: person, Conf: 0.99848
(2) Image: 'images/1_Handshaking_Handshaking_1_46.jpg'
        + Label: person, Conf: 0.99999
        + Label: person, Conf: 0.99998
        + Label: person, Conf: 0.99989
        + Label: person, Conf: 0.99982
        + Label: person, Conf: 0.99990
(3) Image: 'images/1_Handshaking_Handshaking_1_59.jpg'
        + Label: tie, Conf: 0.99971
        + Label: person, Conf: 0.99986
        + Label: person, Conf: 0.99998
        + Label: person, Conf: 0.99989
(4) Image: 'images/1_Handshaking_Handshaking_1_61.jpg'
        + Label: person, Conf: 0.99983
        + Label: person, Conf: 0.99548
        + Label: car, Conf: 0.99524
(base) lhw@czcv:~/workspace/yz/yolov3$
```



person
Download from
Dreamstime.com
This watermarked comp image is for previewing purposes only.
ID 14834345
Dmitriy Shironosov | Dreamstime.com

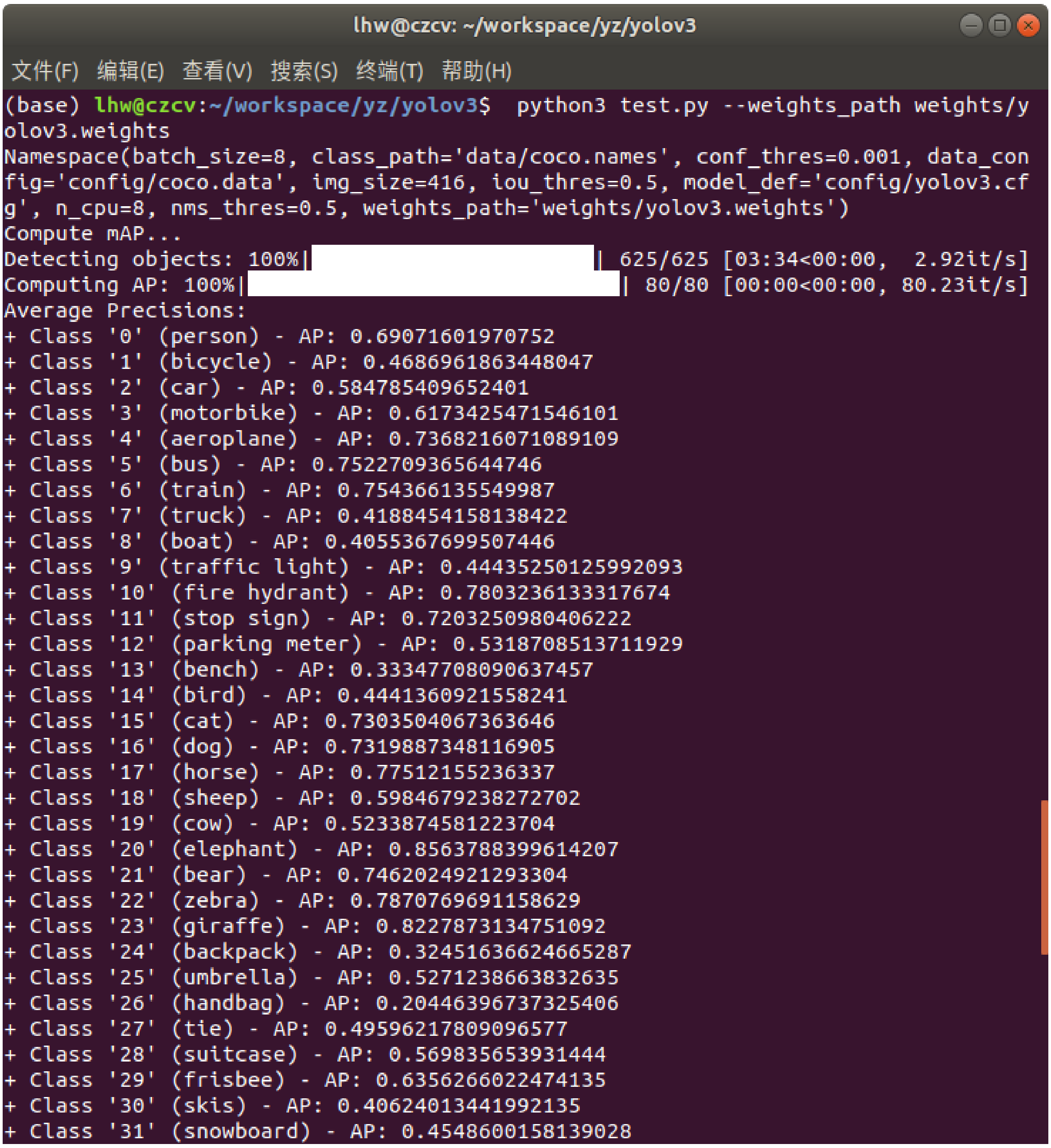
person
person
person
person
person


person
person
person
tie
Download from
Dreamstime.com
This watermarked comp image is for previewing purposes only.
Dmitriy Shironosov | Dreamstime.com

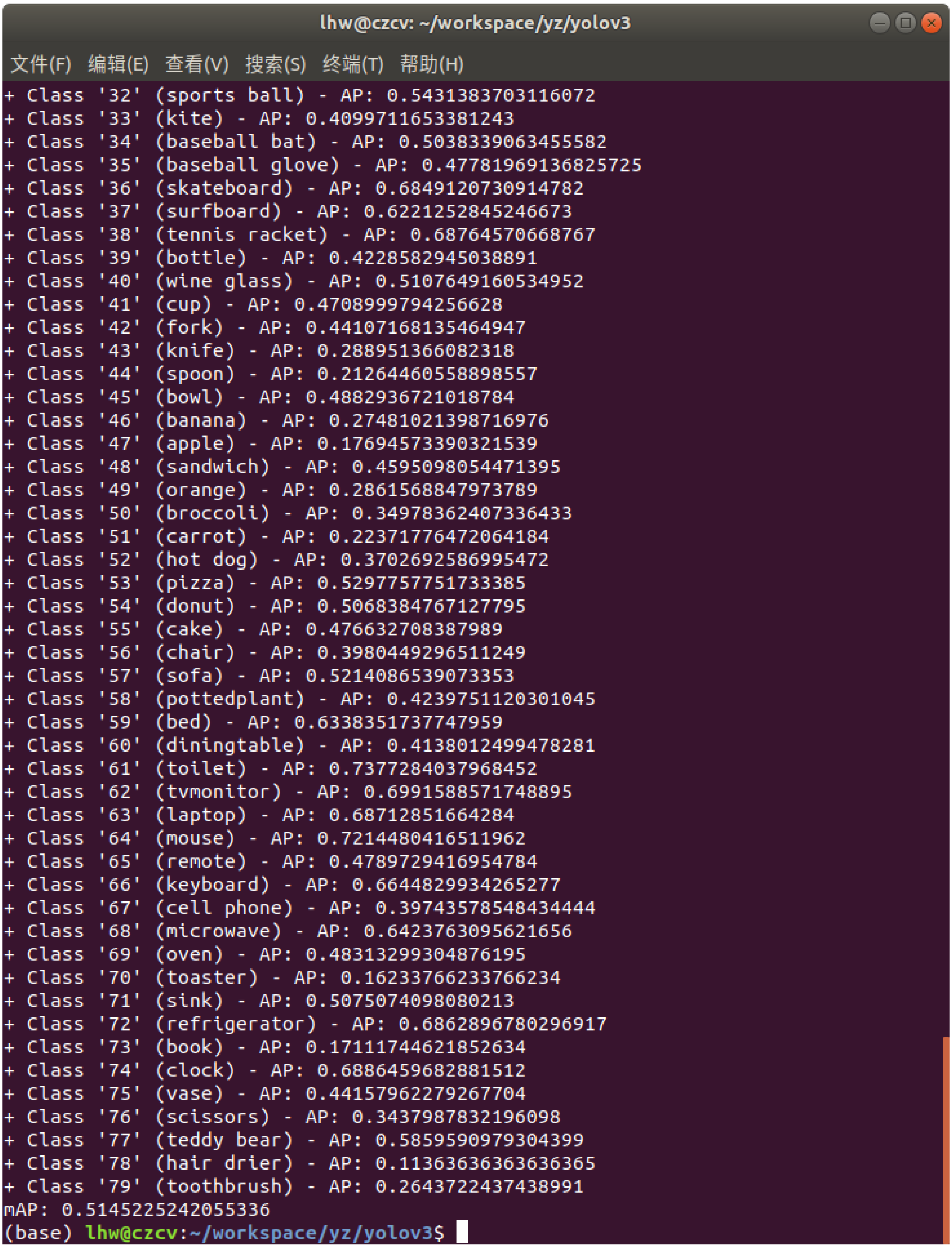

person
person
car
Download from
Dreamstime.com
This watermarked comp image is for previewing purposes only.
Hongqi Zhang &#x28;aka Michael Zhang&#x29; | Dreamstime.com

测试

```
python3 test.py --weights_path weights/yolov3.weights
```


lhw@czcv: ~/workspace/yz/yolov3
文件(F) 编辑(E) 查看(V) 搜索(S) 终端(T) 帮助(H)

```
(base) lhw@czcv:~/workspace/yz/yolov3$  python3 test.py --weights_path weights/y
olov3.weights
Namespace(batch_size=8, class_path='data/coco.names', conf_thres=0.001, data_con
fig='config/coco.data', img_size=416, iou_thres=0.5, model_def='config/yolov3.cf
g', n_cpu=8, nms_thres=0.5, weights_path='weights/yolov3.weights')
Compute mAP...
Detecting objects: 100%|██████████████████████| 625/625 [03:34<00:00,  2.92it/s]
Computing AP: 100%|█████████████████████████████| 80/80 [00:00<00:00, 80.23it/s]
Average Precisions:
+ Class '0' (person) - AP: 0.69071601970752
+ Class '1' (bicycle) - AP: 0.4686961863448047
+ Class '2' (car) - AP: 0.584785409652401
+ Class '3' (motorbike) - AP: 0.6173425471546101
+ Class '4' (aeroplane) - AP: 0.7368216071089109
+ Class '5' (bus) - AP: 0.7522709365644746
+ Class '6' (train) - AP: 0.754366135549987
+ Class '7' (truck) - AP: 0.4188454158138422
+ Class '8' (boat) - AP: 0.4055367699507446
+ Class '9' (traffic light) - AP: 0.44435250125992093
+ Class '10' (fire hydrant) - AP: 0.7803236133317674
+ Class '11' (stop sign) - AP: 0.7203250980406222
+ Class '12' (parking meter) - AP: 0.5318708513711929
+ Class '13' (bench) - AP: 0.33347708090637457
+ Class '14' (bird) - AP: 0.4441360921558241
+ Class '15' (cat) - AP: 0.7303504067363646
+ Class '16' (dog) - AP: 0.7319887348116905
+ Class '17' (horse) - AP: 0.77512155236337
+ Class '18' (sheep) - AP: 0.5984679238272702
+ Class '19' (cow) - AP: 0.5233874581223704
+ Class '20' (elephant) - AP: 0.8563788399614207
+ Class '21' (bear) - AP: 0.7462024921293304
+ Class '22' (zebra) - AP: 0.787076969115862 9
+ Class '23' (giraffe) - AP: 0.8227873134751092
+ Class '24' (backpack) - AP: 0.32451636624665287
+ Class '25' (umbrella) - AP: 0.5271238663832635
+ Class '26' (handbag) - AP: 0.20446396737325406
+ Class '27' (tie) - AP: 0.49596217809096577
+ Class '28' (suitcase) - AP: 0.569835653931444
+ Class '29' (frisbee) - AP: 0.635626602247 4135
+ Class '30' (skis) - AP: 0.40624013441992135
+ Class '31' (snowboard) - AP: 0.4548600158139028
```


lhw@czcv: ~/workspace/yz/yolov3
文件(F) 编辑(E) 查看(V) 搜索(S) 终端(T) 帮助(H)

```
+ Class '32' (sports ball) - AP: 0.5431383703116072
+ Class '33' (kite) - AP: 0.4099711653381243
+ Class '34' (baseball bat) - AP: 0.5038339063455582
+ Class '35' (baseball glove) - AP: 0.47781969136825725
+ Class '36' (skateboard) - AP: 0.6849120730914782
+ Class '37' (surfboard) - AP: 0.6221252845246673
+ Class '38' (tennis racket) - AP: 0.68764570668767
+ Class '39' (bottle) - AP: 0.4228582945038891
+ Class '40' (wine glass) - AP: 0.5107649160534952
+ Class '41' (cup) - AP: 0.4708999794256628
+ Class '42' (fork) - AP: 0.44107168135464947
+ Class '43' (knife) - AP: 0.288951366082318
+ Class '44' (spoon) - AP: 0.21264460558898557
+ Class '45' (bowl) - AP: 0.4882936721018784
+ Class '46' (banana) - AP: 0.27481021398716976
+ Class '47' (apple) - AP: 0.17694573390321539
+ Class '48' (sandwich) - AP: 0.4595098054471395
+ Class '49' (orange) - AP: 0.2861568847973789
+ Class '50' (broccoli) - AP: 0.34978362407336433
+ Class '51' (carrot) - AP: 0.22371776472064184
+ Class '52' (hot dog) - AP: 0.3702692586995472
+ Class '53' (pizza) - AP: 0.5297757751733385
+ Class '54' (donut) - AP: 0.5068384767127795
+ Class '55' (cake) - AP: 0.476632708387989
+ Class '56' (chair) - AP: 0.3980449296511249
+ Class '57' (sofa) - AP: 0.5214086539073353
+ Class '58' (pottedplant) - AP: 0.4239751120301045
+ Class '59' (bed) - AP: 0.6338351737747959
+ Class '60' (diningtable) - AP: 0.4138012499478281
+ Class '61' (toilet) - AP: 0.7377284037968452
+ Class '62' (tvmonitor) - AP: 0.6991588571748895
+ Class '63' (laptop) - AP: 0.68712851664284
+ Class '64' (mouse) - AP: 0.7214480416511962
+ Class '65' (remote) - AP: 0.4789729416954784
+ Class '66' (keyboard) - AP: 0.6644829934265277
+ Class '67' (cell phone) - AP: 0.39743578548434444
+ Class '68' (microwave) - AP: 0.6423763095621656
+ Class '69' (oven) - AP: 0.48313299304876195
+ Class '70' (toaster) - AP: 0.16233766233766234
+ Class '71' (sink) - AP: 0.5075074098080213
+ Class '72' (refrigerator) - AP: 0.6862896780296917
+ Class '73' (book) - AP: 0.17111744621852634
+ Class '74' (clock) - AP: 0.6886459682881512
+ Class '75' (vase) - AP: 0.44157962279267704
+ Class '76' (scissors) - AP: 0.3437987832196098
+ Class '77' (teddy bear) - AP: 0.5859590979304399
+ Class '78' (hair drier) - AP: 0.11363636363636365
+ Class '79' (toothbrush) - AP: 0.2643722437438991
mAP: 0.5145225242055336
(base) lhw@czcv:~/workspace/yz/yolov3$ 
```

coco数据集训练

利用imagenet预训练权重能加快收敛速度：

```
python3 train.py --data_config config/coco.data --model_def config/yolov3.cfg --pretrained_weights weights/darknet53.conv.74
```

遇到的问题

AttributeError: module 'tensorflow' has no attribute 'Summary'

解决方案，tensorflow降级至1.13.1

torch.uint8类型被弃用引发的警告

警告如下，它会影响训练过程中log的观察

[W IndexingUtils.h:20] Warning: indexing with dtype torch.uint8 is now deprecated, please use a dtype torch.bool instead. (function expandTensors)

解决方法：

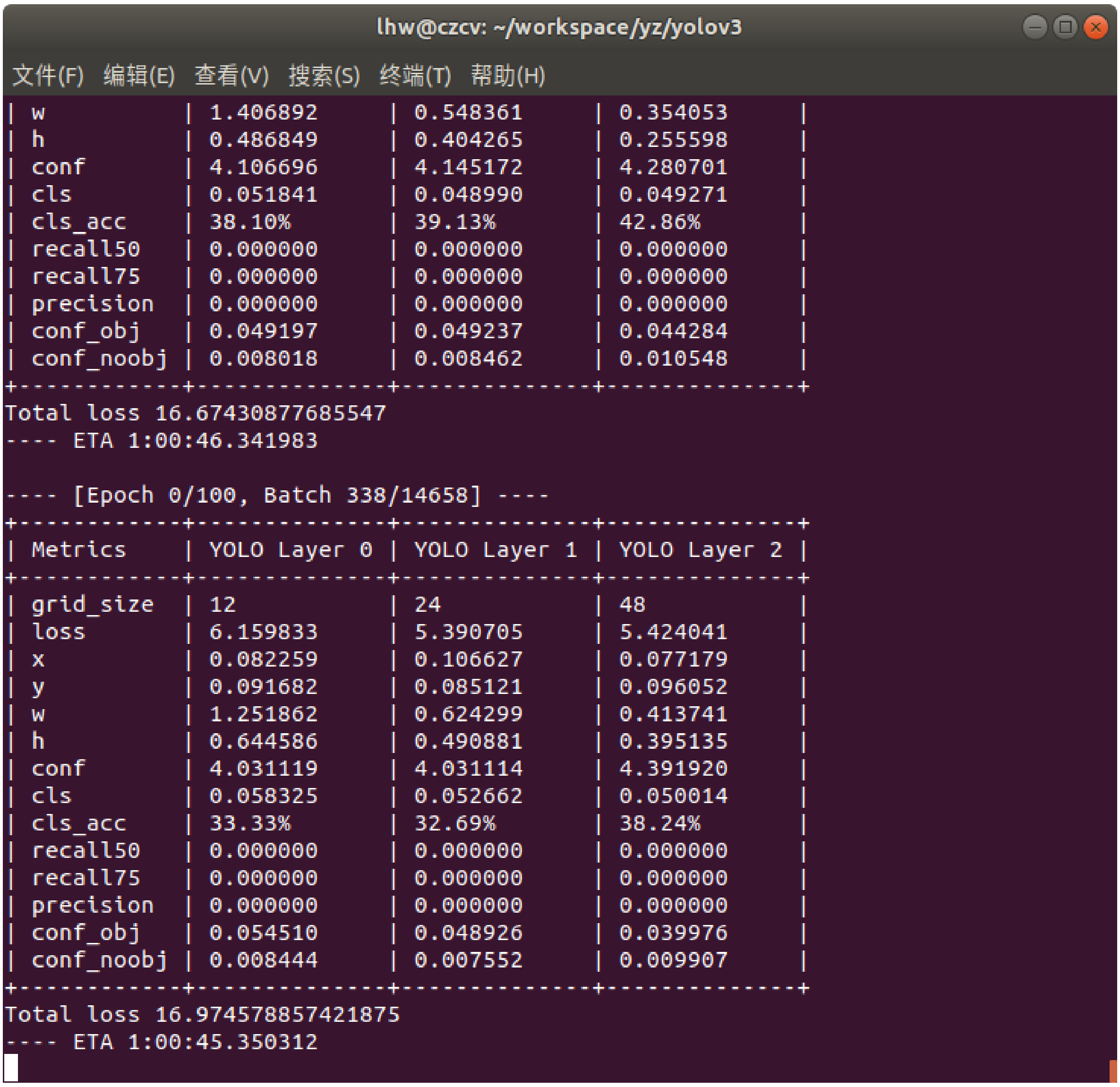
打开models.py,找到如下代码段：

```
iou_scores, class_mask, obj_mask, noobj_mask, tx, ty, tw, th, tcls, tconf = build_targets(
  pred_boxes=pred_boxes,          # (b, 3, 13, 13, 4)
  pred_cls=pred_cls,              # (b, 3, 13, 13, 80)
  target=targets,                 # (n_boxes, 6) [details in build_targets function]
  anchors=self.scaled_anchors,      # (3, 2) 3个anchor，每个2维
  ignore_thres=self.ignore_thres,   # 0.5 (hard code in YOLOLayer self.init())
)
```

后面添加：

```
obj_mask = obj_mask.bool()
noobj_mask = noobj_mask.bool()
```

训练过程可视化


lhw@czcv: ~/workspace/yz/yolov3
文件(F) 编辑(E) 查看(V) 搜索(S) 终端(T) 帮助(H)

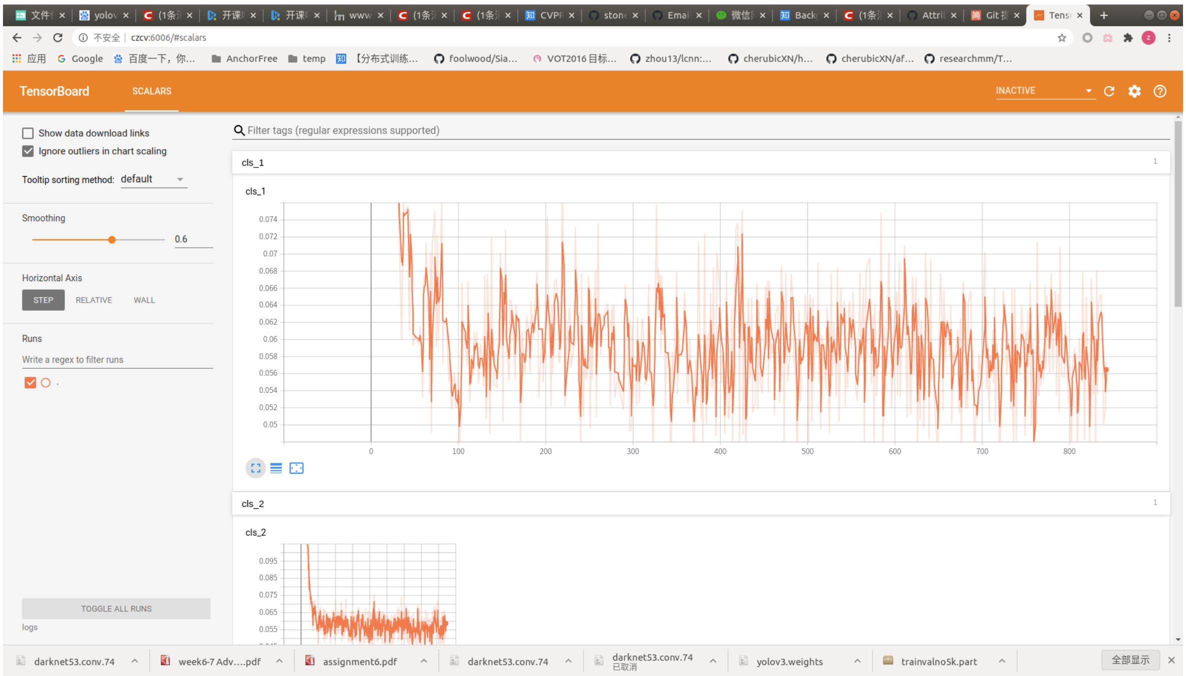
```
| w            | 1.406892     | 0.548361     | 0.354053     |
| h            | 0.486849     | 0.404265     | 0.255598     |
| conf         | 4.106696     | 4.145172     | 4.280701     |
| cls          | 0.051841     | 0.048990     | 0.049271     |
| cls_acc      | 38.10%       | 39.13%       | 42.86%       |
| recall50     | 0.000000     | 0.000000     | 0.000000     |
| recall75     | 0.000000     | 0.000000     | 0.000000     |
| precision    | 0.000000     | 0.000000     | 0.000000     |
| conf_obj     | 0.049197     | 0.049237     | 0.044284     |
| conf_noobj   | 0.008018     | 0.008462     | 0.010548     |
+--------------+--------------+--------------+--------------+
Total loss 16.67430877685547
---- ETA 1:00:46.341983

---- [Epoch 0/100, Batch 338/14658] ----
+--------------+--------------+--------------+--------------+
| Metrics      | YOLO Layer 0 | YOLO Layer 1 | YOLO Layer 2 |
+--------------+--------------+--------------+--------------+
| grid_size    | 12           | 24           | 48           |
| loss         | 6.159833     | 5.390705     | 5.424041     |
| x            | 0.082259     | 0.106627     | 0.077179     |
| y            | 0.091682     | 0.085121     | 0.096052     |
| w            | 1.251862     | 0.624299     | 0.413741     |
| h            | 0.644586     | 0.490881     | 0.395135     |
| conf         | 4.031119     | 4.031114     | 4.391920     |
| cls          | 0.058325     | 0.052662     | 0.050014     |
| cls_acc      | 33.33%       | 32.69%       | 38.24%       |
| recall50     | 0.000000     | 0.000000     | 0.000000     |
| recall75     | 0.000000     | 0.000000     | 0.000000     |
| precision    | 0.000000     | 0.000000     | 0.000000     |
| conf_obj     | 0.054510     | 0.048926     | 0.039976     |
| conf_noobj   | 0.008444     | 0.007552     | 0.009907     |
+--------------+--------------+--------------+--------------+
Total loss 16.974578857421875
---- ETA 1:00:45.350312
```

tensorboard

项目根目录执行：

```
tensorboard --logdir='logs' --port=6006
```

打开浏览器，输入[host_name]:6006


czcv:6006/#scalars

TensorBoard SCALARS INACTIVE

☐ Show data download links
☑ Ignore outliers in chart scaling
Tooltip sorting method: default

Smoothing 0.6

Horizontal Axis: STEP RELATIVE WALL

Runs
Write a regex to filter runs
☑ ○ .

TOGGLE ALL RUNS
logs

Filter tags (regular expressions supported)

cls_1

cls_2

darknet53.conv.74 week6-7 Adv....pdf assignment6.pdf darknet53.conv.74 darknet53.conv.74 已取消 yolov3.weights trainvalno5k.part 全部显示

mask 数据集训练

data prepare

xml标注转txt

xml2yolotxt.py

```
import os
from lxml import etree as ET
from lxml.etree import Element, SubElement, tostring, ElementTree
from tqdm import tqdm
import pdb

'''
@param[in]: size. 图片尺寸 [w, h]
@param[in]: box. roi区域 [x0,y0,x1,y1]
@param[out]: 归一化rect info. [normal_center_x, normal_center_y, normal_w, normal_h]
'''
def convert(size, box):
    dw = 1. / (size[0])
    dh = 1. / (size[1])
    cx = (box[0] + box[1]) / 2.0
    cy = (box[2] + box[3]) / 2.0
    w = box[1] - box[0]
    h = box[3] - box[2]
    ncx = cx * dw
    w = w * dw
    ncy = cy * dh
    h = h * dh
    return (ncx, ncy, w, h)


def convert_annotation(image_add, in_path, out_path, classes):
    image_name = os.path.basename(image_add)
    image_name = image_name.replace('.jpg', '')
    in_file = open(os.path.join(in_path, image_name + '.xml'), 'r')
    out_file = open(os.path.join(out_path, image_name+'.txt'), 'w')

    tree = ET.parse(in_file)
    root = tree.getroot()

    size = root.find('size')

    w = int(size.find('width').text)
    h = int(size.find('height').text)

    # 在一个XML中每个Object的迭代
    for obj in root.iter('object'):
        # iter()方法可以递归遍历元素/树的所有子元素
        difficult = obj.find('difficult').text
        cls = obj.find('name').text

        # cls_set.add(cls)

        # 如果训练标签中的品种不在程序预定品种，或者difficult = 1，跳过此object
        if cls not in classes or int(difficult) == 1:
            continue
        cls_id = classes.index(cls)  # 这里取索引，避免类别名是中文，之后运行yolo时要在cfg将索引与具体类别配对
        xmlbox = obj.find('bndbox')

        b = (float(xmlbox.find('xmin').text), float(xmlbox.find('xmax').text), float(
            xmlbox.find('ymin').text), float(xmlbox.find('ymax').text))
        bb = convert((w, h), b)
        out_file.write(str(cls_id) + " " + " ".join([str(a) for a in bb]) + '\n')


if __name__ == '__main__':
    classes = ['face', 'face_mask']

    data_root_folder = '/home/lhw/czcv_2t_workspace/yz/face_mask'
    img_folder = os.path.join(data_root_folder, 'JPEGImages')
    xml_folder = os.path.join(data_root_folder, 'Annotations')
    yolotxt_folder = os.path.join(data_root_folder, 'labels')

    if not os.path.exists(yolotxt_folder):
        os.makedirs(yolotxt_folder)

    img_lst_file = os.path.join(data_root_folder, 'img.txt')  # 所有图片数据（绝对路径）
    if not os.path.exists(img_lst_file):
        img_lst = [os.path.join(img_folder, x) for x in os.listdir(img_folder) if 'jpg' in x]
        with open(img_lst_file, 'w') as f:
            for img_file in img_lst:
                f.write("{}\n".format(img_file))

    else:
        with open(img_lst_file, 'r') as f:
            img_lst = f.readlines()
            img_lst = [x.split('\n')[0] for x in img_lst]

    print('converting...')
    for image_add in tqdm(img_lst):
        image_add = image_add.strip()
        convert_annotation(image_add, xml_folder, yolotxt_folder, classes)

    print("Finished")
```

划分训练集和测试集

divide_train_val.py

```
import os
import numpy as np
import cv2
import random
from tqdm import tqdm

if __name__ == '__main__':
    random.seed(0)

    data_root_folder = '/home/lhw/czcv_2t_workspace/yz/face_mask'
    img_lst_file = os.path.join(data_root_folder, 'img.txt')  # 所有图片数据（绝对路径）
    train_lst_file = os.path.join(data_root_folder, 'train.txt')
    val_lst_file = os.path.join(data_root_folder, 'val.txt')
    train_fd = open(train_lst_file, 'w')
    val_fd = open(val_lst_file, 'w')


    with open(img_lst_file, 'r') as f:
        img_lst = f.readlines()
    img_lst = [x for x in img_lst]
    random.shuffle(img_lst)


    for img_file in tqdm(img_lst):
        if random.random() < 0.7:
            train_fd.write('{}'.format(img_file))
        else:
            val_fd.write('{}'.format(img_file))

    train_fd.close()
    val_fd.close()

    print('finish')
```

修改相关配置文件

组织如下结构的数据，images和labels文件夹较大，可使用软连接：

```
{yolov3_root_path}/data/mask
                            ├── classes.names
                            ├── images -> /home/lhw/czcv_2t_workspace/yz/face_mask/JPEGImages
                            ├── img.txt
                            ├── labels -> /home/lhw/czcv_2t_workspace/yz/face_mask/labels
                            ├── train.txt
                            └── valid.txt
```

建立classes.names，内容如下

```
face
face_mask
```

config目录下建立mask.data，内容设置为

```
classes=2
train=data/mask/train.txt
valid=data/mask/valid.txt
names=data/mask/classes.names
```

训练

```
python3 train.py --data_config config/mask.data --model_def config/yolov3.cfg --pretrained_weights weights/darknet53.conv.74
```