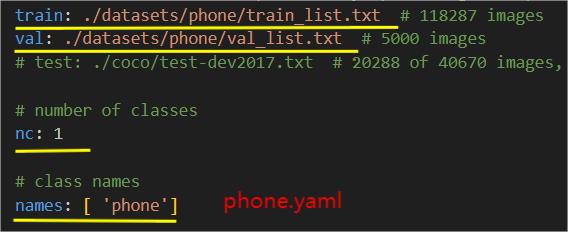
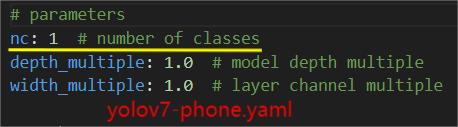
打电话识别检测的文件说明

**train.py：**手机检测部分的训练。

·使用yolo数据集训练了1000+张图片，其中train包含897张图片，valid包含224张图片。

·训练结果保存至"runs\train\exp"。训练参数中, data.yaml文件改为'data/phone.yaml' ; cfg文件改为'cfg/training/yolov7-phone.yaml' ，文件中训练类别改为’phone’，类别数改为1，更改图片路径，具体如下：

|  |
| --- |
| 终端指令：  python train.py --workers 5 --device 0 --batch-size 8 --data data/phone.yaml --img 640 640 --cfg cfg/training/yolov7-phone.yaml --weights '' --name yolov7 --hyp data/hyp.scratch.p5.yaml |

**pose.py:**用于人体姿态检测，并将耳朵可疑点的坐标映射到原图。

关键点说明：关键点有17个：[“nose”,“left\_eye”, “right\_eye”,“left\_ear”, “right\_ear”,“left\_shoulder”, “right\_shoulder”,“left\_elbow”, “right\_elbow”,“left\_wrist”, “right\_wrist”,“left\_hip”, “right\_hip”,“left\_knee”, “right\_knee”,“left\_ankle”, “right\_ankle”]

函数说明：

1. distance(a,b)：计算a,b两点的距离

2）scale\_coordsscale\_coords(img1\_shape, coords, img0\_shape, ratio\_pad=None)：将坐标coords映射到原图im0上，要求coords为torch张量而非numpy数组。其中img1\_shape为缩放后的图片，im0为原始图。

3）clip\_coords(boxes, img\_shape)：将输入的坐标数组裁剪到图像范围内。（*自定义，区别于detect.py中的）*

4）pose\_detect(im0,device)：

·使用yolov7-w6-pose.pt进行关键点检测

·用letterbox函数自适应调整图片大小和填充图像(im0->image)

·使用关键点检测模型得到结果output；非极大值抑制函数non\_max\_suppression\_kpt确保算法对每个对象只检测一次；

·output\_to\_keypoint函数将模型的输出转换为我们想要的格式，即[batch\_id,class\_id,x,y,w,h,conf]

·通过scale\_coords函数将关键点映射到原图上。其中每个关键点的坐标为(x,y,conf置信度)

·以两侧肩膀到耳朵的距离max作为L。

·对于同侧的手腕、肩膀和耳朵，若耳朵到手腕的水平距离x<1.2L,手腕到肩膀的竖直距离y<0.65L或者手腕在耳朵与肩膀之间，则将耳朵定义为可疑检测点，将[ (X\_ear,Y\_ear),Y\_wrist]

加入可疑点数组

·返回可疑点数组。

**detect.py:**用于效果展示，框出打电话行为的手机。

具体解读同yolov5的detect.py相似：<https://www.iotword.com/3300.html>

1）函数说明：

·LoadImages，LoadStreams：读取照片或者视频，并进行处理，包括图片缩放、BGR to RGB

·non\_max\_suppression：非极大值抑制，进行预测框筛选，确保算法对每个对象只检测一次；

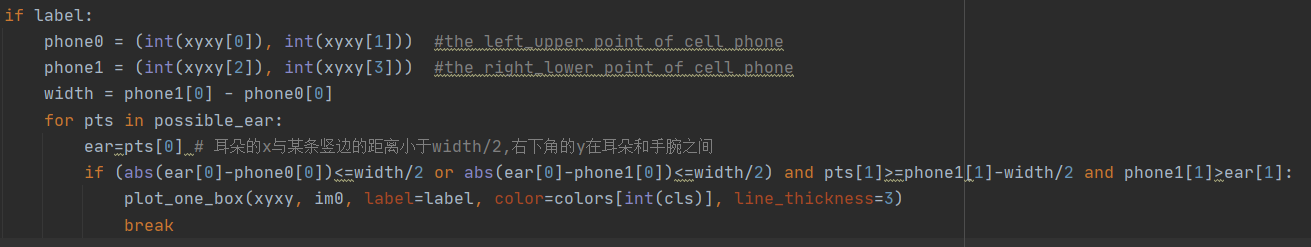
·scale\_coords：将预测信息映射到原图。

·plot\_one\_box：在原图上画框，并显示类别标签

1. 修改部分（120行、131-140行）：



131-140行：



·使用训练参数"runs\train\exp\weights\best.pt"进行手机检测,label作为类别标签(这里只有phone）。

·通过关键点检测（120行）得到打电话行为的可疑点数组possible\_ear

·以手机预测框的左上角为phone0，右下角为phone1；以手机宽度为width;对于possible\_ear中的每个组pts=[ (X\_ear,Y\_ear),Y\_wrist]，则ear=(X\_ear,Y\_ear)，pts[1]=Y\_wrist

·若手机与某可疑耳朵点的水平距离d<=0.5width且手机右下角**大致**在耳朵与手腕之间，则判断为打电话的行为，框出该手机。（由于拍摄角度及预测准确性问题，手机右下角可能会出现在手腕之下，因此扩大距离判断范围：pts[1]>=phone1[1]-width/2）

·结果存至runs/detect

|  |
| --- |
| 终端指令：  python detect.py --weights runs/train/exp/weights/best.pt --conf 0.25 --img-size 640 --source inference/phone |