计算机视觉第十次作业 - YOLO

姓名：岳东旭　　学号：2201212864　　指导老师：张健

1.问题描述

- 下载运行YOLOv4(YOLOv5).py代码，测试5幅图
- 文档中说明跟之前版本的具体改进和不同

2.下载运行截图

2.1环境安装

安装所需依赖：

```
(base) PS D:\jupyterlab\HW\yolov5> pip install -r requirements.txt
WARNING: Ignore distutils configs in setup.cfg due to encoding errors.
Requirement already satisfied: gitpython in d:\anaconda\lib\site-packages (from -r requirements.txt (line 5)) (3.1.29)
Requirement already satisfied: ipython in d:\anaconda\lib\site-packages (from -r requirements.txt (line 6)) (7.31.1)
Requirement already satisfied: matplotlib>=3.2.2 in d:\anaconda\lib\site-packages (from -r requirements.txt (line 7)) (3.5.2)
Requirement already satisfied: numpy>=1.18.5 in d:\anaconda\lib\site-packages (from -r requirements.txt (line 8)) (1.23.4)
Requirement already satisfied: opencv-python>=4.1.1 in d:\anaconda\lib\site-packages (from -r requirements.txt (line 9)) (4.6.0.66)
Requirement already satisfied: Pillow>=7.1.2 in d:\anaconda\lib\site-packages (from -r requirements.txt (line 10)) (9.2.0)
Requirement already satisfied: psutil in d:\anaconda\lib\site-packages (from -r requirements.txt (line 11)) (5.9.0)
Requirement already satisfied: PyYAML>=5.3.1 in d:\anaconda\lib\site-packages (from -r requirements.txt (line 12)) (6.0)
Requirement already satisfied: requests>=2.23.0 in d:\anaconda\lib\site-packages (from -r requirements.txt (line 13)) (2.28.1)
Requirement already satisfied: scipy>=1.4.1 in d:\anaconda\lib\site-packages (from -r requirements.txt (line 14)) (1.9.1)
Collecting thop>=0.1.1
  Downloading thop-0.1.1.post2209072238-py3-none-any.whl (15 kB)
Requirement already satisfied: torch>=1.7.0 in d:\anaconda\lib\site-packages (from -r requirements.txt (line 16)) (1.12.1)
Requirement already satisfied: torchvision>=0.8.1 in d:\anaconda\lib\site-packages (from -r requirements.txt (line 17)) (0.13.1)
Requirement already satisfied: tqdm>=4.64.0 in d:\anaconda\lib\site-packages (from -r requirements.txt (line 18)) (4.64.1)
Requirement already satisfied: tensorboard>=2.4.1 in d:\anaconda\lib\site-packages (from -r requirements.txt (line 22)) (2.10.1)
Requirement already satisfied: pandas>=1.1.4 in d:\anaconda\lib\site-packages (from -r requirements.txt (line 27)) (1.4.4)
Requirement already satisfied: seaborn>=0.11.0 in d:\anaconda\lib\site-packages (from -r requirements.txt (line 28)) (0.11.2)
Requirement already satisfied: gitdb<5,>=4.0.1 in d:\anaconda\lib\site-packages (from gitpython->-r requirements.txt (line 5)) (4.0.9)
Requirement already satisfied: colorama in d:\anaconda\lib\site-packages (from ipython->-r requirements.txt (line 6)) (0.4.5)
Requirement already satisfied: backcall in d:\anaconda\lib\site-packages (from ipython->-r requirements.txt (line 6)) (0.2.0)
Requirement already satisfied: decorator in d:\anaconda\lib\site-packages (from ipython->-r requirements.txt (line 6)) (5.1.1)
Requirement already satisfied: pygments in d:\anaconda\lib\site-packages (from ipython->-r requirements.txt (line 6)) (2.11.2)
Requirement already satisfied: pickleshare in d:\anaconda\lib\site-packages (from ipython->-r requirements.txt (line 6)) (0.7.5)
Requirement already satisfied: jedi>=0.16 in d:\anaconda\lib\site-packages (from ipython->-r requirements.txt (line 6)) (0.18.1)
Requirement already satisfied: prompt-toolkit!=3.0.0,!=3.0.1,<3.1.0,>=2.0.0 in d:\anaconda\lib\site-packages (from ipython->-r requirements.txt (line 6)) (3.0.20)
Requirement already satisfied: setuptools>=18.5 in d:\anaconda\lib\site-packages (from ipython->-r requirements.txt (line 6)) (63.4.1)
Requirement already satisfied: matplotlib-inline in d:\anaconda\lib\site-packages (from ipython->-r requirements.txt (line 6)) (0.1.6)
Requirement already satisfied: traitlets>=4.2 in d:\anaconda\lib\site-packages (from ipython->-r requirements.txt (line 6)) (5.1.1)
Requirement already satisfied: kiwisolver>=1.0.1 in d:\anaconda\lib\site-packages (from matplotlib>=3.2.2->-r requirements.txt (line 7)) (1.4.2)
Requirement already satisfied: cycler>=0.10 in d:\anaconda\lib\site-packages (from matplotlib>=3.2.2->-r requirements.txt (line 7)) (0.11.0)
Requirement already satisfied: fonttools>=4.22.0 in d:\anaconda\lib\site-packages (from matplotlib>=3.2.2->-r requirements.txt (line 7)) (4.25.0)
```

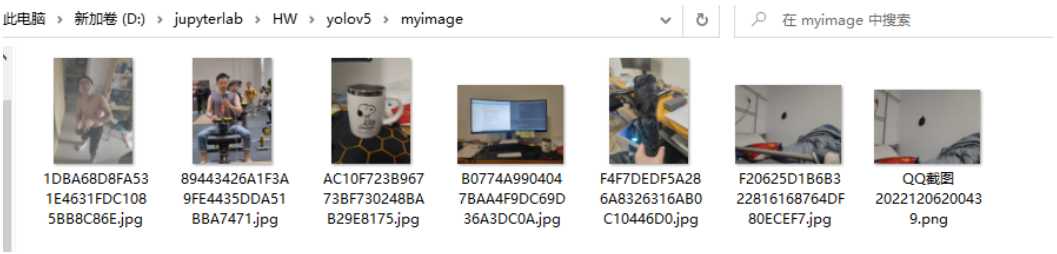
运营测试代码，顺便下载预训练模型：

```
(base) PS D:\jupyterlab\HW\yolov5> python detect.py --source Webcam
detect: weights=yolov5s.pt, source=Webcam, data=data\coco128.yaml, imgsz=[640, 640], conf_thres=0.25, iou_thres=0.45, max_det=1000, device=, view_img=False, save_txt=False, save_conf=F
alse, save_crop=False, nosave=False, classes=None, agnostic_nms=False, augment=False, visualize=False, update=False, project=runs\detect, name=exp, exist_ok=False, line_thickness=3, hi
de_labels=False, hide_conf=False, half=False, dnn=False, vid_stride=1
YOLOv5  v7.0-24-gf8539a6 Python-3.9.13 torch-1.12.1 CUDA:0 (NVIDIA TITAN X (Pascal), 12288MiB)

Downloading https://github.com/ultralytics/yolov5/releases/download/v7.0/yolov5s.pt to yolov5s.pt...
100%|████████████████████████████████████████████████████████████████████████████████████████████████████████████████████████| 14.1M/14.1M [00:10<00:00, 1.40MB/s]
Fusing layers...
YOLOv5s summary: 213 layers, 7225885 parameters, 0 gradients
```

2.2构建自己的测试数据

将待测图片放在一个文件夹内：



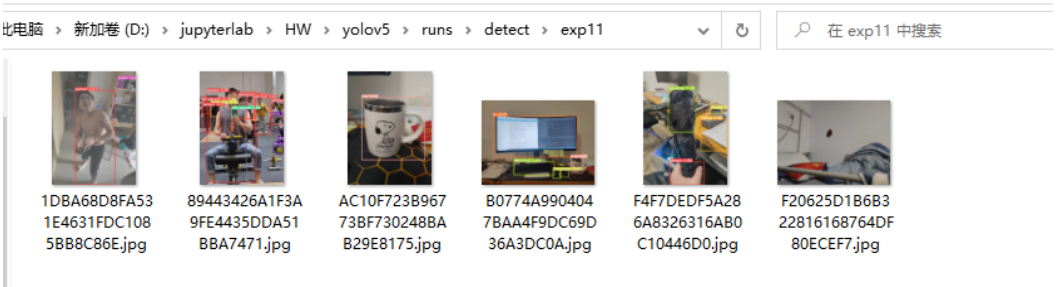
2.3运行YOLO代码得到测试结果

运行代码：

```
python detect.py --source OPTION=D:/jupyterlab/HW/yolov5/myimage
```

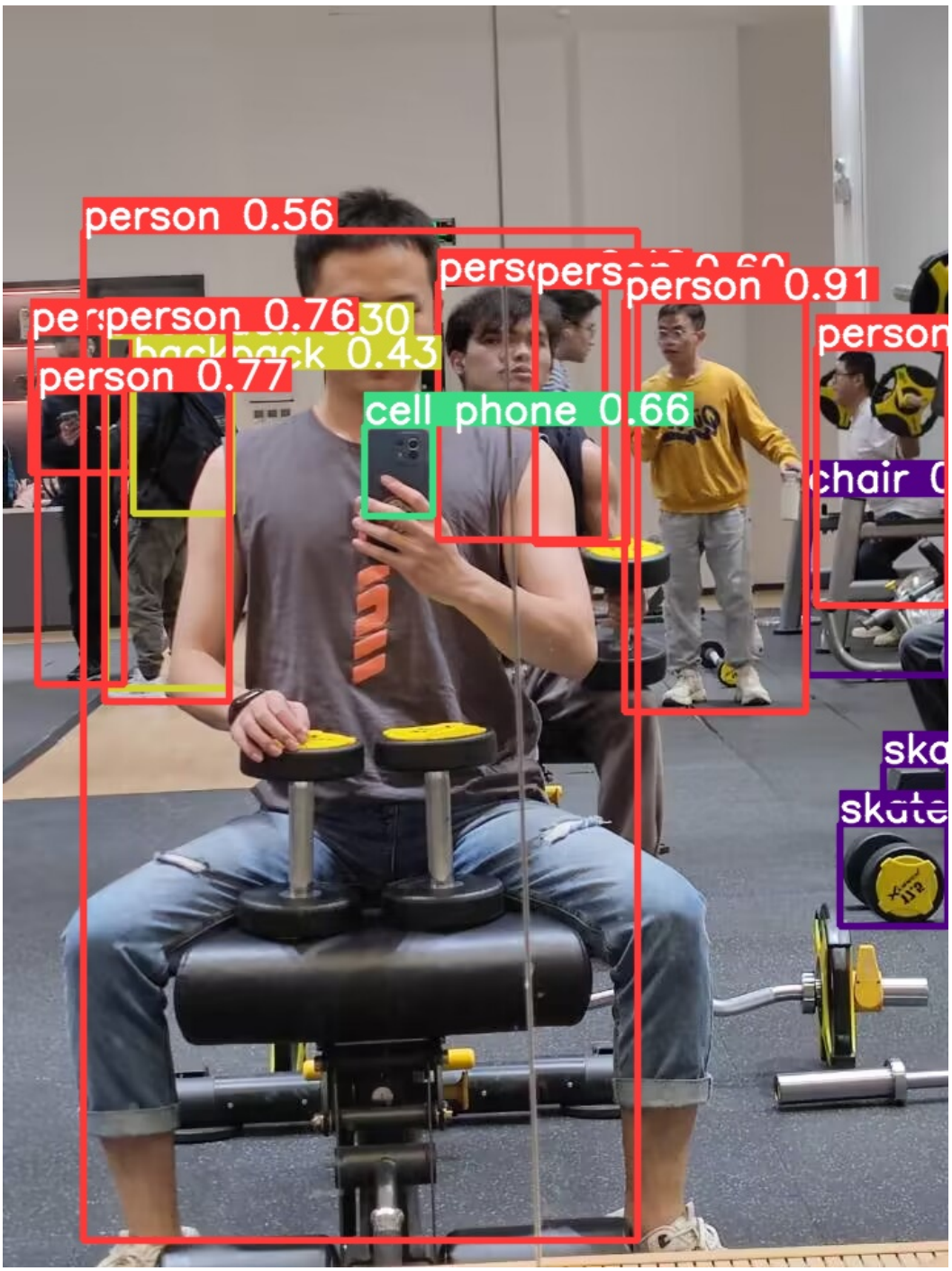
```
YOLOv5  v7.0-24-gf8539a6 Python-3.9.13 torch-1.12.1 CUDA:0 (NVIDIA TITAN X (Pascal), 12288MiB)

Fusing layers...
YOLOv5s summary: 213 layers, 7225885 parameters, 0 gradients
image 1/7 D:\jupyterlab\HW\yolov5\myimage\1DBA68D8FA531E4631FDC1085BB8C86E.jpg: 640x480 1 person, 1 bottle, 11.0ms
image 2/7 D:\jupyterlab\HW\yolov5\myimage\89443426A1F3A9FE4435DDA51BBA7471.jpg: 640x480 8 persons, 2 backpacks, 2 skateboards, 1 chair, 1 cell phone, 6.0ms
image 3/7 D:\jupyterlab\HW\yolov5\myimage\AC10F723B96773BF730248BAB29E8175.jpg: 640x480 1 cup, 7.0ms
image 4/7 D:\jupyterlab\HW\yolov5\myimage\B0774A9904047BAA4F9DC69D36A3DC0A.jpg: 480x640 1 cup, 1 tv, 2 mouses, 1 keyboard, 10.0ms
image 5/7 D:\jupyterlab\HW\yolov5\myimage\F20625D1B6B322816168764DF80ECEF7.jpg: 480x640 (no detections), 7.0ms
image 6/7 D:\jupyterlab\HW\yolov5\myimage\F4F7DEDF5A286A8326316AB0C10446D0.jpg: 640x480 1 person, 1 handbag, 1 laptop, 1 mouse, 8.0ms
image 7/7 D:\jupyterlab\HW\yolov5\myimage\QQ20221206200439.png: 480x640 (no detections), 8.0ms
Speed: 0.6ms pre-process, 8.1ms inference, 1.6ms NMS per image at shape (1, 3, 640, 640)
Results saved to runs\detect\exp
(base) PS D:\jupyterlab\HW\yolov5>
```



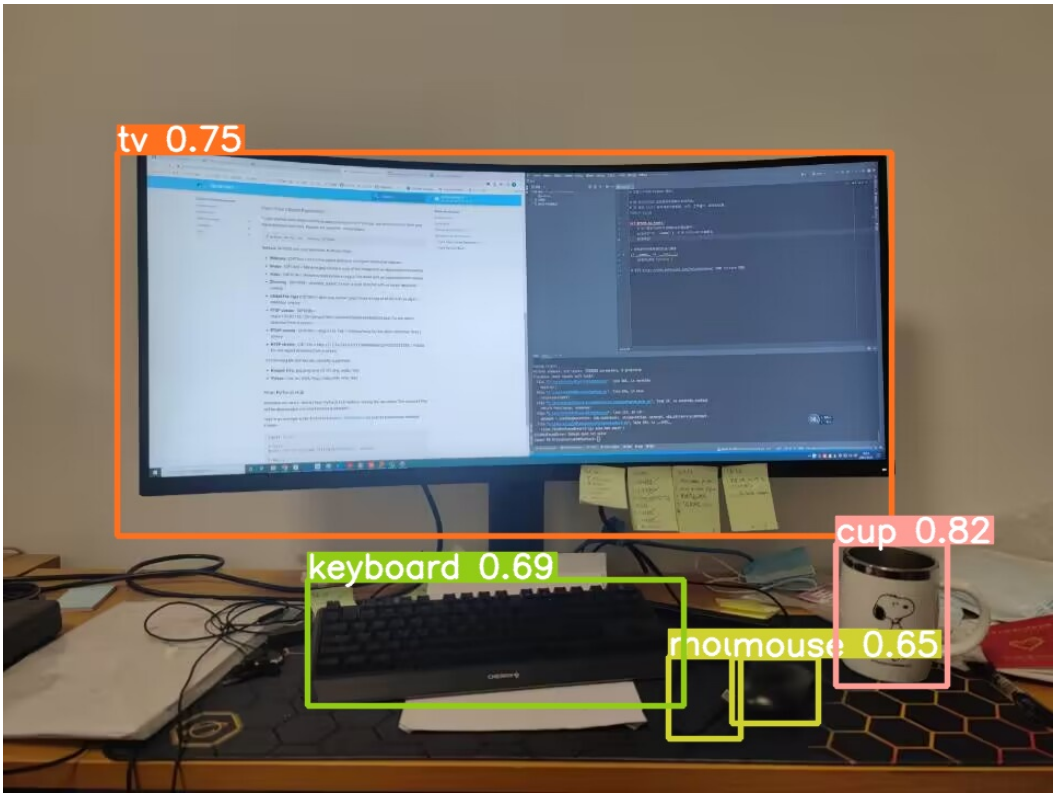
选取四张图：

pic1：健身房随手拍hh

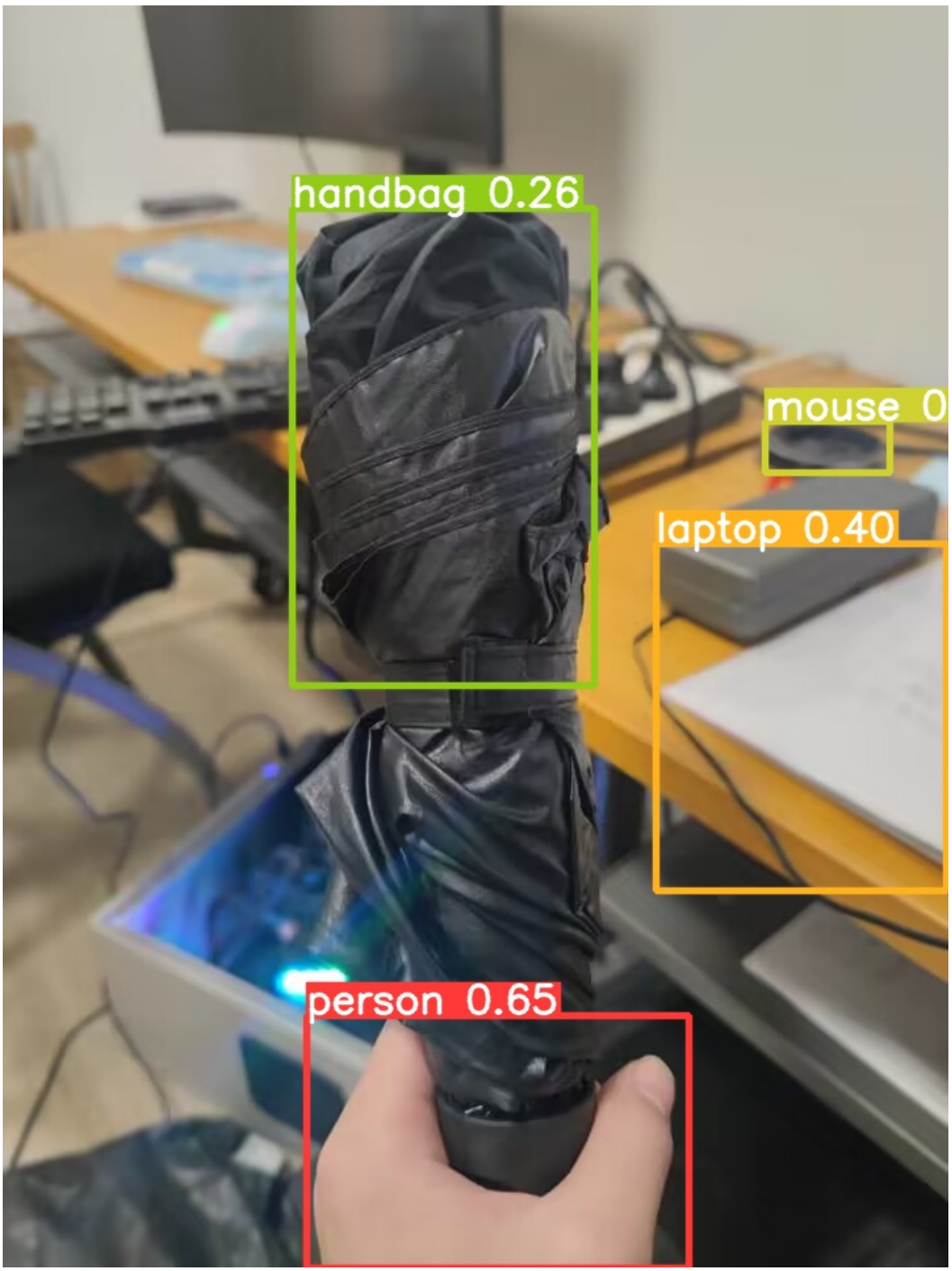


pic2：新买的马克杯

pic3：工位电脑



pic4：我的雨伞，貌似YOLO不太认识雨伞



3.与之前版本的比较

1. yolov4采用了较多的数据增强方法(图像增强方法(Data Augmentation)_YMilton的专栏-CSDN博客,博客中方法都使用)，而yolov5进行了3中数据增强：缩放、色彩空间调整与Mosaic数据增强。
2. yolov5锚点框是基于训练数据集自动学习的，而yolov4没有自适应锚点框。
3. yolov5采用的激活函数包括leakyReLU和Sigmoid，yolov5的中间隐藏层使用的是leakyReLU激活函数，最后的检测层使用的是Sigmoid激活函数。而yolov4使用的是mish与leakyReLU激活函数，主干网络使用的mish。mish激活函数的复杂度较高。
4. yolov5提供了两个优化函数Adam与SGD，并且都预设了与之匹配的训练超参数，默认使用SGD。而yolov4采用SGD优化函数。

5. yolo系列损失计算包括目标置信度、类别概率与边界框回归损失。yolov5中的边界框损失前期采用的是GIoU Loss，后期使用CIoU Loss，yolov4中采用的是CIoU Loss，与其他方法相比，CIoU带来了更快的收敛和更好的性能。

6. 目标检测在前向推理过程都会采用NMS(非极大值抑制)，yolov4在前向推理的过程中使用的方法是DIoU_nms，而yolov5采用加权nms的方式。