

训练

(1) 数据集摆放:

- VOC_Devkit
 - VOC2007
 - Annotations
 - xxx.xml
 - ImageSets
 - train.txt : 图片名
 - test.txt
 - JPEGImages
 - xxx.jpg

(2) 使用 `voc_annotation.py`文件生成训练所需的两个文件
(训练时候读取图片的路径)

直接运行, 生成两个文件(文件名可修改):

```
2007_train.txt
2007_test.txt
```

(3) 运行 `train_open.py`开始训练

- 77行 `classes_path = "open_classes.txt"` (一类) or `"open_classes_2.txt"`(两类)
- 训练两类的时候, 需要修改 `ICloss.py` 中的 237, 243行

```
for j in range(4):
```

```
    loss =
```

```
self.hingeloss(dist, torch.tensor(labels).reshape(-1, 4).cuda())
```

(4) 权重保存路径:

```
train_open.py : save_dir = 'logs'
```

推理

测试图和标签在test_code/test_imgs 文件夹中

(1) 调用 predict.py 进行推理, 检测结果保存在 result3 文件夹 (predict.py 16-18行)

可对 一类、两类开集 以及 闭集三种情况下进行推理:

- 对于一类开集, 使用 yolo.py
- 两类开集, 使用 yolo_2.py
- 闭集, 使用 yolo_close.py

(2) 调用read_annotation_open.py 获取 可视化结果

注意每一个 调用 os.listdir () 都需要排序 : 例如
img_lists.sort()