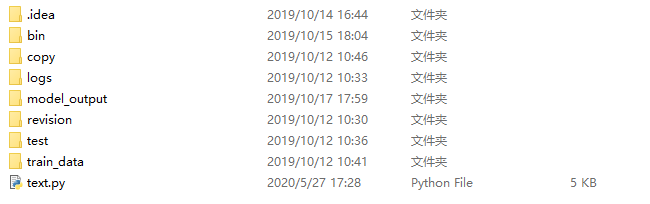
1. 算法文件夹结构说明



包含

bin目录:主程序目录，包含train.py、test.py、preproc.py三个主要使用的脚本

copy目录:不使用

logs目录:不使用

model\_output目录:包含region、screw两个目录，用于存放第一步和第二步检测训练和使用的权重文件及训练日志文件

revision目录:修改mmdet输出的脚本，具体使用在环境搭建文档有说明

test目录:存放测试数据目录，包含photo和xml目录

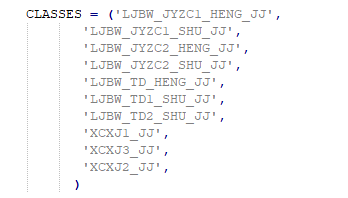
train\_data目录:存放训练数据目录，包含photo和xml目录

1. 超参数修改说明
2. 修改缺陷类别
3. 修改bin/configs目录下rcnn\_region.py、rcnn\_screw.py脚本，查找和替换num\_classes对应数值为第一步检测和第二步检测训练数据的实际类别数（类别数目需计算背景类别数量，即标注类别总数+1为实际类别数量）

**大类别：7-8**

**小类别：5-6**

1. 修改bin\mmdet\datasets目录下region.py和screw.py，在region.py脚本中修改CLASSES元组为标注类别名（备注，最后一个逗号要保留不然容易出错），在screw.py脚本中修改CLASSES元组为标注类别对应编号（注：类别内容均不算背景类别）：



region.py和screw.py内容修改

**编号从0开始数**

1. 回到bin目录下，使用python setup.py develop命令应用修改
2. preproc.py文件region和classes分别对应区域标签和目标标签，使用python preproc.py -trdata {train\_data文件夹路径} ，将数据预处理问两步骤的数据。
3. 修改gpu数量：修改bin\tools目录下train\_step1.sh、train\_step2.sh，修改GPUS数量为需要的数量

**程序运行的时候可能会卡住，重新运行，换显卡有可能解决卡住的问题，需要多尝试。**

1. 修改输入图像尺寸：修改bin/configs目录下，rcnn\_region.py和rcnn\_screw.py文件，dict(type='Resize', img\_scale=(1333, 800), keep\_ratio=True),修改图像尺寸为需要的尺寸信息

**这个图像尺寸还用改吗？**

1. 修改输入图像数量：修改bin/configs目录下，rcnn\_region.py和rcnn\_screw.py文件，数据字典data 内容：imgs\_per\_gpu=2,workers\_per\_gpu=2,修改为需要的参数

**这个是规定一个gpu上一个batch的图片？**

1. 修改学习率：修改bin/configs目录下，rcnn\_region.py#223和rcnn\_screw.py #224下的lr，修改位所需要的学习率
2. 修改anchor：修改bin/configs目录下，rcnn\_region.py#31-32和rcnn\_screw.py#30-31的anchor\_ratios和anchor\_strides. Anchor\_ratios为anchor长宽比，默认为1:2,1:1,2:1. Anchor\_strides为anchor的步长。

**这些应该是根据经验设定吧，需要改吗？**

1. 修改检测结果输出最大框数：修改bin/configs目录下，rcnn\_region.py#163和rcnn\_screw.py#162的max\_num为所需检测框最大数，降低max\_num可以提高检测速度。

**我先保持不变**

1. 修改输入尺寸：修改bin/configs 目录下，rcnn\_region.py#192和rcnn\_screw.py#191下的img\_scale=(1333, 800)。默认下将图片缩放为短边为800像素，长边等比缩放但不超过1333.

**两个图像尺寸有什么区别吗？**

1. 修改训练迭代数：修改bin/configs目录下,rcnn\_region.py#242和rcnn\_screw.py#241的total\_epochs为所需训练的最大迭代数。
2. 算法启动

环境不使用conda，使用/usr/local/bin/python3

1. 前置条件：训练数据放置在train\_data目录下，图像和标注文件分别放置在photo和xml文件夹，xml标注符合voc格式要求；类别等超参数已经按照要求进行修改 并且在bin目录下使用python setup.py develop命令进行了应用。
2. 启动数据预处理命令：python preproc.py –trdata ../train\_data
3. 启动算法训练命令：python train.py