

本周在对训练网络的代码进行调试,首先发现的问题是在多 GPU 训练时生成的 anchor 与输入的数据可能不在一块 GPU 上,解决办法是用 `torch.tensor.to` 函数来对需要与 input 的数据进行比较的数据指定 GPU,这个地方需要在数据定义的时候就进行 to 操作,否则会报错。之后发生的问题是多 gpu 训练的时候会报 tensor 在维度上有 size 不同,后面发现是因为不同样本送入的 ground truth 的个数是不同的,在 dataloader 的时候会造成如果用多 gpu batch 那一维大于 1 的时候后面的数据没法进行 stack 操作,当时的解决方法是自己写了个 collate 函数,对于每次送进来的多组 ground truth 取数量最少的那个数值,将多余的 ground truth 全部丢弃,后来想到应该取最大值,但还没有进行调整。之后调试了 roi pooling 部分,在网上找到一份 `cpp_extension` 相关的代码 <https://github.com/facebookresearch/maskrcnn-benchmark>,修改了其中的 roi pool 部分,最后编译成功能作为扩展库 import 了。现在遇到的问题是在整个网络迭代十多次后会出现 loss 暴增的情况,分析可能是由于 learning rate 设置大了,当 loss 过小以后会发生突变,目前正在验证是不是这个问题。现在的计划是通过一个只有一个视频的小样本进行测试,看看多次迭代后能否使 loss 趋向于 0,再修改前面提到的 ground truth 的输入问题,如果之后训练集能比较好的训练,开始写测试集的相关代码。