人工神经网络原理期末大作业

姓名：叶茂青

学号：17363092

a

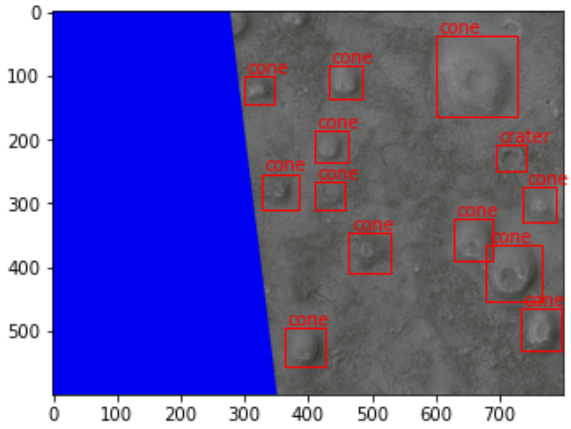
以下预测时间仅供参考，运行的设备不一样没有比较的意义

Faster RCNN实现及其结果

运行过程可见https://www.kaggle.com/clicker3/faster-rcnn

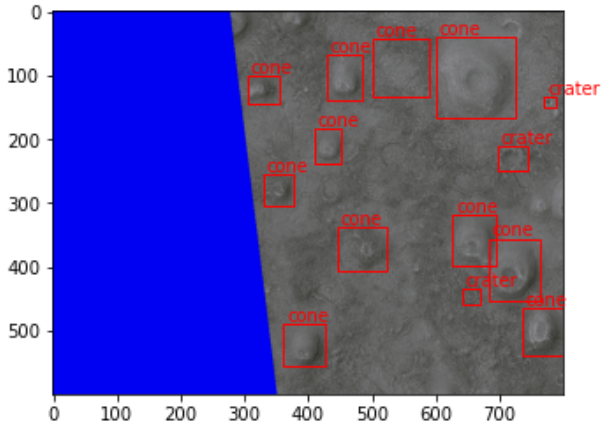
模型评测结果：

```
Average Precision  (AP) @[ IoU=0.50:0.95 | area=   all | maxDets=100 ] = 0.404
Average Precision  (AP) @[ IoU=0.50      | area=   all | maxDets=100 ] = 0.728
Average Precision  (AP) @[ IoU=0.75      | area=   all | maxDets=100 ] = 0.389
Average Precision  (AP) @[ IoU=0.50:0.95 | area= small | maxDets=100 ] = 0.067
Average Precision  (AP) @[ IoU=0.50:0.95 | area=medium | maxDets=100 ] = 0.417
Average Precision  (AP) @[ IoU=0.50:0.95 | area= large | maxDets=100 ] = 0.575
Average Recall     (AR) @[ IoU=0.50:0.95 | area=   all | maxDets=  1 ] = 0.132
Average Recall     (AR) @[ IoU=0.50:0.95 | area=   all | maxDets= 10 ] = 0.504
Average Recall     (AR) @[ IoU=0.50:0.95 | area=   all | maxDets=100 ] = 0.535
Average Recall     (AR) @[ IoU=0.50:0.95 | area= small | maxDets=100 ] = 0.067
Average Recall     (AR) @[ IoU=0.50:0.95 | area=medium | maxDets=100 ] = 0.564
Average Recall     (AR) @[ IoU=0.50:0.95 | area= large | maxDets=100 ] = 0.675
```



In [15]:

```
# 正确结果如下
image, targets = test[0]
boxes = targets["boxes"]
labels = [idx2class[l.item()] for l in targets["labels"]]
visualize.show_labeled_image(image, boxes, labels)
```



模型预测时间（在CPU上）：

```
CPU times: user 7.09 s, sys: 957 ms, total: 8.05 s
Wall time: 8.06 s
```

YOLO实现及其结果

实现参考https://github.com/eriklindernoren/PyTorch-YOLOv3，修改了部分代码以抛弃对tensorflow的依赖，另外修改了评测时期的代码

运行过程可见https://www.kaggle.com/clicker3/pytorch-yolov3

模型评测结果：

```
+-------+------------+-----------+---------+---------+
| Index | Class name | Precision | Recall  | AP      |
+-------+------------+-----------+---------+---------+
| 0     | crater     | 0.02870   | 1.00000 | 0.90520 |
| 1     | cone       | 0.02874   | 0.97727 | 0.82251 |
+-------+------------+-----------+---------+---------+
---- mAP 0.8638555662885201
```

模型预测时间：

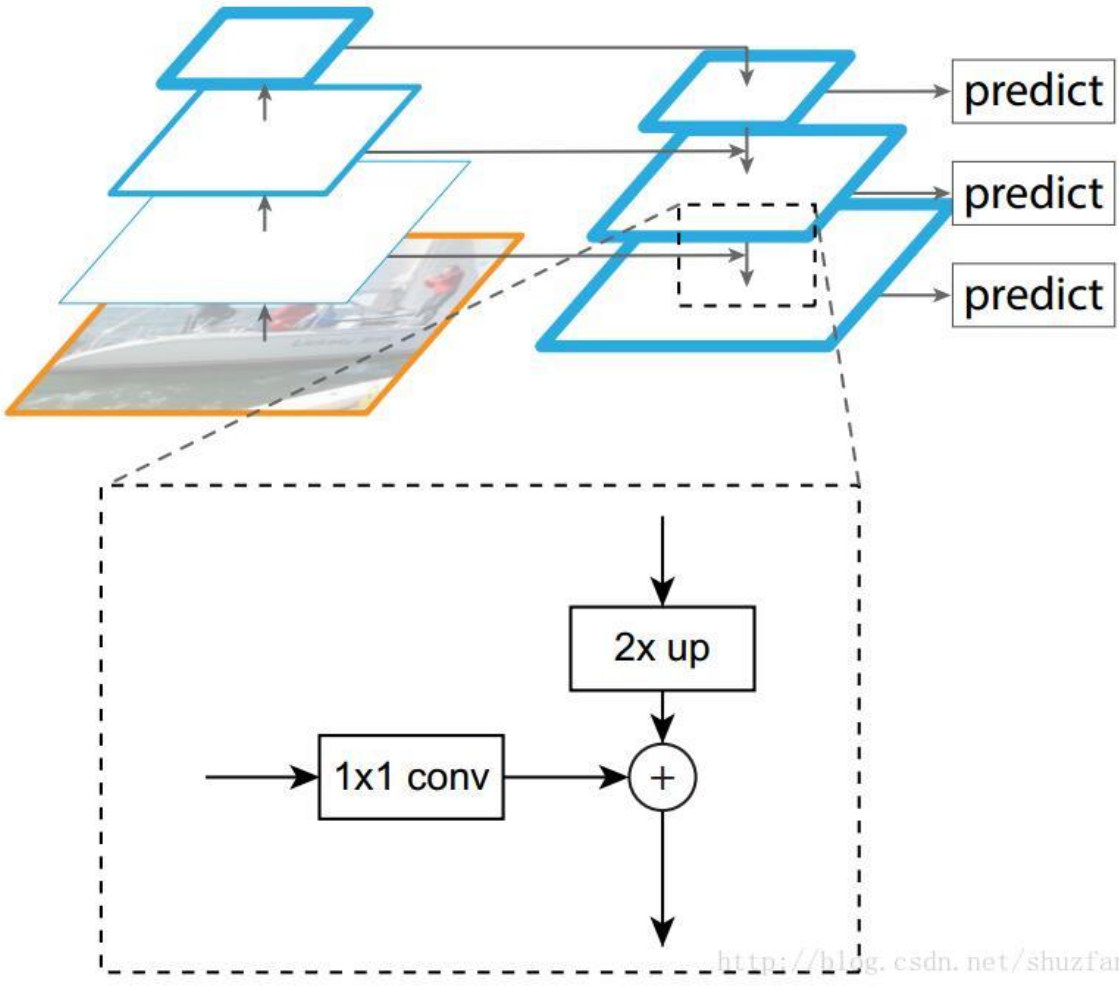
```
Performing object detection:
        + Batch 0, Inference Time: 0:00:01.767437
        + Batch 1, Inference Time: 0:00:00.063047
        + Batch 2, Inference Time: 0:00:00.057052
        + Batch 3, Inference Time: 0:00:00.062056
        + Batch 4, Inference Time: 0:00:00.060055
        + Batch 5, Inference Time: 0:00:00.062056
        + Batch 6, Inference Time: 0:00:00.065059
        + Batch 7, Inference Time: 0:00:00.059054
        + Batch 8, Inference Time: 0:00:00.064057
        + Batch 9, Inference Time: 0:00:00.061055
```

b

网络加入FPN

参考论文：Feature Pyramid Networks for Object Detection

引入FPN的目的是为了提高模型对于不同大小物体的分辨能力，如果只用最后一层feature做预测，由于感受野很大，模型不容易捕捉到小物体。在FPN前提出的方法还有`Featurized image pyramid`，`pyramidal feature hierarchy`等，`Featurized image pyramid`需要resize图片进行多次预测，耗时高，`pyramidal feature hierarchy`直接使用每一层的feature做预测，表现不佳。FPN的想法如下，通过上采样的方法将多层特征融合，再将这些特征用于预测。



实验

在将Faster RCNN的backbone改为带有FPN的resnet50后，结果如下

```
Average Precision  (AP) @[ IoU=0.50:0.95 | area=   all | maxDets=100 ] = 0.557
Average Precision  (AP) @[ IoU=0.50      | area=   all | maxDets=100 ] = 0.894
Average Precision  (AP) @[ IoU=0.75      | area=   all | maxDets=100 ] = 0.602
Average Precision  (AP) @[ IoU=0.50:0.95 | area= small | maxDets=100 ] = 0.383
Average Precision  (AP) @[ IoU=0.50:0.95 | area=medium | maxDets=100 ] = 0.558
Average Precision  (AP) @[ IoU=0.50:0.95 | area= large | maxDets=100 ] = 0.692
Average Recall     (AR) @[ IoU=0.50:0.95 | area=   all | maxDets=  1 ] = 0.157
Average Recall     (AR) @[ IoU=0.50:0.95 | area=   all | maxDets= 10 ] = 0.622
Average Recall     (AR) @[ IoU=0.50:0.95 | area=   all | maxDets=100 ] = 0.633
Average Recall     (AR) @[ IoU=0.50:0.95 | area= small | maxDets=100 ] = 0.383
Average Recall     (AR) @[ IoU=0.50:0.95 | area=medium | maxDets=100 ] = 0.641
Average Recall     (AR) @[ IoU=0.50:0.95 | area= large | maxDets=100 ] = 0.730
```

对比之前的结果，对于小物体的召回率和精度都有很大的提升

```
Average Precision  (AP) @[ IoU=0.50:0.95 | area=   all | maxDets=100 ] = 0.404
Average Precision  (AP) @[ IoU=0.50      | area=   all | maxDets=100 ] = 0.728
Average Precision  (AP) @[ IoU=0.75      | area=   all | maxDets=100 ] = 0.389
Average Precision  (AP) @[ IoU=0.50:0.95 | area= small | maxDets=100 ] = 0.067
Average Precision  (AP) @[ IoU=0.50:0.95 | area=medium | maxDets=100 ] = 0.417
Average Precision  (AP) @[ IoU=0.50:0.95 | area= large | maxDets=100 ] = 0.575
Average Recall     (AR) @[ IoU=0.50:0.95 | area=   all | maxDets=  1 ] = 0.132
Average Recall     (AR) @[ IoU=0.50:0.95 | area=   all | maxDets= 10 ] = 0.504
Average Recall     (AR) @[ IoU=0.50:0.95 | area=   all | maxDets=100 ] = 0.535
Average Recall     (AR) @[ IoU=0.50:0.95 | area= small | maxDets=100 ] = 0.067
Average Recall     (AR) @[ IoU=0.50:0.95 | area=medium | maxDets=100 ] = 0.564
Average Recall     (AR) @[ IoU=0.50:0.95 | area= large | maxDets=100 ] = 0.675
```

改进nms

参考文献：Soft-NMS -- Improving Object Detection With One Line of Code

代码可见https://www.kaggle.com/clicker3/nms-method

相比传统nms直接对重叠区域高于阈值的框进行剔除，soft-nms通过iou对得分进行惩罚（线性惩罚或高斯惩罚），再通过得分阈值选取框，数学表述如下

线性惩罚

$$s_i = \begin{cases} s_i, & \text{iou } (\mathcal{M}, b_i) < N_t \\ s_i \left(1 - \text{iou}(\mathcal{M}, b_i)\right), & \text{iou } (\mathcal{M}, b_i) \geq N_t \end{cases}$$

高斯惩罚

$$s_i = s_i e^{-\frac{\text{iou } (M, b_i)^2}{\sigma}}, \forall b_i \in \mathcal{D}$$

结果如下

```
eval(fasterRCNN, test_loader, nms_threshold=0.5, iou_threshold=0.4, conf_threshold=0.001, nms_m
ethod="tradition")
eval(fasterRCNN, test_loader, nms_threshold=0.5, iou_threshold=0.4, conf_threshold=0.001, nms_m
ethod="soft")
```

```
Computing AP: 100%|██████████| 2/2 [00:00<00:00, 539.98it/s]


+-------+------------+-----------+---------+---------+
| Index | Class name | Precision | Recall  | AP      |
+-------+------------+-----------+---------+---------+
| 1     | crater     | 0.14706   | 0.83333 | 0.63461 |
| 2     | cone       | 0.17603   | 0.42727 | 0.35373 |
+-------+------------+-----------+---------+---------+
---- mAP 0.4941713066650625

Computing AP: 100%|██████████| 2/2 [00:00<00:00, 389.37it/s]


+-------+------------+-----------+---------+---------+
| Index | Class name | Precision | Recall  | AP      |
+-------+------------+-----------+---------+---------+
| 1     | crater     | 0.10965   | 0.83333 | 0.65488 |
| 2     | cone       | 0.15878   | 0.42727 | 0.35571 |
+-------+------------+-----------+---------+---------+
---- mAP 0.5052967103486282
```