

A nighttime photograph of a university campus. In the background, a large, multi-story building with many lit windows stands behind a row of bare trees. The scene is reflected in a body of water in the foreground. The text 'TONGJI UNIVERSITY' is overlaid in a large, white, serif font.

TONGJI UNIVERSITY

汇报小组：计科二组

CONTECT

PART ONE

最终成果展示

PART TWO

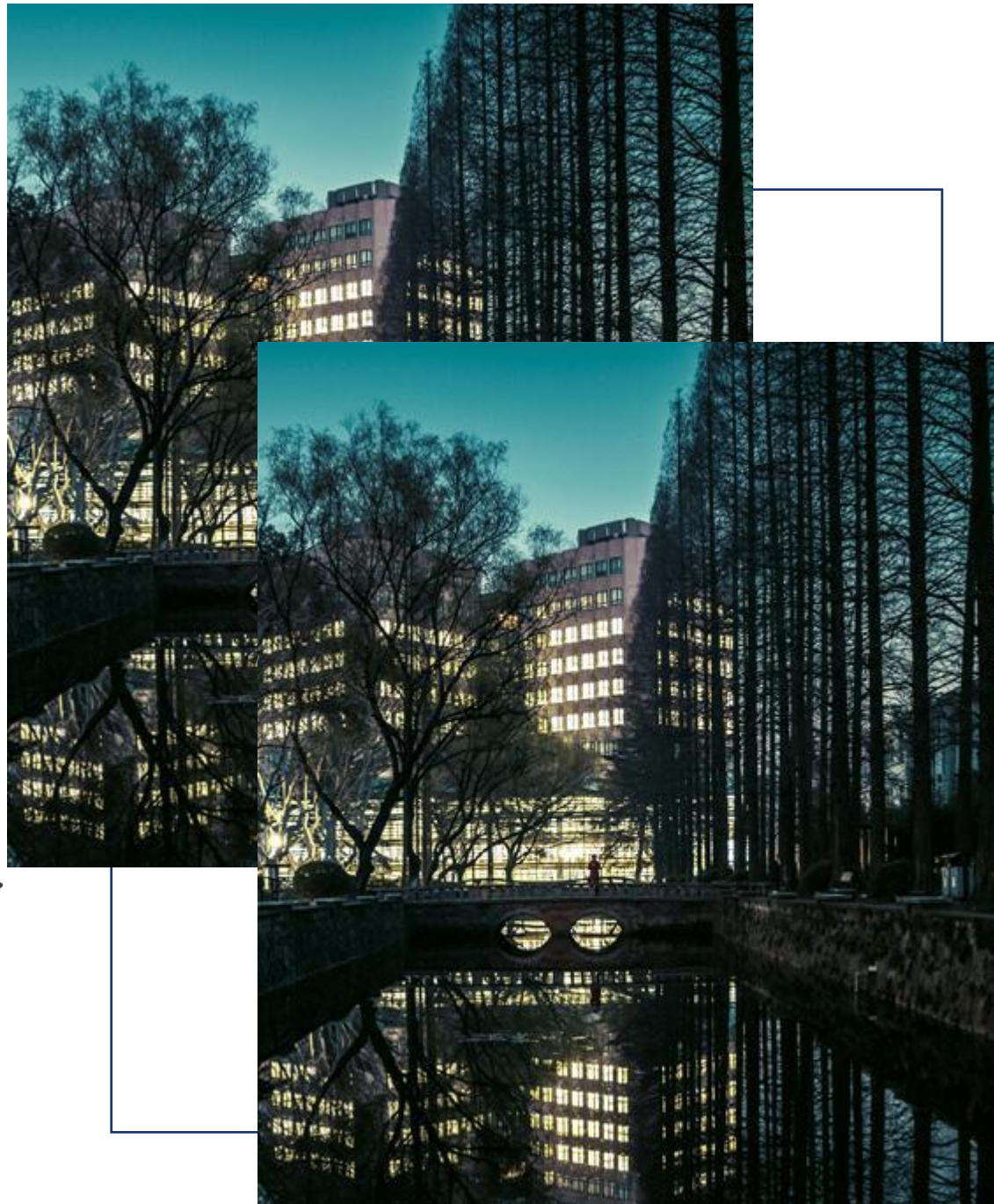
数据集的收集与制作

PART Three

模型的调参历程

PART ONE

模型的纵向比较



PART ONE

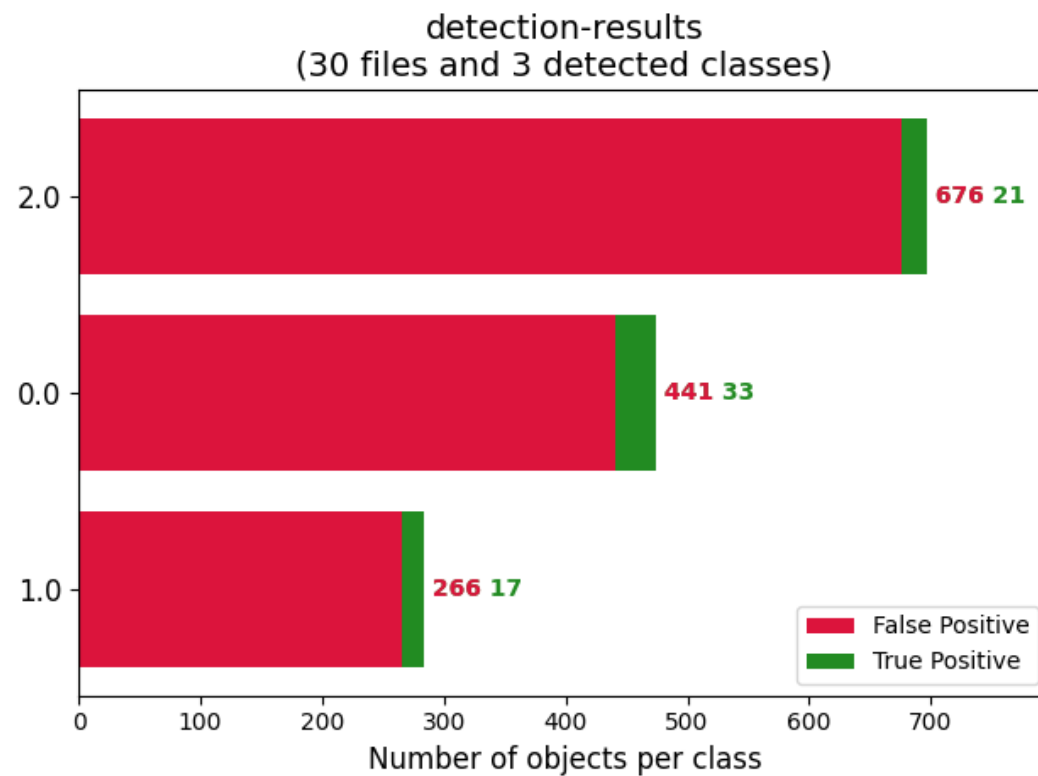
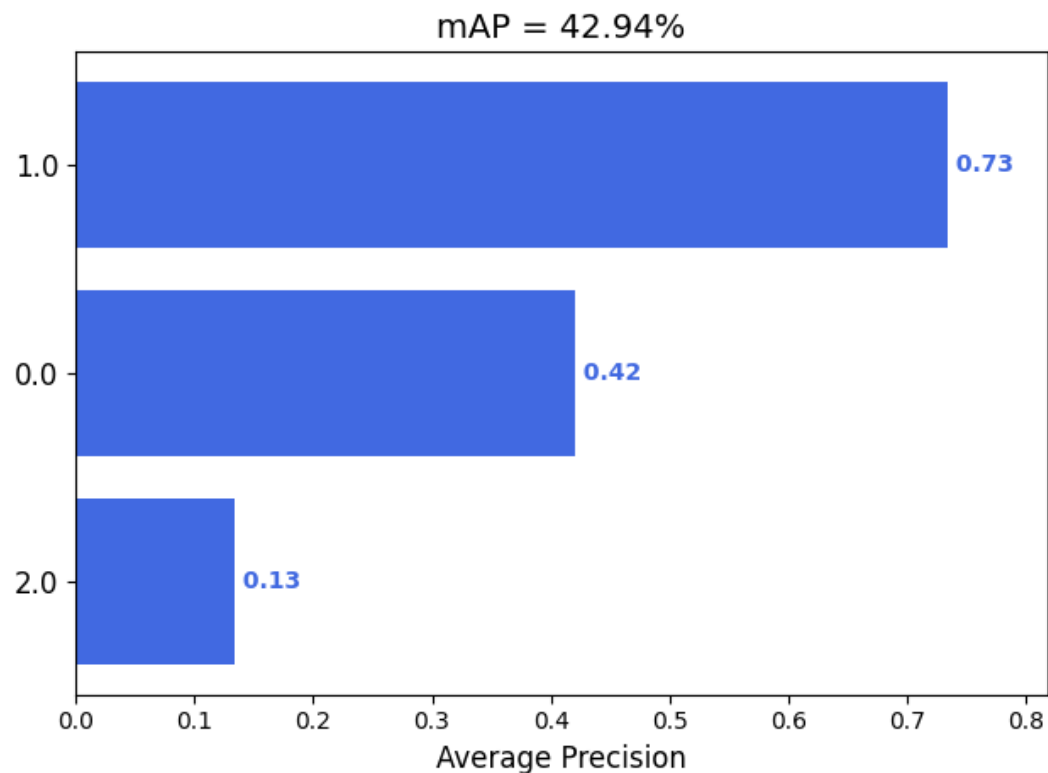
PART ONE

最终成果展示

PART ONE

42.94%

- 第一次迭代

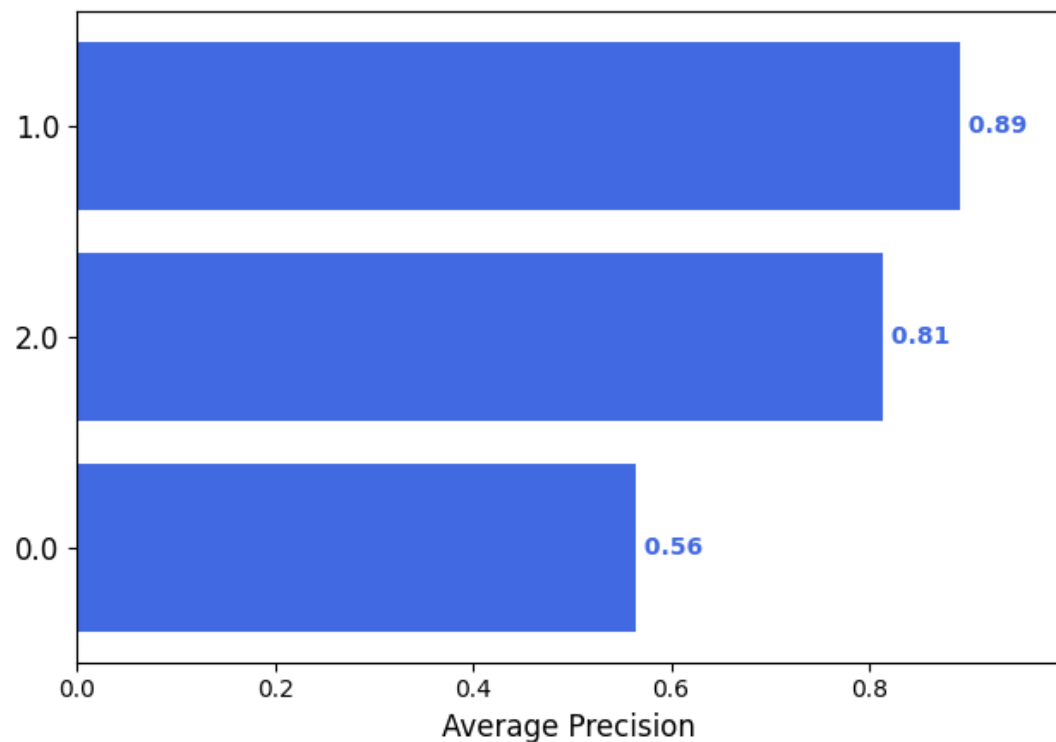


PART ONE

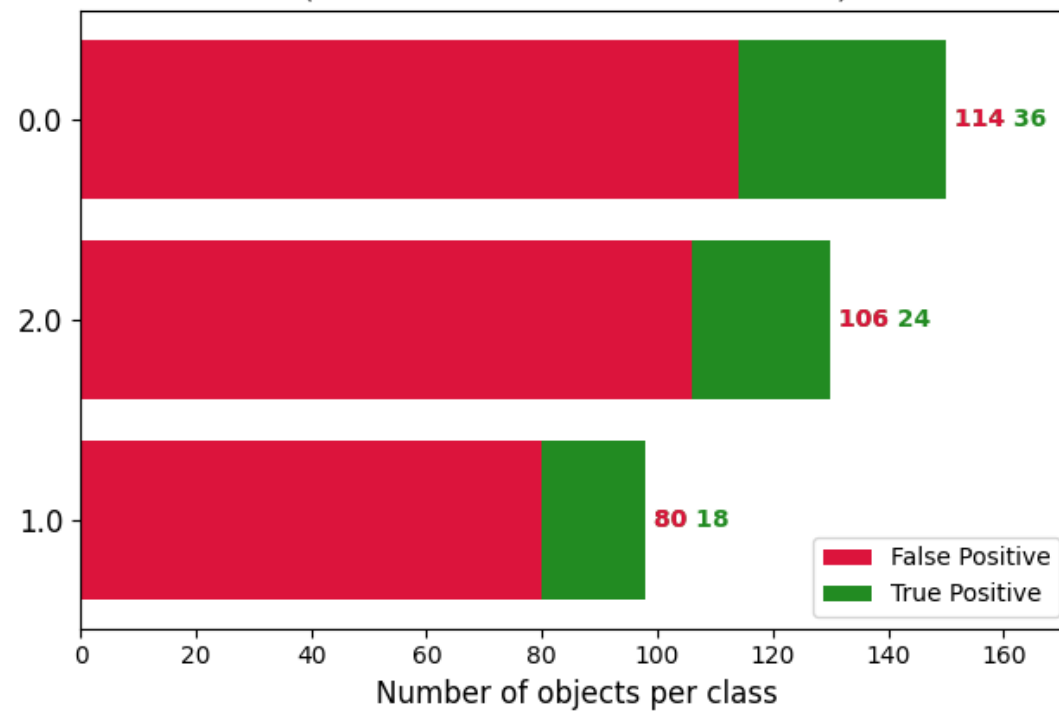
75.67%

- 第二次迭代

mAP = 75.67%



detection-results
(30 files and 3 detected classes)

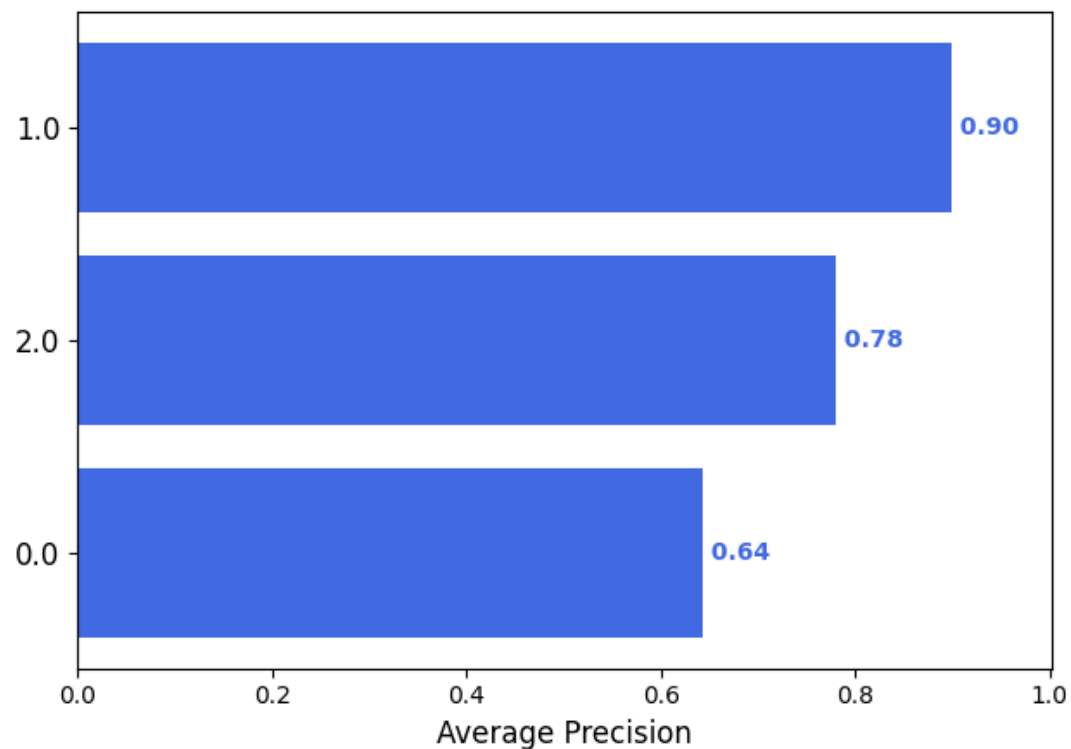


PART ONE

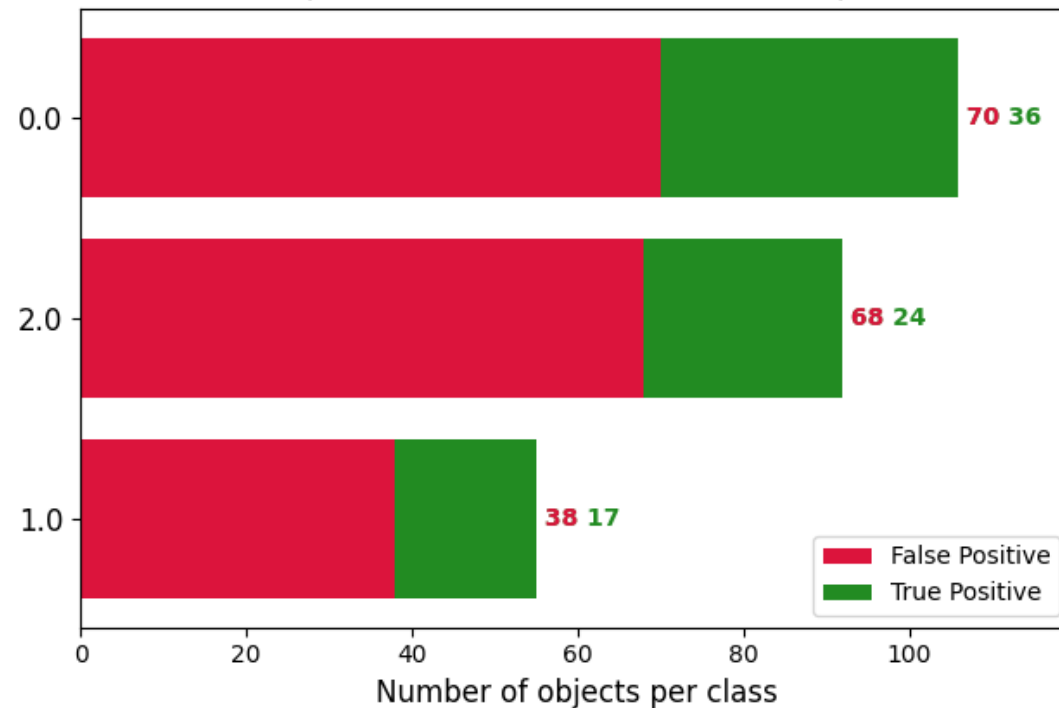
77.37%

- 第三次迭代

mAP = 77.37%

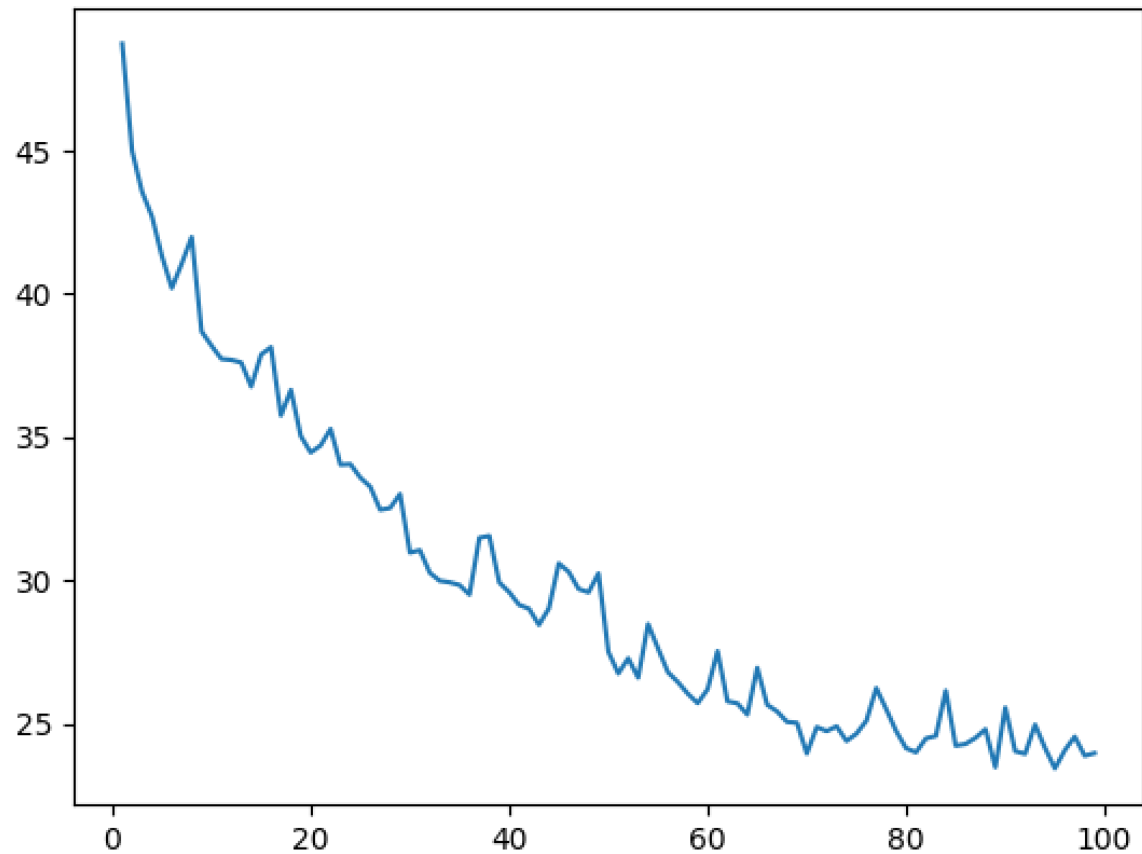
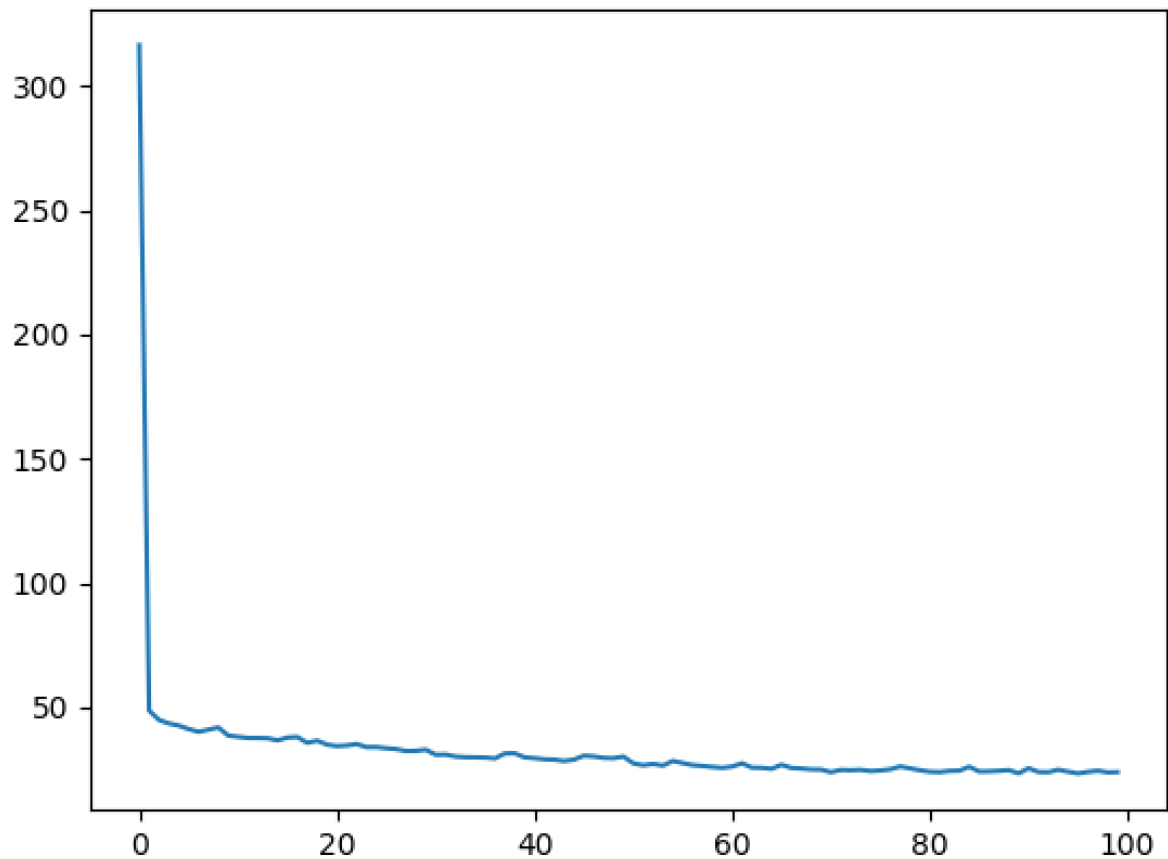


detection-results
(30 files and 3 detected classes)



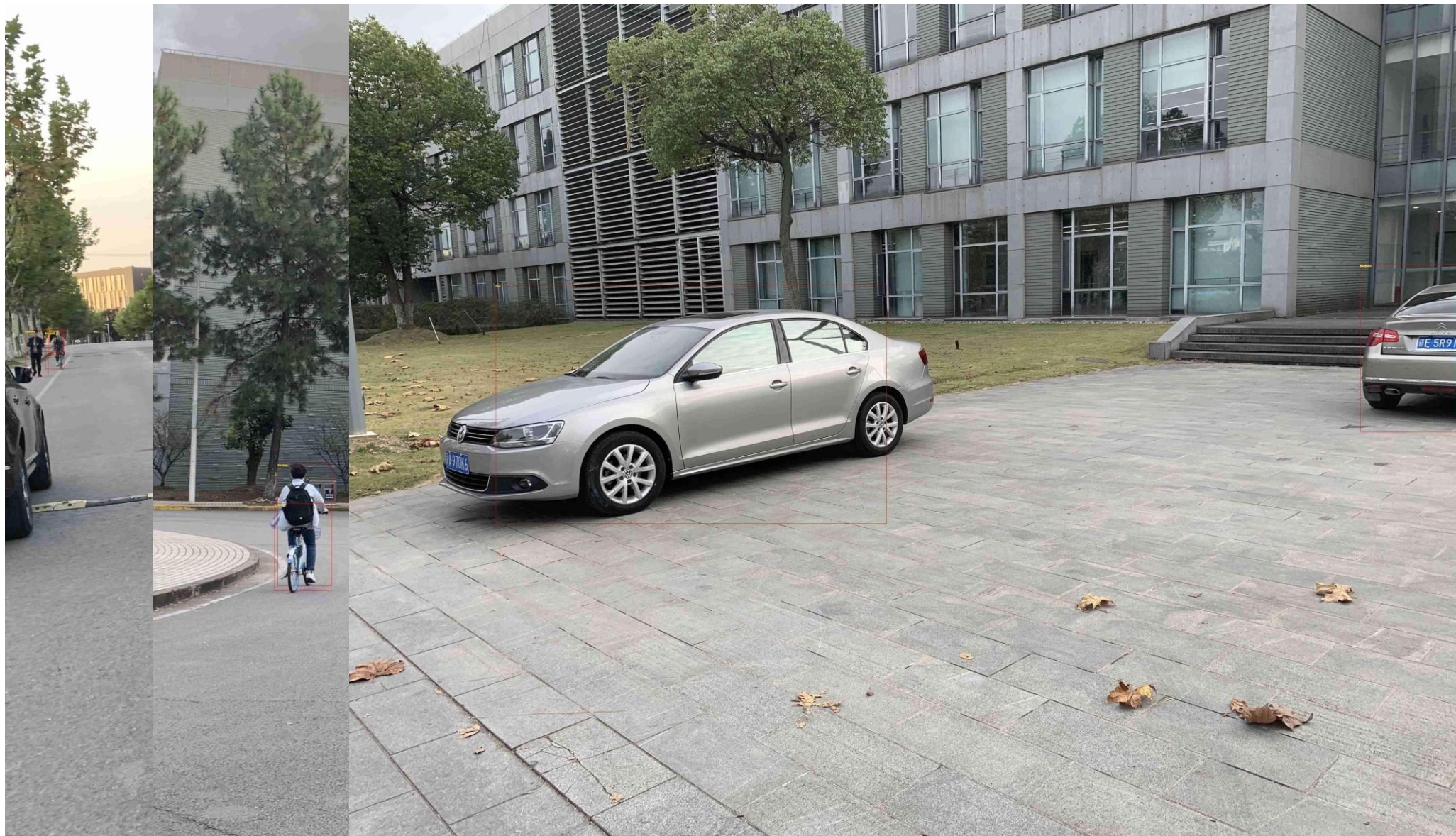
PART ONE

- loss函数的下降曲线，其中由于第一个点的loss值过大，在第二个图中删除了第一个点并重新绘制了下降曲线



PART ONE

识别效果展示



PART TWO

PART TWO

数据集的收集与制作

PART TWO



1. 在校园内拍照

2. 使用python对照片进行预处理

- 对照片进行处理，长宽同时调整为1024pix，方便后续的程序处理
- 预留处理空间
- 加载更快

3.使用YOLOV5

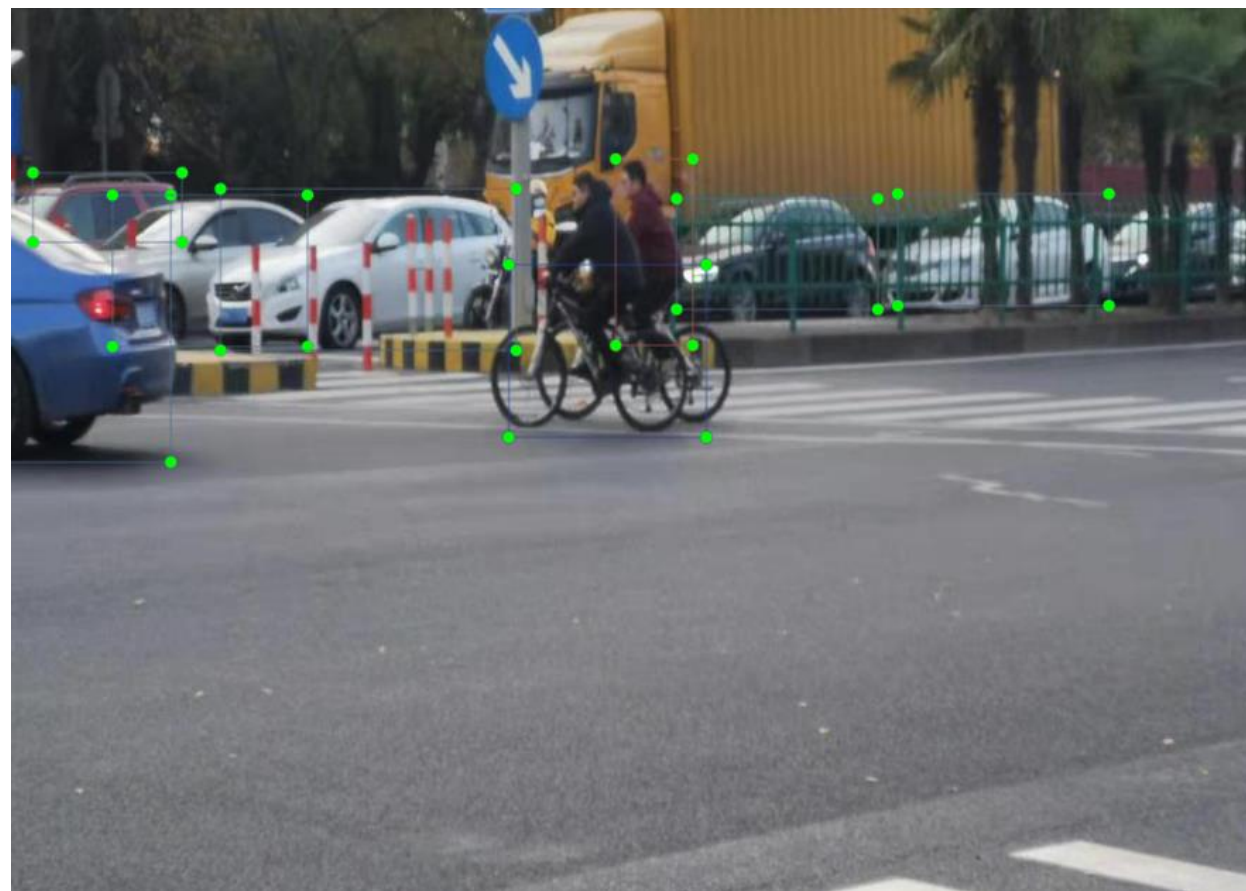
- 最初发现手工标注的效率与精度较低
- 在网上学习后使用YOLOV5自动标注
- 最后进行手工验证修改

4. 标注格式的调整

- 对自动标注的图片的格式进行修改，使得程序可以使用

PART TWO

YOLOv5标注情况



PART THREE

PART THREE

模型的调参历程

PART THREE

0%

第一次尝试中，我们发现默认的batch_size较大，直接运行YOLOV3模型会导致显存不足，于是使用了YOLOV3_TINY模型，但是该模型精度较低，导致结果不令人满意，没有标注出有意义的目标。

解决方案：将batch_size改到16，选用YOLOV3模型。

PART THREE

0%

第二次尝试中，将参数 `nms_thresh`(非最大抑制)改的过大，造成要求过于严格，使得没有物体符合要求，图上没有任何标注。

解决方案：逐渐提高 `nms_thresh` 的值，在迭代中逐渐增强模型的识别能力。

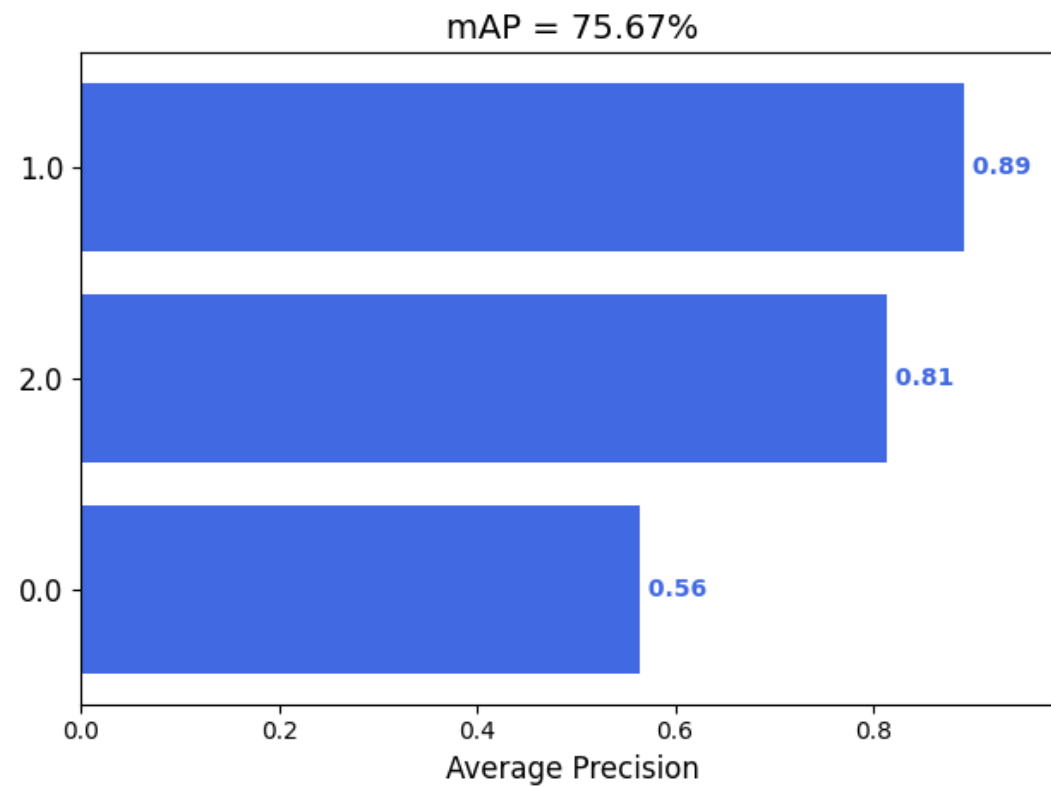
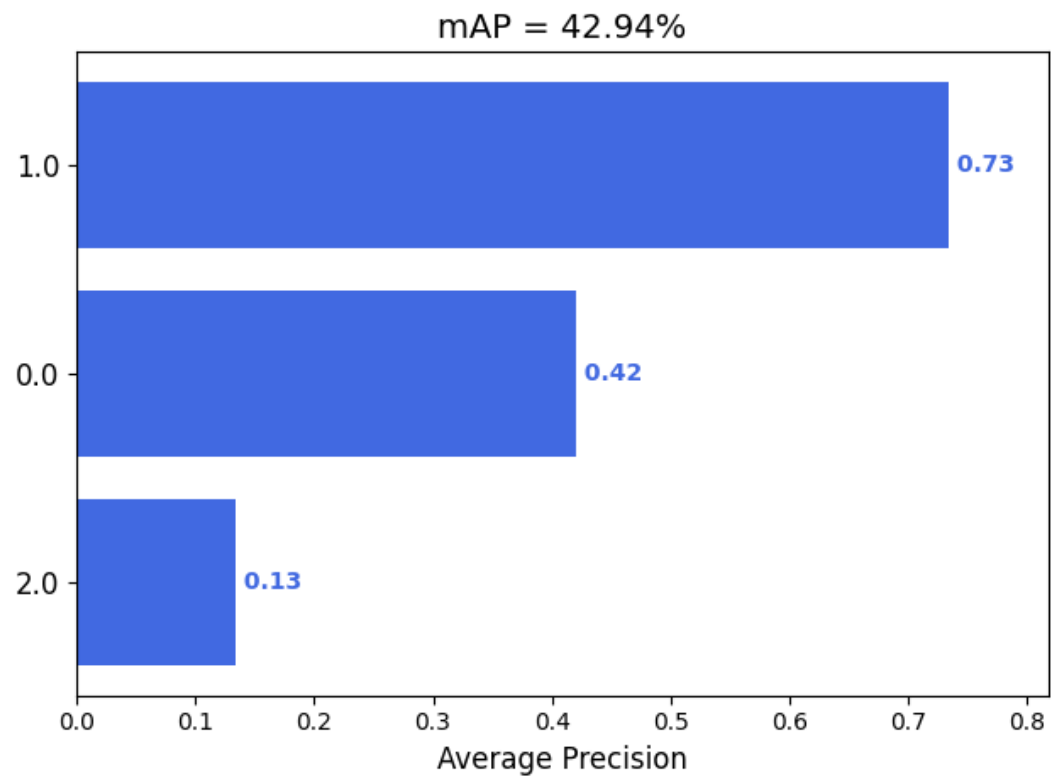
PART THREE

49%

第三次尝试中，我们整体的识别效果较好，但是自行车的识别效果很低，原因在于数据集中自行车大多为成片出现，没有清楚地标注出所有自行车，导致学习效果较差，自行车的识别准确度仅有49%。

解决方案：再次拍摄了部分自行车照片，配合YOLOV5的标注，优化了数据集。

PART THREE



PART THREE

当我们已经训练到了75%左右的时候，发现梯度不断波动，无法下降，我们将学习率改小，然后结果优化地更好



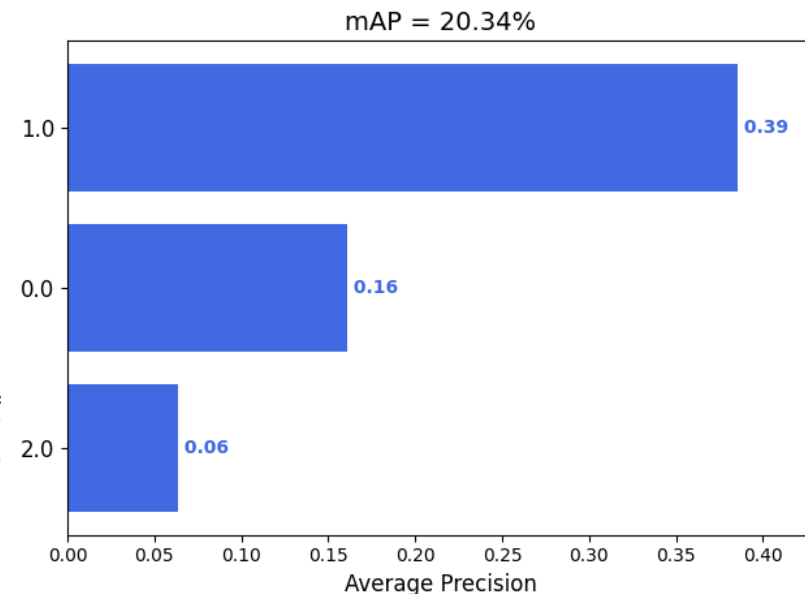
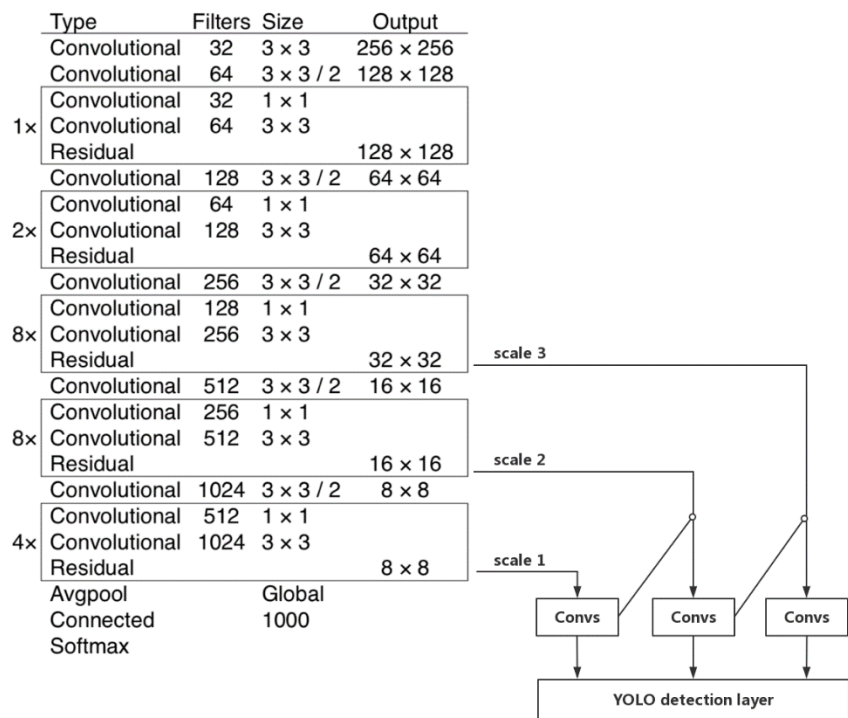
PART FOUR

PART FOUR

模型的纵向比较

PART FOUR

```
def basicblock(self, input, num_filters):
    conv1 = self.conv_bn(input, num_filters, filter_size=1, stride=1, padding=0)
    conv2 = self.conv_bn(conv1, num_filters, filter_size=3, stride=1, padding=1)
    conv3 = self.conv_bn(conv2, num_filters * 2, filter_size=3, stride=1, padding=1)
    out = fluid.layers.elementwise_add(x=input, y=conv3, act=None)
    return out
```



A night scene of a city street. In the background, a large, multi-story building with many lit windows stands behind a row of bare trees. The scene is reflected in a body of water in the foreground, which is bordered by a low stone wall. The word "THANK" is written in large, white, handwritten-style capital letters across the center of the image.

THANK