基于场景草图的
目标定位与检测技术研究

汇报人　刘卓云

2021.03.12

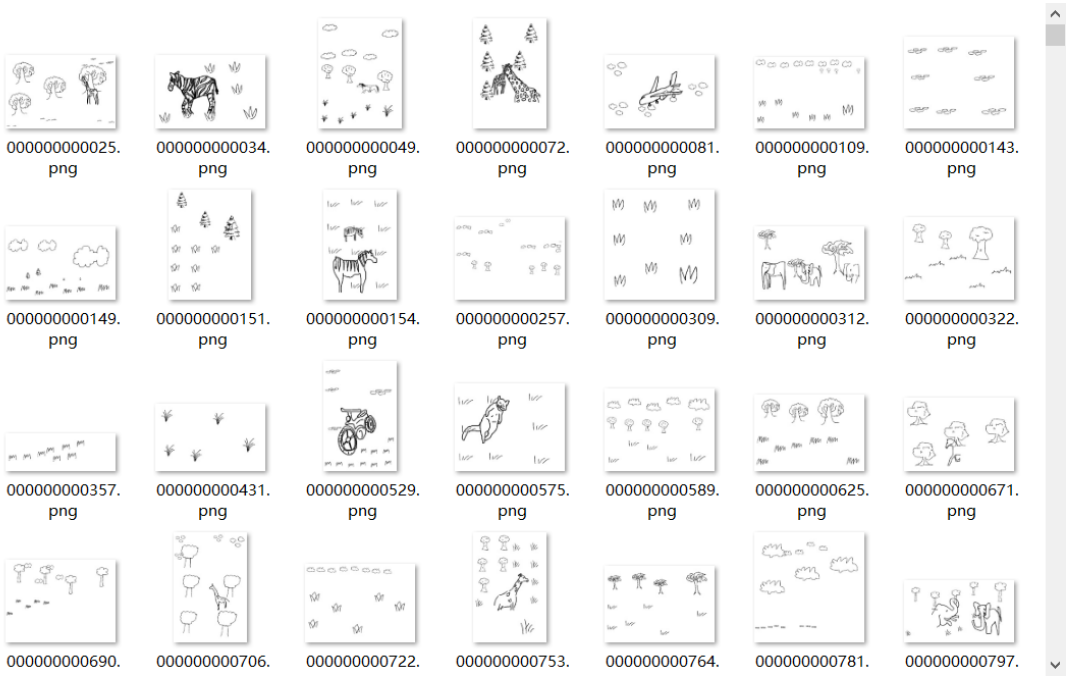
目录

CONTENTS

数据集准备

数据集准备

SketchyCoCo



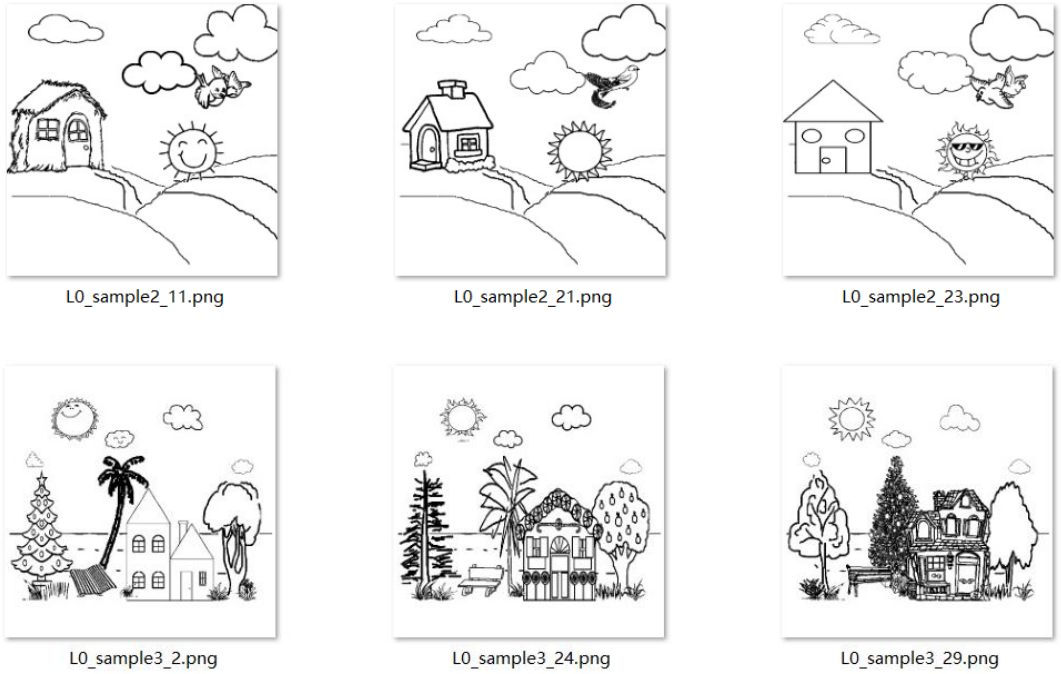
场景级 约14000

对象级 14个类别

数据集准备

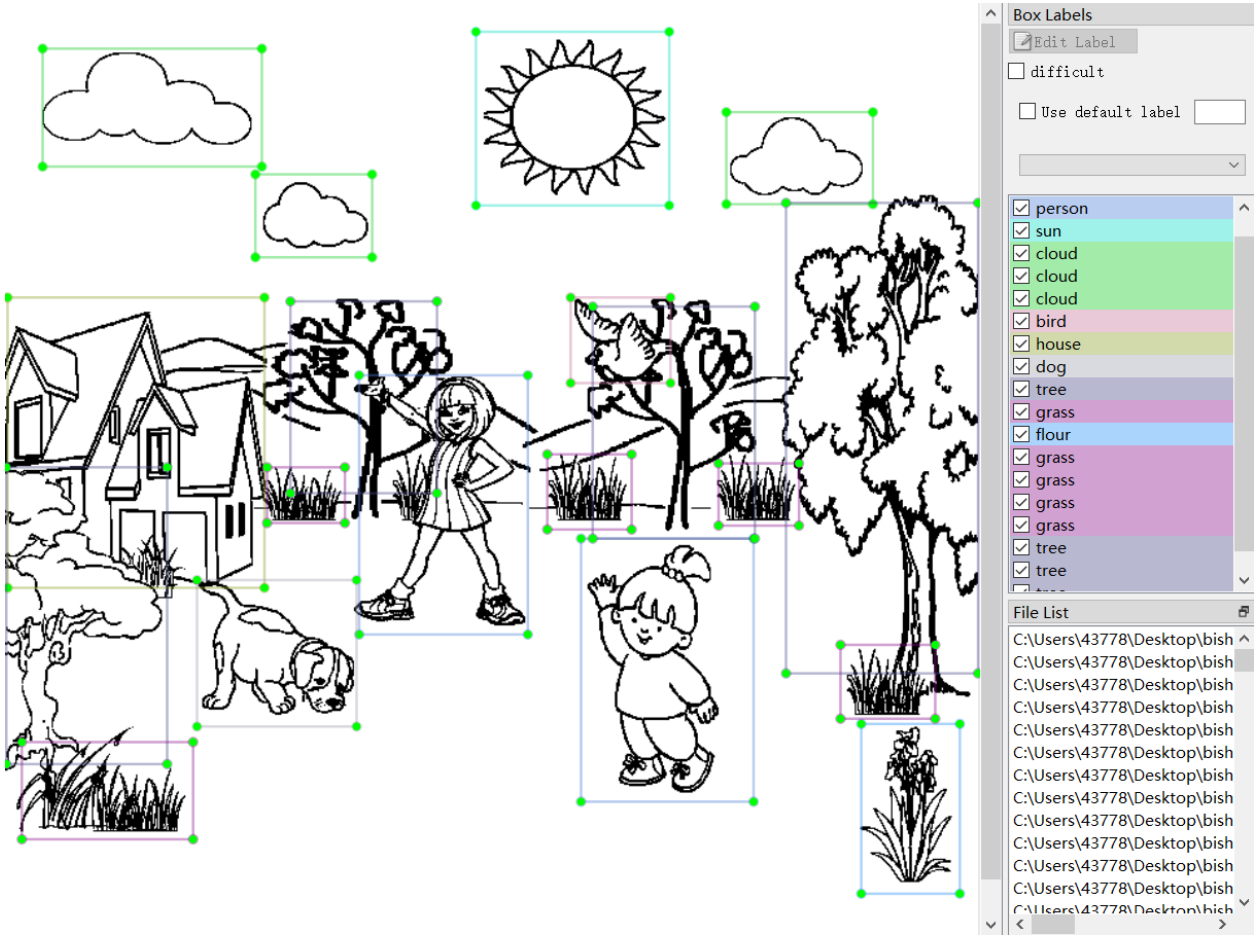
SketchyScene

不同场景 约5600



相同场景 3张

数据集准备



Labelimg标注

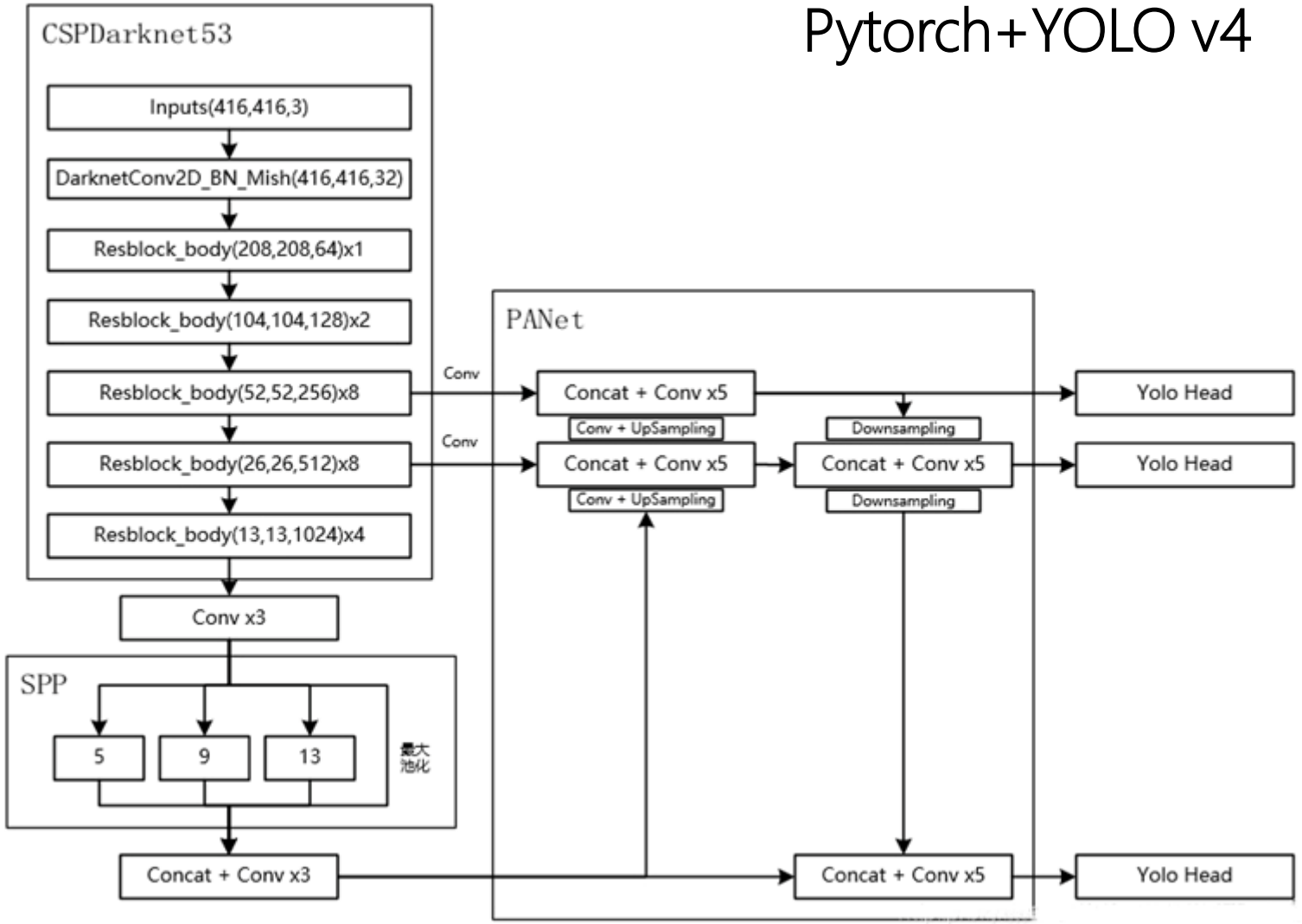
名称	修改日期	类型	大小
L0_sample6.xml	2021/2/3 11:25	XML 文档	3 KB
L0_sample7.xml	2021/2/3 11:27	XML 文档	2 KB
L0_sample8.xml	2021/2/3 11:30	XML 文档	3 KB
L0_sample9.xml	2021/2/3 11:33	XML 文档	3 KB
L0_sample10.xml	2021/2/3 11:33	XML 文档	1 KB
L0_sample11.xml	2021/2/3 11:35	XML 文档	2 KB
L0_sample12.xml	2021/2/3 11:36	XML 文档	2 KB
L0_sample13.xml	2021/2/3 11:37	XML 文档	2 KB
L0_sample14.xml	2021/2/3 20:55	XML 文档	3 KB
L0_sample15.xml	2021/2/3 11:42	XML 文档	5 KB
L0_sample16.xml	2021/2/3 11:43	XML 文档	3 KB
L0_sample17.xml	2021/2/3 11:46	XML 文档	4 KB
L0_sample18.xml	2021/2/3 11:50	XML 文档	4 KB
L0_sample19.xml	2021/2/3 11:56	XML 文档	3 KB
L0_sample20.xml	2021/2/3 15:44	XML 文档	3 KB
L0_sample21.xml	2021/2/3 15:46	XML 文档	3 KB
L0_sample22.xml	2021/2/3 15:50	XML 文档	4 KB
L0_sample23.xml	2021/2/3 15:52	XML 文档	4 KB
L0_sample24.xml	2021/2/3 15:55	XML 文档	4 KB
L0_sample25.xml	2021/2/3 15:58	XML 文档	4 KB
L0_sample26.xml	2021/2/3 16:05	XML 文档	3 KB
L0_sample27.xml	2021/2/3 16:08	XML 文档	3 KB
L0_sample28.xml	2021/2/3 16:11	XML 文档	3 KB

102个标注

YOLO算法了解

YOLO算法了解

Pytorch+YOLO v4

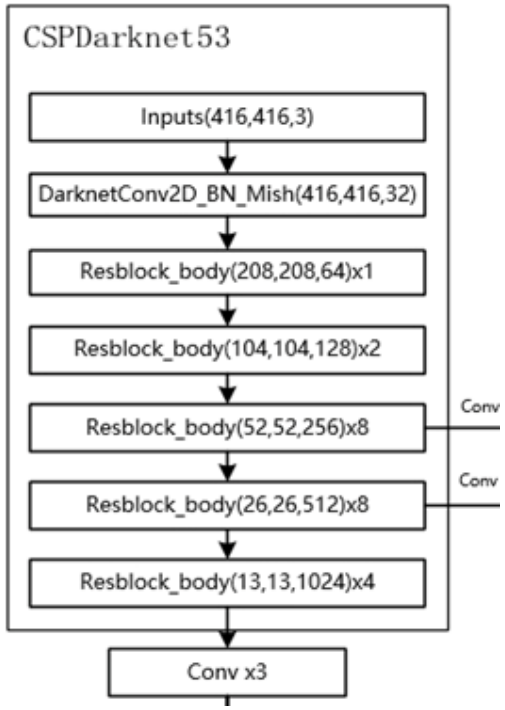


YOLO算法了解

```
def forward(self, x):
    return x * torch.tanh(F.softplus(x))
```

$$Mish=x\times\tanh(\ln(1+e^{x}))$$

主干特征提取网络




```
self.conv = nn.Conv2d(in_channels, out_channels, kernel_size, stride, kernel_size//2, bias=False)
self.bn = nn.BatchNorm2d(out_channels)
self.activation = Mish()
```

BasicConv

```
        self.block = nn.Sequential(
            BasicConv(channels, hidden_channels, 1),
            BasicConv(hidden_channels, channels, 3)
        )

    def forward(self, x):
        return x + self.block(x)
```

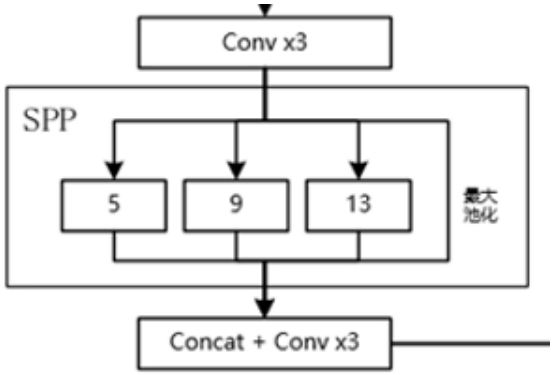
Resblock

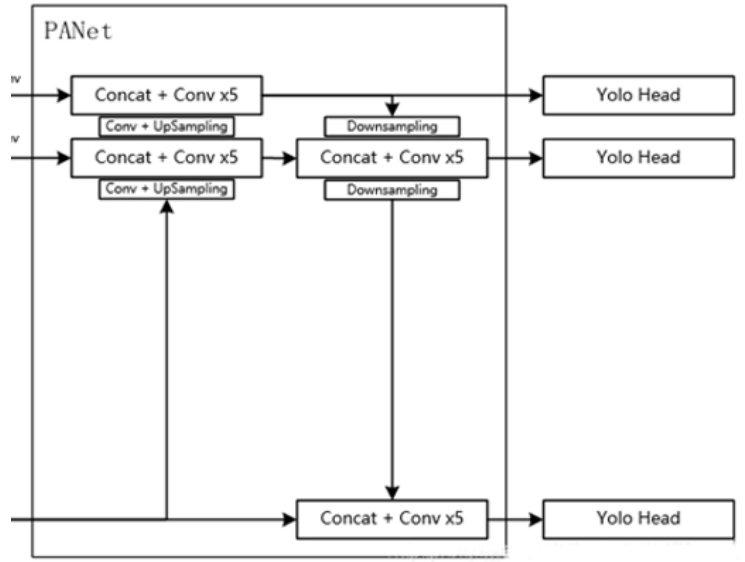
```
self.split_conv0 = BasicConv(out_channels, out_channels, 1)
self.split_conv1 = BasicConv(out_channels, out_channels, 1)
self.blocks_conv = nn.Sequential(
    Resblock(channels=out_channels, hidden_channels=out_channels//2),
    BasicConv(out_channels, out_channels, 1)
)

self.concat_conv = BasicConv(out_channels*2, out_channels, 1)
```

```
def forward(self, x):
    x = self.downsample_conv(x)

    x0 = self.split_conv0(x)

    x1 = self.split_conv1(x)
    x1 = self.blocks_conv(x1)

    x = torch.cat([x1, x0], dim=1)

    x = self.concat_conv(x)

    return x
```

Resblock_body

YOLO算法了解





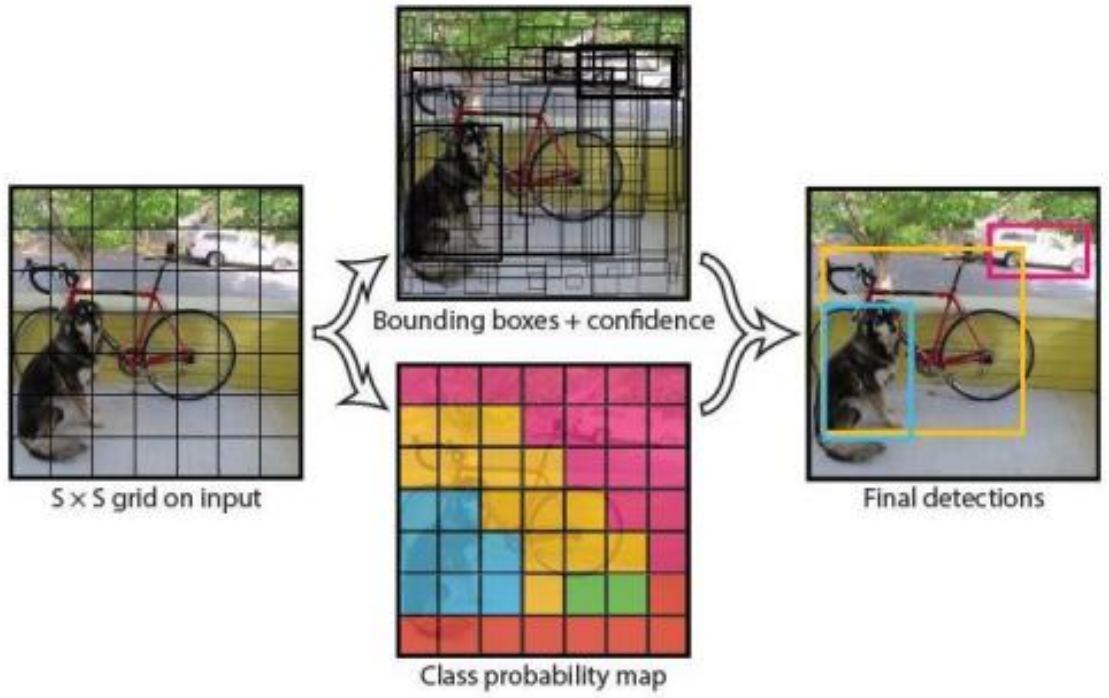

```python
class SpatialPyramidPooling(nn.Module):
    def __init__(self, pool_sizes=[5, 9, 13]):
        super(SpatialPyramidPooling, self).__init__()

        self.maxpools = nn.ModuleList([nn.MaxPool2d(pool_size, 1, pool_size//2) for pool_size in pool_sizes])

    def forward(self, x):
        features = [maxpool(x) for maxpool in self.maxpools[::-1]]
        features = torch.cat(features + [x], dim=1)

        return features
```

不同大小的池化核进行池化并堆叠

```python
class Upsample(nn.Module):
    def __init__(self, in_channels, out_channels):
        super(Upsample, self).__init__()

        self.upsample = nn.Sequential(
            conv2d(in_channels, out_channels, 1),
            nn.Upsample(scale_factor=2, mode='nearest')
        )

    def forward(self, x,):
        x = self.upsample(x)
        return x
```

torch.nn.Upsample

YOLO算法了解



S×S ×(5 × 3+20)

Yolo Head

网络将输入图片分割成 S × S 的网格，每个单元格检测中心点落在该格子内的目标。每个单元格预测 3 个边界框 以及边界框的置信度。每个边界框对应着 5 个预测参数， 即边界框的中心点坐标(x,y)，宽和高(w,h)还有置信度。其中，置信度包括了边界框含有目标的可能性以及边界框的准确度两个因子。

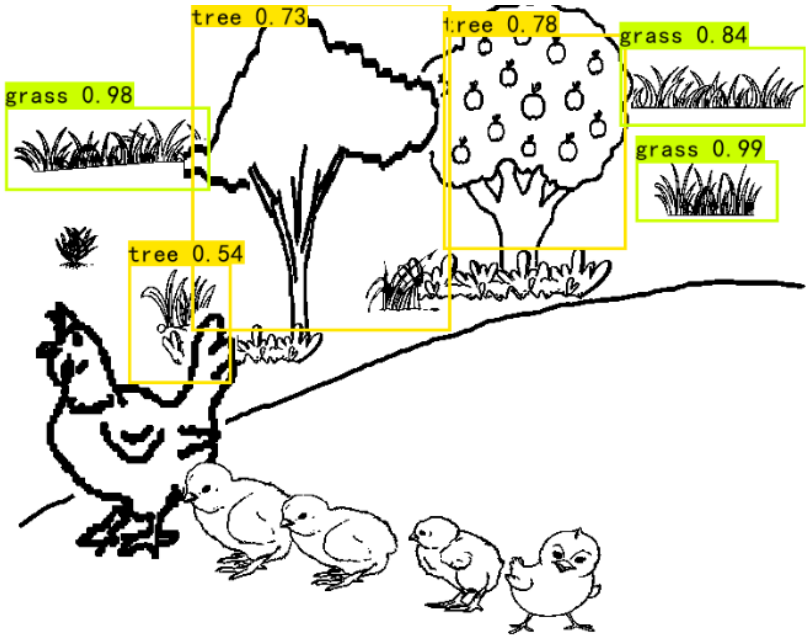
YOLO算法了解

由于可能出现多个物体重叠，使得不同目标的中心点落在同一单元格内。此时需要使用先验框帮助一个单元格检测到多个对象。

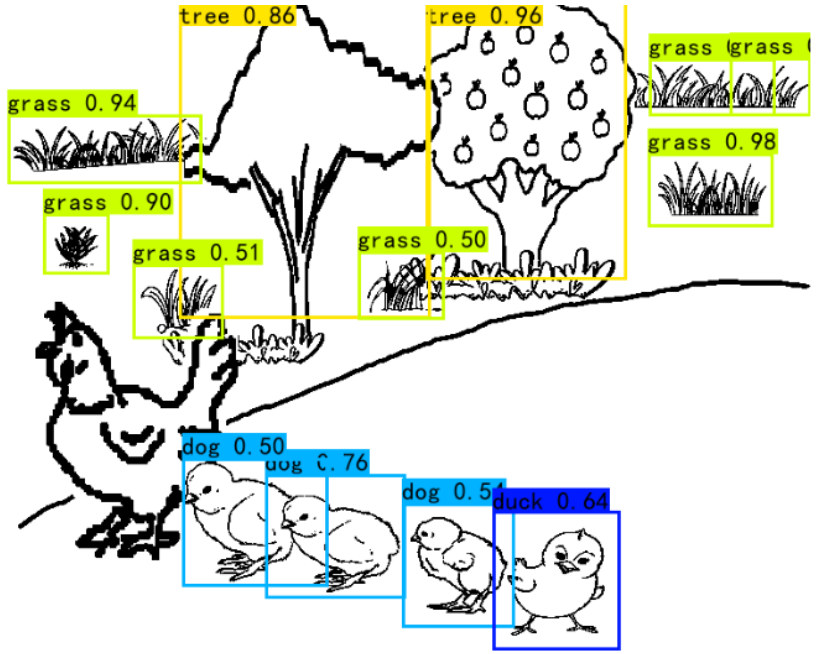
例如，我们要检测很宽的汽车和站着的行人，则每个单元格最多检测到两个目标（B=2）。我们将定义一个形状大致是汽车形状的锚点框，这个方框的宽比高大，然后定义另一个能在里面填充站立行人的锚点框，高比宽大。

遇到的问题

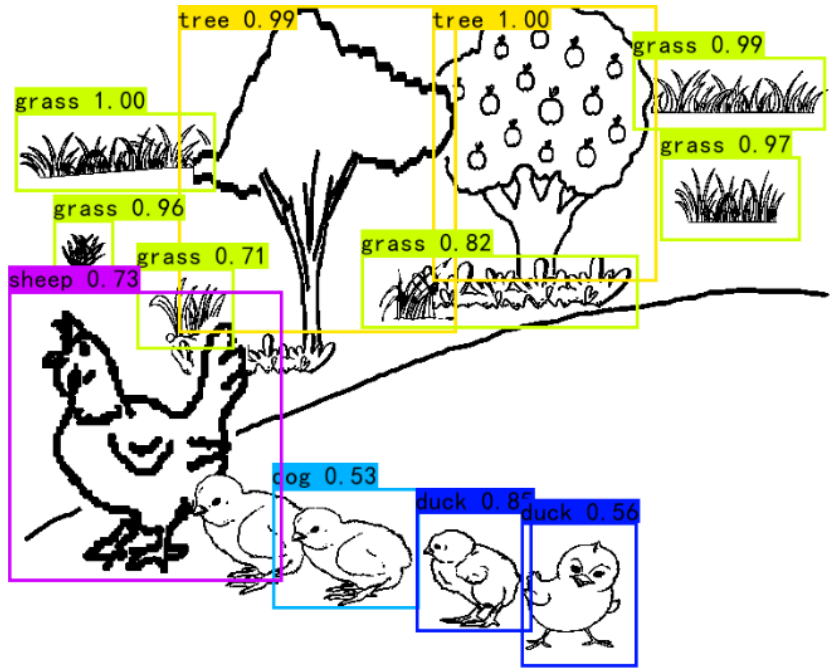
遇到的问题



Epoch 5

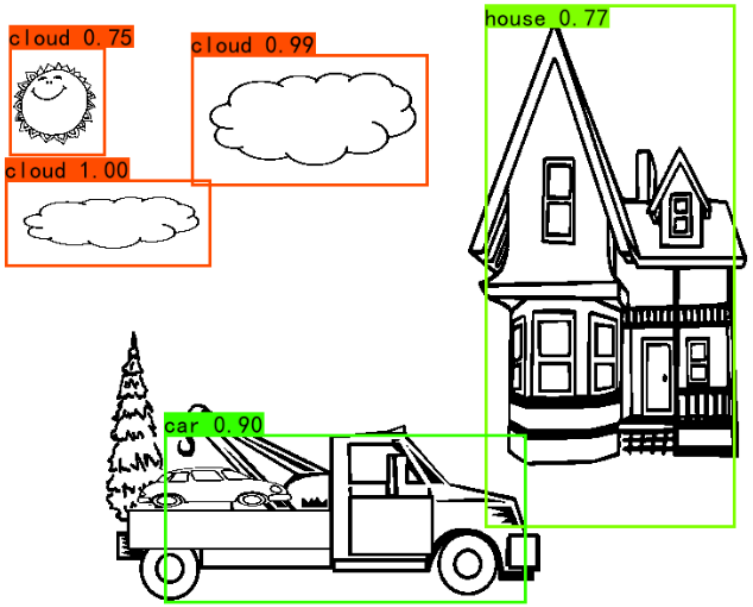


Epoch 20

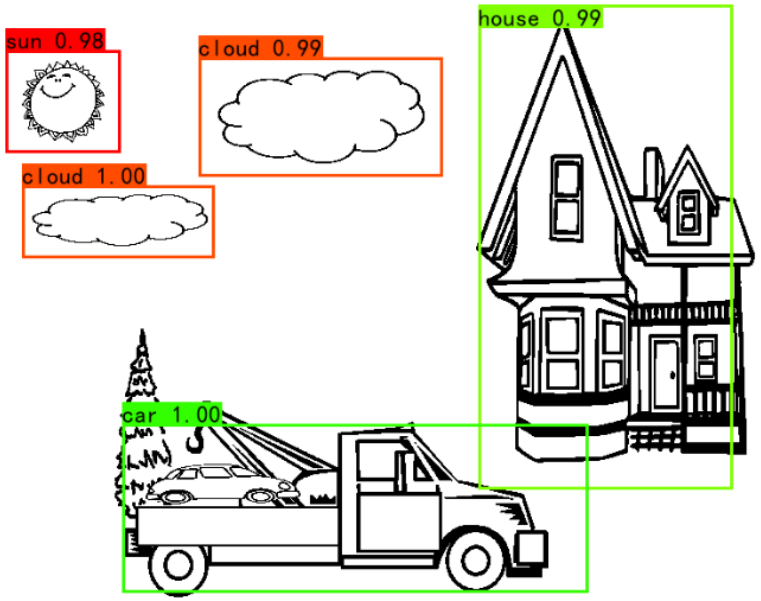


Epoch 100

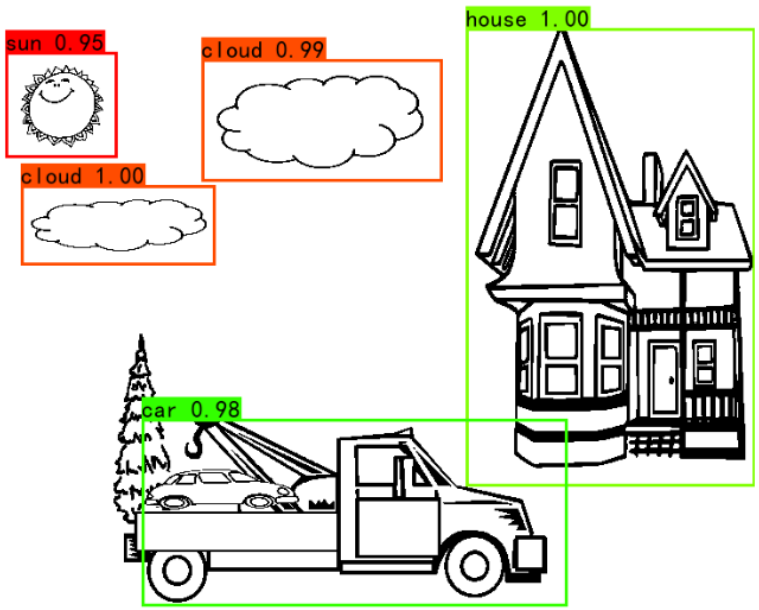
遇到的问题



Epoch 5



Epoch 20

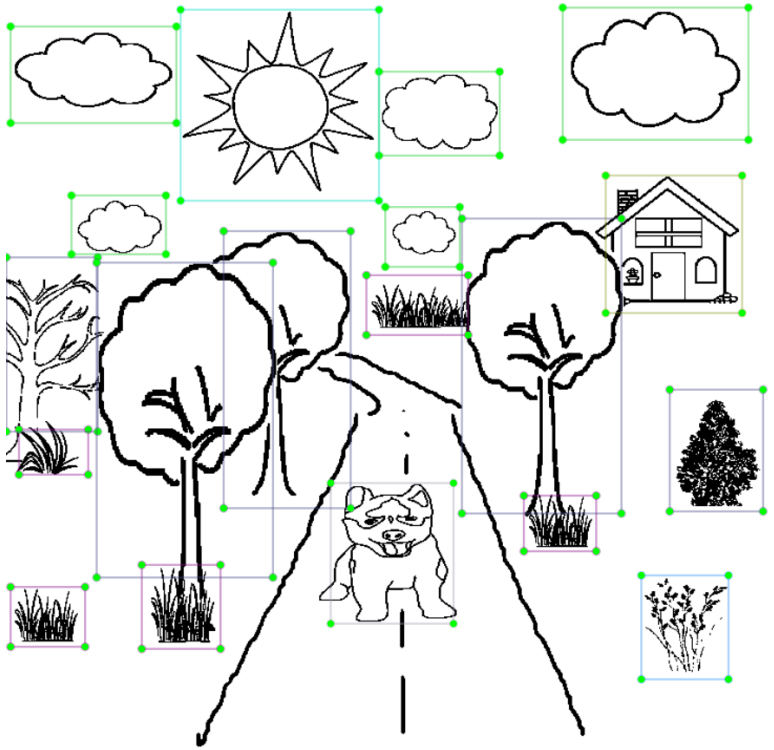


Epoch 100

遇到的问题

数据集规模较小

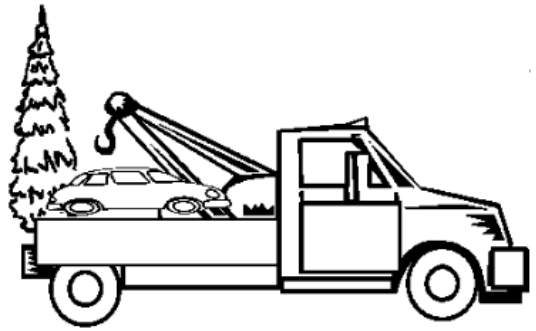
标注过程中发现如sun、cloud、bird等一般单独出现没有重叠，grass、tree、house等虽然常常与其他对象重叠但出现的数量很多，这些种类的识别结果准确度较高。而pig、duck、chicken等训练样本太少且形态变化大，

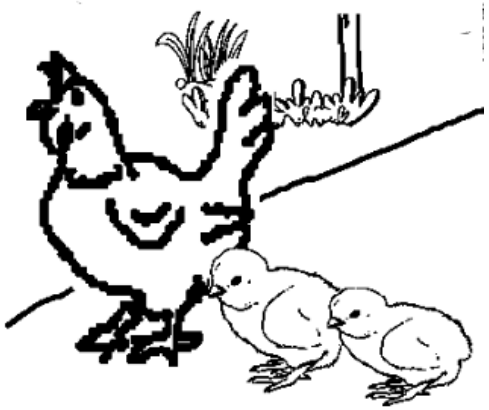


遇到的问题

重叠对象检测困难

重合部分比较多的地方没能很好检测出来。



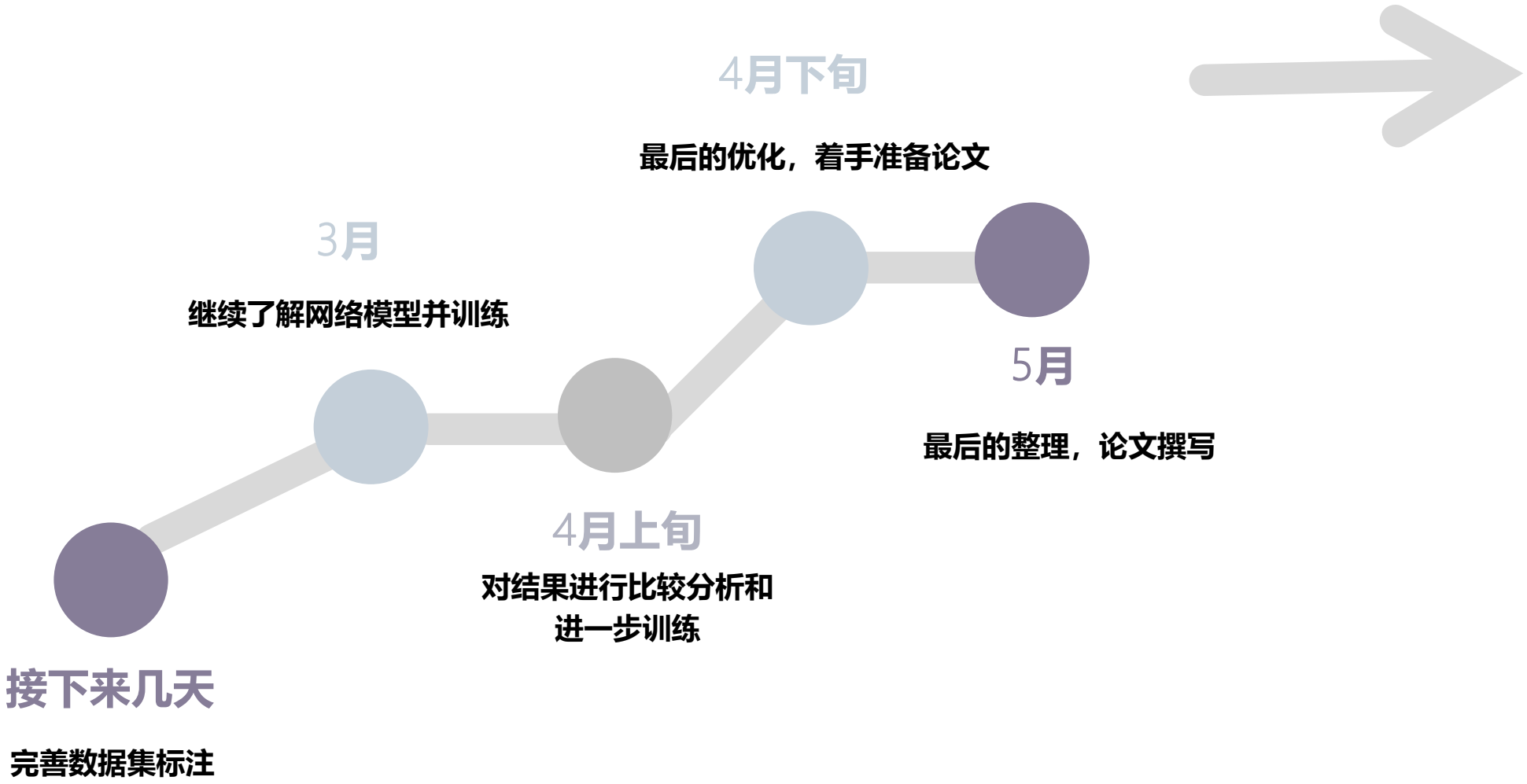


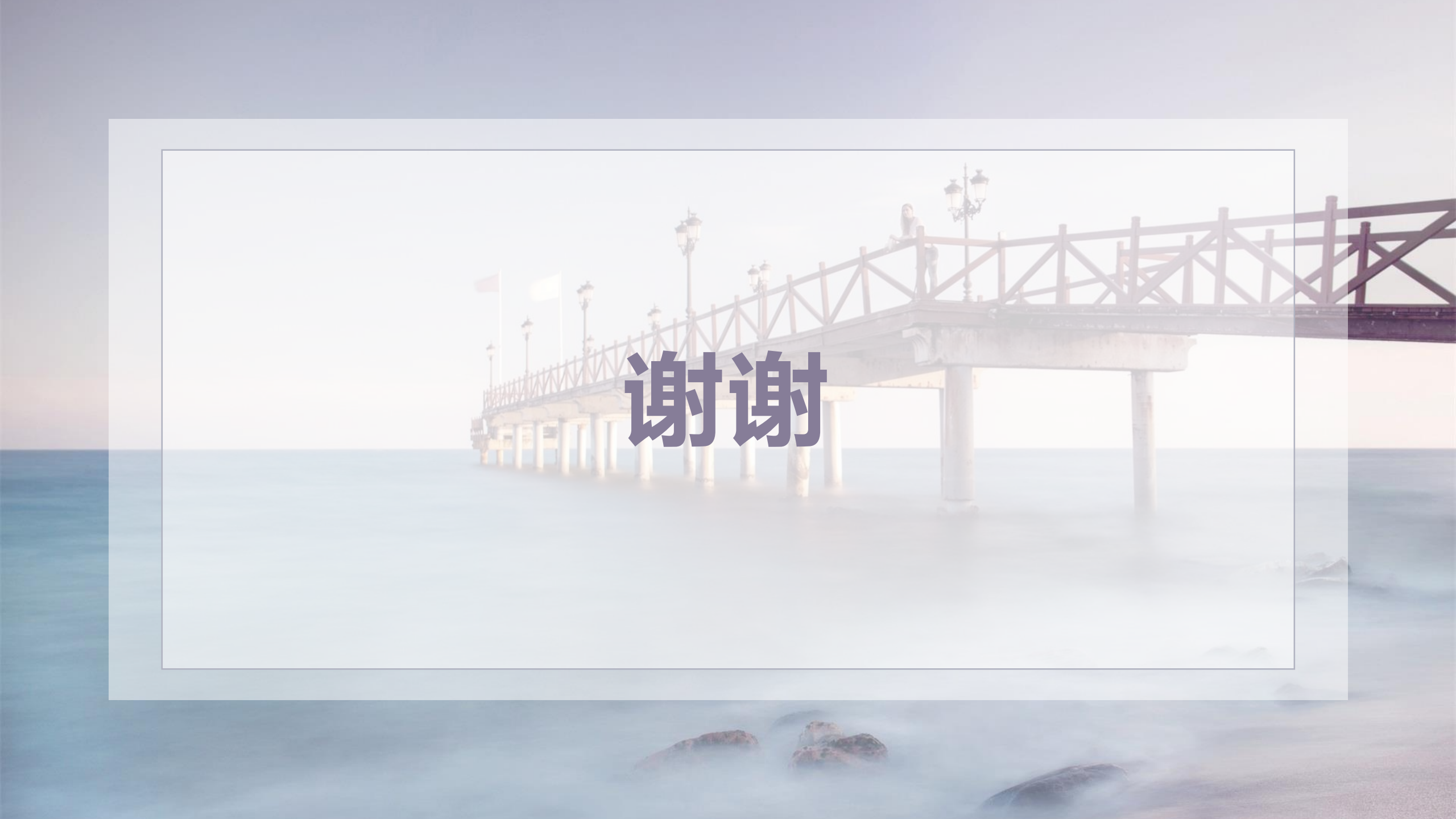
即使定义了先验框也可能会存在一些问题

- 当两个形态相似的目标重合（例如两辆车重合），那么检测车的先验框只能检测出一辆车。
- 此外，如果三个重叠对象但是只定义了两个先验框，此时也只能检测到两个目标。

后续安排

后续安排


接下来几天
完善数据集标注
3月
继续了解网络模型并训练
4月上旬
对结果进行比较分析和
进一步训练
4月下旬
最后的优化，着手准备论文
5月
最后的整理，论文撰写


谢谢