基于场景草图的
目标定位与检测技术研究

汇报人　刘卓云

目录

CONTENTS

YOLO算法了解

YOLO算法了解

R-CNN

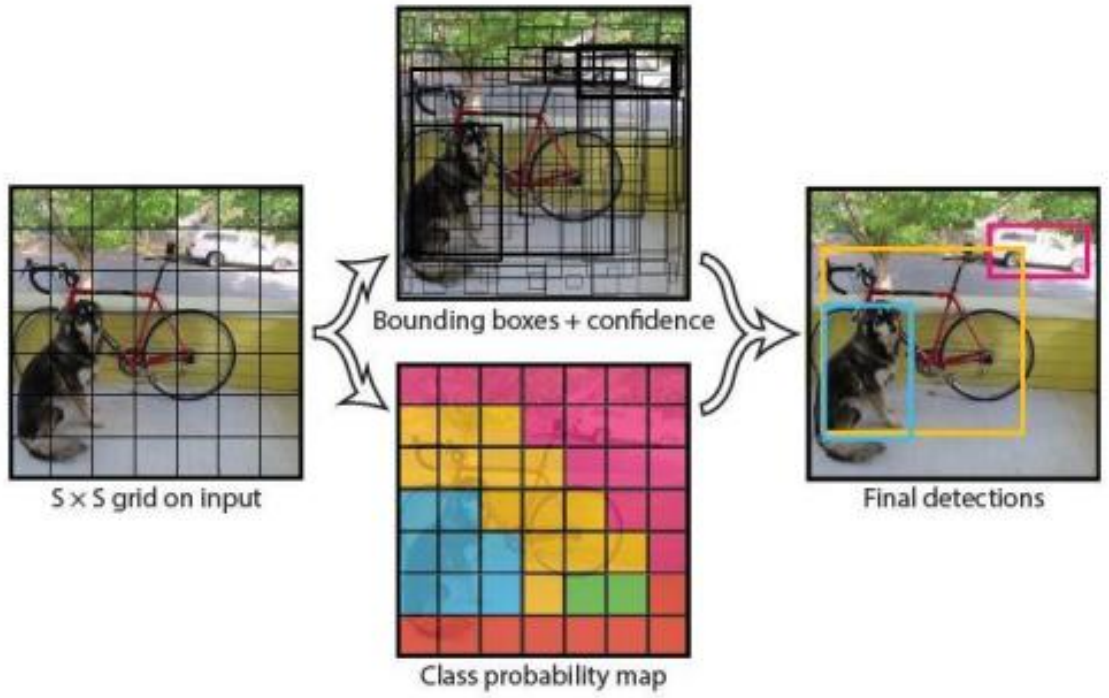
Two-Stage 目标检测算法将检测划为两个阶段，先产生Region Proposal，然后再在Region Proposal上进行分类与回归。

YOLO

One-Stage 检测算法不需要 Region Proposal 的阶段，直接使用一个 CNN 网络预测不同目标的类别与位置。

由于YOLO速度更快，后续修改、调试可能会更方便一些，暂定使用YOLO进行训练。

YOLO算法了解



S×S ×(5B+C)

C为目标种类个数

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 B 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

如图，目标中心点落在红色格子内，所以检测该（黄色边界框中）目标由红色格子负责检测。边界框实际坐标不需要位于网格单元内，可以超出网格单元的范围。

YOLO算法了解

由于每个单元格都要进行预测，结果可能会有不同单元格检测到同一物体。此时可以使用交并比（IoU），找到交并比很大的所有边界框，只保留置信度最高的一个。

如上图，有三个单元格检测到了目标，最后只保留红色单元格检测的结果。

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。

YOLO算法了解

即使定义了先验框也可能会存在一些问题

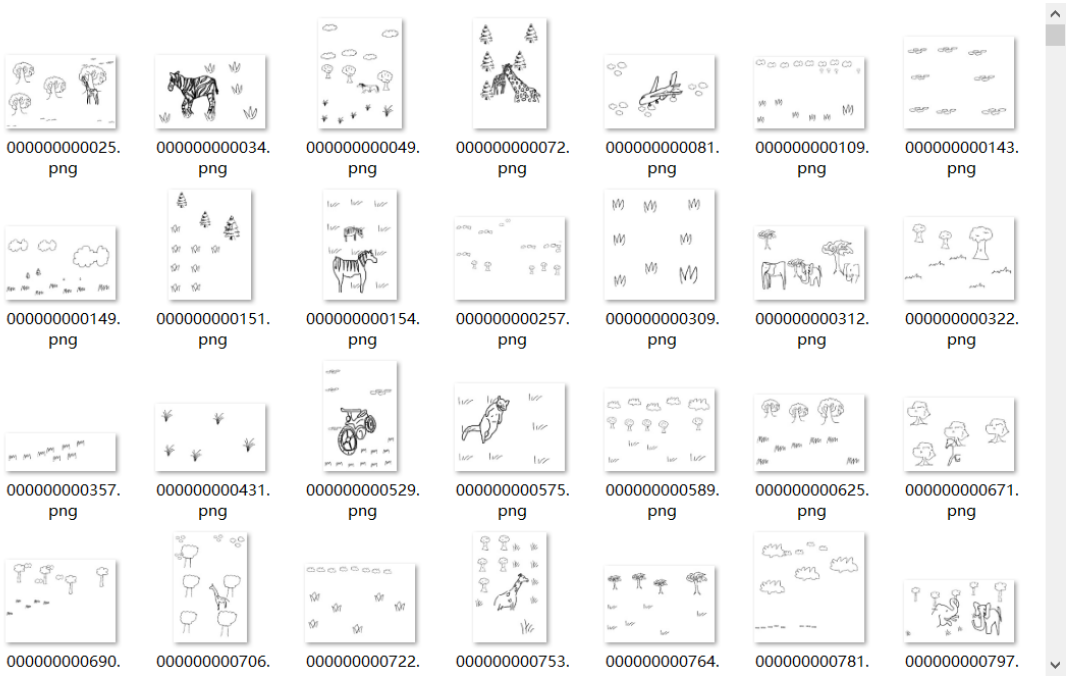
- 当两个形态相似的目标重合（例如两辆车重合），那么检测车的先验框只能检测出一辆车。
- 此外，如果三个重叠对象但是只定义了两个先验框，此时也只能检测到两个目标。

但是重叠的现象大部分存在于自然图像中。场景草图中这样的情况比较少见。

数据集准备

数据集准备

SketchyCoCo



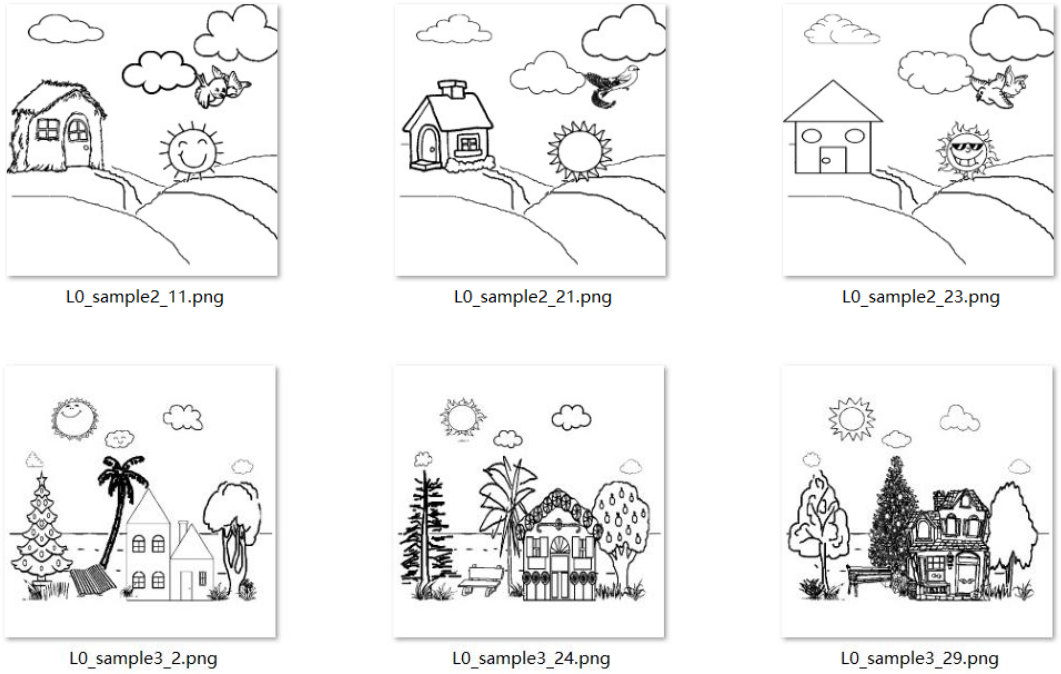
场景级 约14000

对象级 14个类别

数据集准备

SketchyScene

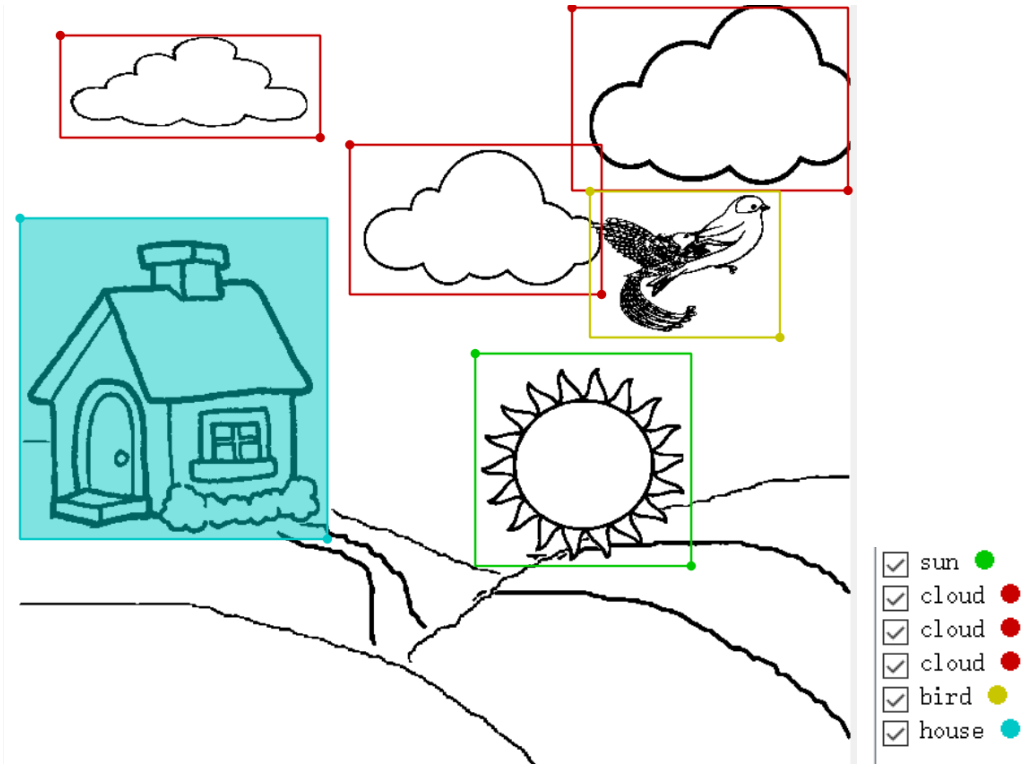
不同场景 约5600



相同场景 3张

数据集准备

Labelme标注



L0_sample1.json	2021/1/28 21:55	Adobe After Eff...	15 KB
L0_sample1_6.json	2021/1/28 21:31	Adobe After Eff...	17 KB
L0_sample1_10.json	2021/1/28 21:29	Adobe After Eff...	17 KB
L0_sample1_11.json	2021/1/28 21:30	Adobe After Eff...	18 KB
L0_sample2_11.json	2021/1/28 18:54	Adobe After Eff...	17 KB
L0_sample2_21.json	2021/1/28 18:55	Adobe After Eff...	16 KB
L0_sample2_23.json	2021/1/28 18:58	Adobe After Eff...	16 KB
L0_sample3_2.json	2021/1/28 21:20	Adobe After Eff...	21 KB
L0_sample3_24.json	2021/1/28 21:23	Adobe After Eff...	25 KB
L0_sample3_29.json	2021/1/28 21:27	Adobe After Eff...	29 KB
L0_sample4_15.json	2021/1/28 21:34	Adobe After Eff...	24 KB
L0_sample4_19.json	2021/1/28 21:36	Adobe After Eff...	20 KB
L0_sample4_25.json	2021/1/28 21:39	Adobe After Eff...	25 KB
L0_sample5_7.json	2021/1/28 21:45	Adobe After Eff...	22 KB
L0_sample5_13.json	2021/1/28 21:41	Adobe After Eff...	24 KB
L0_sample5_18.json	2021/1/28 21:43	Adobe After Eff...	21 KB
L0_sample6_3.json	2021/1/28 21:50	Adobe After Eff...	28 KB
L0_sample6_16.json	2021/1/28 21:47	Adobe After Eff...	27 KB
L0_sample6_19.json	2021/1/28 21:48	Adobe After Eff...	26 KB
L0_sample10.json	2021/1/28 21:54	Adobe After Eff...	14 KB
L0_sample100.json	2021/1/28 22:00	Adobe After Eff...	19 KB
L0_sample101.json	2021/1/28 22:09	Adobe After Eff...	18 KB

数据集准备

Labelme标注

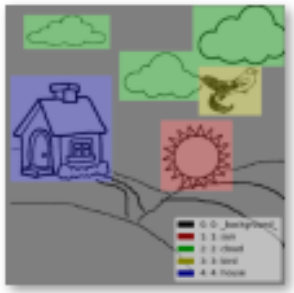
labelme_json_to_dataset.exe


img.png


info.yaml


label.png


label_names.txt


label_viz.png

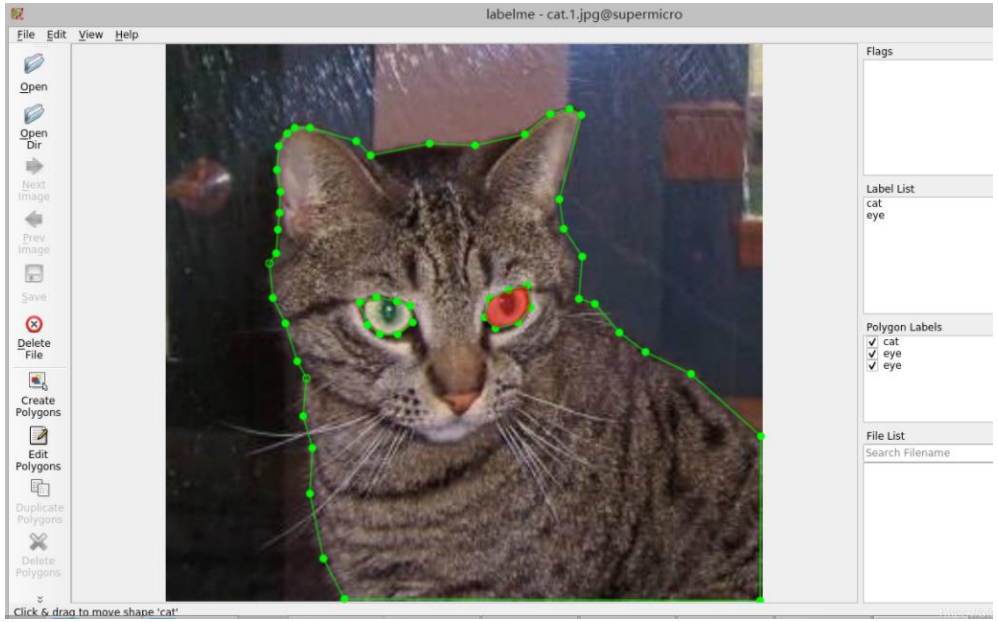
后来发现YOLO其实不需要进行这一步操作, Labelme可能更适合做语义分割

遇到的问题

遇到的问题

更合适的标注工具

labelimg



labelme

labelme生成的.json文件也可以进行转换，但是使用labelimg可能会更方便

遇到的问题

多尺度问题

YOLO的多尺度适应性一般，样本需要尽量多样化，长宽比例、物体角度等。

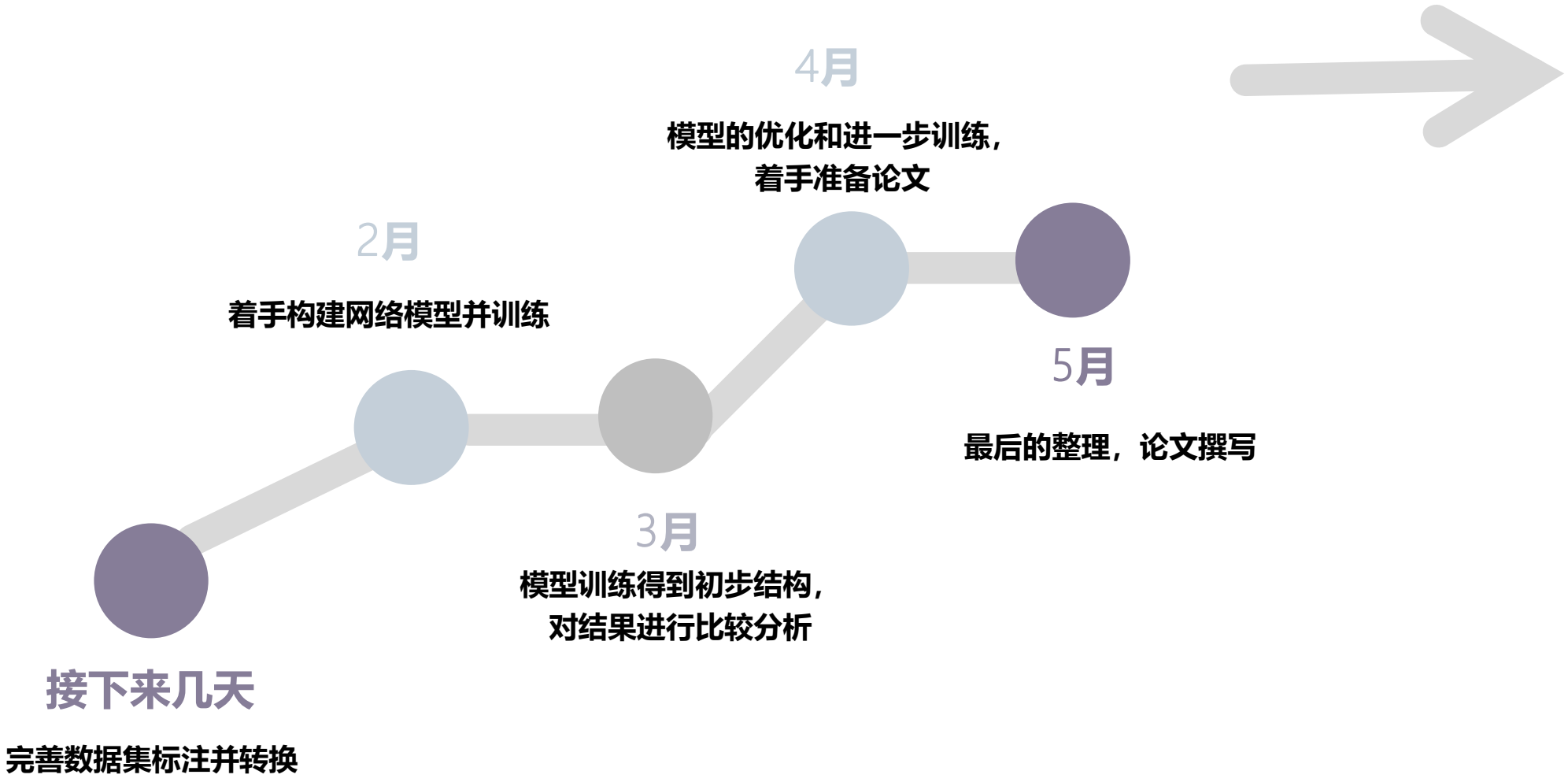
损失函数

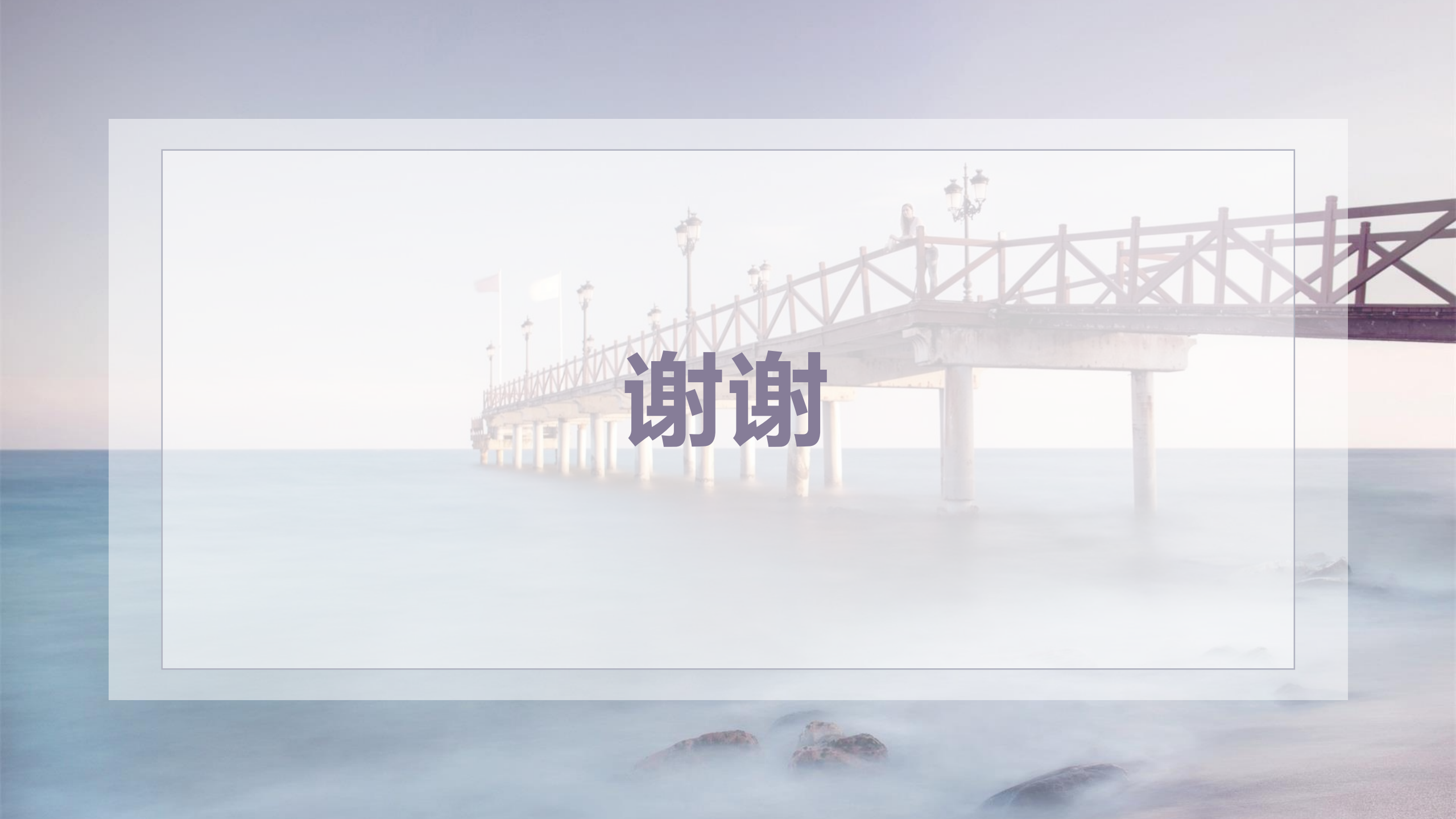
YOLO 系列算法除了 Cls Loss 和 Loc Loss 外还有 Obj Loss。一些实验表明YOLO算法的正负样本不均衡。

可以使用Hard Negative Mining和 Focal Loss 进行优化。

后续安排

后续安排


接下来几天
完善数据集标注并转换
2月
着手构建网络模型并训练
3月
模型训练得到初步结构，
对结果进行比较分析
4月
模型的优化和进一步训练，
着手准备论文
5月
最后的整理，论文撰写


谢谢