徽标, 公司名称

描述已自动生成

**2022年（第15届）**

**中国大学生计算机设计大赛**

人工智能实践赛作品报告

作品编号：　　　　　 2022051038

作品名称：DeepSafe——COVID-19大流行中的社交距离监测和风险评估系统

填写日期：　　　　　　2022-05-01

填写说明：

1. 本文档适用于人工智能实践赛小类；
2. 正文、标题格式已经在本文中设定，请勿修改；标题#的快捷键为“Ctrl+#”，正文快捷键为“Ctrl + 0”；
3. 本文档应结构清晰，突出重点，适当配合图表，描述准确，不易冗长拖沓；
4. 提交文档时，以PDF格式提交；
5. 本文档内容是正式参赛内容的组成部分，务必真实填写。如不属实，将导致奖项等级降低甚至终止本作品参加比赛。

**目 录**

[第1章 作品概述 1](#_Toc102506243)

[1.1 产生背景 1](#_Toc102506244)

[1.2 用户群体 1](#_Toc102506245)

[1.3 主要功能与特色 1](#_Toc102506246)

[1.4 应用价值 1](#_Toc102506247)

[1.5 推广前景 2](#_Toc102506248)

[第2章 问题分析 2](#_Toc102506249)

[2.1 问题来源 2](#_Toc102506250)

[2.2 现有解决方案 3](#_Toc102506251)

[2.3 本作品要解决的痛点问题 4](#_Toc102506252)

[2.4 解决问题的思路 4](#_Toc102506253)

[2.4.1 作品的功能需求 4](#_Toc102506254)

[2.4.2 作品的性能需求 4](#_Toc102506255)

[2.4.3 使用的数据集 5](#_Toc102506256)

[第3章 技术方案 7](#_Toc102506257)

[3.1 关键技术 7](#_Toc102506258)

[3.1.1 YOLOv5s -目标检测 7](#_Toc102506259)

[3.1.2 Deep SORT -多目标追踪 7](#_Toc102506260)

[3.1.3 DBSCAN -人群聚类 8](#_Toc102506261)

[3.1.4 社交距离计算 8](#_Toc102506262)

[3.2 作品特色 10](#_Toc102506263)

[第4章 系统实现 11](#_Toc102506264)

[4.1 数据的加工和扩增 11](#_Toc102506265)

[4.2 数据管理 12](#_Toc102506266)

[4.3 用户界面 12](#_Toc102506267)

[第5章 测试分析 14](#_Toc102506268)

[5.1 数据来源 14](#_Toc102506269)

[5.2 数据规模 14](#_Toc102506270)

[5.3 环境配置 15](#_Toc102506271)

[5.4 测试过程 15](#_Toc102506272)

[第6章 作品总结 17](#_Toc102506273)

[6.1 作品特色与创新点 17](#_Toc102506274)

[6.2 应用推广 18](#_Toc102506275)

[6.3 作品展望 18](#_Toc102506276)

[参考文献 19](#_Toc102506277)

# 作品概述

## 产生背景

2022年，疫情依旧没有消失。多地新冠疫情出现大规模聚集传播，现阶段全国疫情的聚集化传播扩散面积还在急剧增加。疫区正常生产活动遭到重大干扰，人民安全没有保障，疫情聚集化扩散制约经济和社会持续发展。

专家表示，未佩戴口罩，相隔0.5米左右就能造成疫情传播。因此，保持社交距离是对抗Covid-19的重要手段。然而，人工监督防疫工作耗费警力大，疏散聚集人群效率低，并且人工管控对于防疫人员来说也具有一定感染风险。为了解决这个难题，我们开发了Deepsafe——COVID-19大流行中的社会距离监测和感染风险评估系统。

## 用户群体

疫情防控下的全体居民。

## 主要功能与特色

* YOLOv5 行人检测与口罩佩戴检测；
* Deepsort行人目标追踪；
* DBSCAN行人聚类；
* 社交距离计算与感染风险评估；
* 路径规划自主巡检。

## 应用价值

本作品解决了人工监督防疫工作耗费人力物力、疏散聚集人群效率低、防疫人员感染风险高等问题，以无人机为载体，通过自动化执勤、随调随飞，实现防疫工作自动化，建立了有效的监管手段，以科技赋能疫情防控，真正实现“科技服务民生”。

## 推广前景

由于新冠疫情感染性强、传播范围广，而人工防疫存在效率低、耗费人力、自身感染风险高等问题，因此在易出现人群聚集的公共场所亟需一款具备自动化监测社交距离、评估感染风险等功能的产品，来分担疫情防控工作的压力。因此，我们的作品Deepsafe作为空中防疫尖兵，契合上述市场需求，具有良好的推广前景。

# 问题分析

## 问题来源

我国专家对于疫情形势的估计一直持谨慎态度，认为新冠病毒变异一直在发生，新的变异毒株可能还会出现。我们对新冠流行规律的认识还非常有限，从发病和死亡两个流行曲线分析过去两年多的起伏变化，疫情防控形势仍然严峻复杂。上海市连续8天报告新增感染者数超过1万例，疫情处于快速上升期，社区传播尚未得到有效遏制，且外溢多省多市，预计未来几日新增感染者人数仍将处于高位。3月1日至4月11日，31个省（区、市）和新疆生产建设兵团累计报告本土感染者超32万例，波及30个省（区、市）。

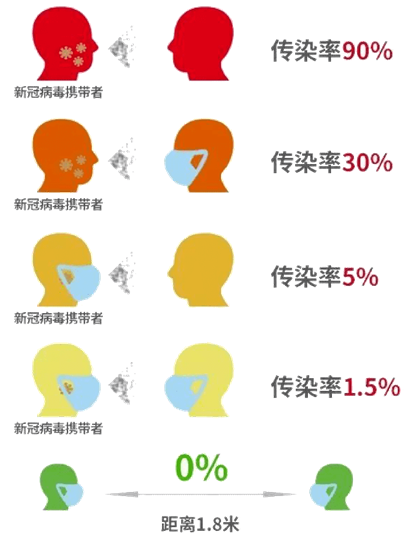


图 1 社交距离与传染率的关系

新冠肺炎有三大传播途径：飞沫传播、接触传播和粪口传播，由此看来，减少人员聚集、保持社交距离、戴口罩出门已经成为硬性要求，但是存在部分民众防疫意识太差，不执行防疫措施，大大增加病毒传播的风险。而人工监督防疫工作存在耗费人力物力、疏散聚集人群效率低、防疫人员感染风险高等一系列问题，因此，如何实现疫情防控工作自动化是迫在眉睫、亟需解决的社会痛点问题。

## 现有解决方案

目前采用的疫情防控手段主要有：

* 人工监督

人工监督费时费力，提醒群众保持社交距离，此方法耗费人力物力大，效率得不到保障，人工管控对于防疫人员来说也具有一定感染风险，是低效率的无奈之举。

* 事后查询公共场所监控。

在发现新冠肺炎确诊病例后，在确诊患者到达的每一个公共场所逐一排查监控记录，找出密接者，显然也非常耗时耗力。

因此，自动化与AI结合的措施能大大提高防疫效率，使疫情防控立体化、全覆盖，更具针对性、时效性。

## 本作品要解决的痛点问题

在当前危机中，减少传播的措施之一是隔离。尽管许多城市现在都在谨慎地重新开放，但是人们在外出时仍需要保持安全距离。因此城市对人们的安全距离是否符合规则进行评估并采取相应的行动是很重要的。如果大多数人都遵守疫情期间的命令，那么就可以安全地开放更多的公共场合。

本作品要解决的痛点问题是，针对特定的应用场景——疫情防控，基于YOLOv5 与深度排序多目标跟踪，以无人机为载体，实现自动化监测社交距离和评估感染风险，减轻疫情防控人员和志愿者维持公共场所秩序的压力，并且提高社交距离监测的准确度，促进疫情防控科技化、自动化。

## 解决问题的思路

### 作品的功能需求

* 对一个或多个行人的精准识别。
* 对一个或多个行人进行低延迟动态追踪。
* 计算视频中两个人之间的像素距离，并转换为现实距离。
* 进行DBSCAN算法聚类分析
* 检测行人是否佩戴口罩。
* 结合社交距离及口罩佩戴情况，评估是否存在感染风险。

### 作品的性能需求

1. 空间特性：数据最大储存不少于20GB。
2. 数据精确度：
   1. 在执行数据的增加、删除、修改操作时，不允许因为程序原因导致操作失败。
   2. 在执行数据增加时，不允许发生多增加或重复增加数据的情况。
   3. 在执行数据删除时，对有关联的数据要求册除完全，如不能删除，请给予提示。
   4. 在执行数据修改时，也要求保持对应的准确性。
3. 时间特性
   1. 用户执行监控查询导出等操作时，数据库响应时间要求在0.8秒之内。
   2. 当有10个以上的用户同时对系统执行查询操作时，系统的相应时间应当不多于2秒，页面刷新频率应当在0.2次/秒~0.3次/秒。
   3. 识别的延迟要求在1s以下。

### 使用的数据集

表 1 数据集说明

|  |  |
| --- | --- |
| 标注编号(label) | 描述 |
| 0 | 行人(people) |
| 1 | 佩戴口罩的脸(masked) |
| 2 | 未佩戴口罩的脸(unmasked) |
|  |  |

我们通过划分几个类别，针对确定的类别进行数据获取，主要考虑为行人(people)、佩戴口罩的脸(masked)、未佩戴口罩的脸(unmasked)， 于是我们发现了 Kaggle——数据建模和数据分析竞赛平台，其提供了大量开源优质数据集，我们的物体识别模型所需要数据部分来源于 https://www.kaggle.com/，其余来源于自我收集与标注，其中包含了各类常见的人群与口罩的数据图片，同时我们从网页和现实生活中采集了一些数据，增加其数据的鲁棒性。

我们一共收集了2351张图像，对于每张图像，都标注了其中行人、口罩的所在位置。图像不是单一尺寸的，它们有不同的长宽大小。对于本作品，我们使用视频中人的真实位置来进行分割，然后创建一个目录，其中包含每个视频序列的每个人的样本。 目录中的样本具有原始尺寸，但在我们的实验中将其调整为32x64像素值。

图 2 数据样例1



图 3 数据样例2



图 4 数据样例3

# 技术方案

## 关键技术

### YOLOv5s -目标检测

捕捉速度快，运算量较小，能实现多目标多样本的同时捕捉。行人检测与行人重识别如何更好的进行结合。reid使得行人检测可以存在多一些的误检，要求recall比较高，因为一些背景可以通过reid进行相似度比较从而排除掉。但是如过置信度阈值设的太低，会有很多的proposal要进入reid模型，降低模型的速度。因此，需要进行合理的设置，本作品就设置了conf-thres=0.1，nms-thres=0.4，最后，对行人重识别模型进行剪枝、蒸馏，加快推理速度。

本作品以无人机为载体，视频造成模糊的原因有：① 快速移动 ② 面部失焦 ③ 视频噪声，这会大大影响检测的精度，所以我们对数据集进行运动模糊（模仿快速运动）、平均模糊（模仿失焦），高斯模糊（模拟随机噪声）来增强处理，更好的适应实际应用，如图：

****

图 5 初始图像及三种模糊图像

### Deep SORT -多目标追踪

目标跟踪速度快，精准度高，可绘制物体轨迹，便于追踪个体。无人机能进行播报警告，当存在感染风险时，无人机将发出提醒或警报，并回传数据，本作品的追踪流程如下图所示：

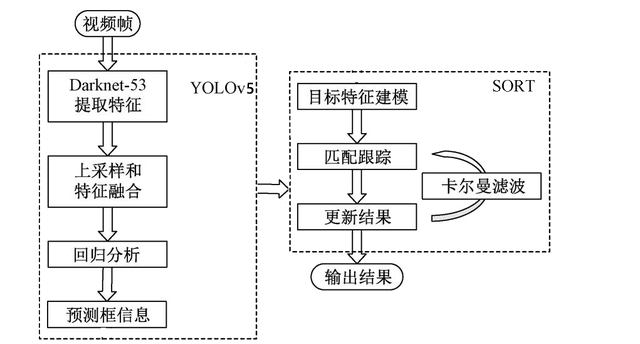


图 6 多目标追踪流程图

结合口罩佩戴情况与社交距离分别在安全或有感染风险时标记绿色或橙色边界框：

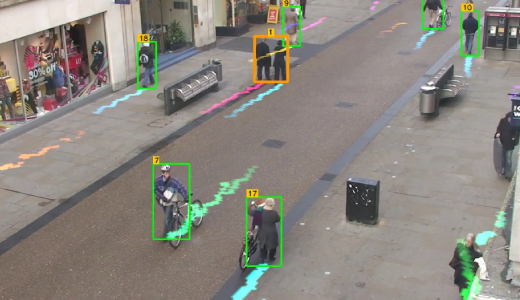


图 7 效果展示

### DBSCAN -人群聚类

DBSCAN算法 是一种基于密度的聚类算法：

• 聚类的时候不需要预先指定簇的个数。

• 最终的簇的个数不定。

### 社交距离计算

在正式应用社交距离之前，需要两种类型的距离校准，以实现像素距离到现实距离的转换。

**1.相机视图到顶视图的转换**

通过机载摄像头的拍摄角度计算变换矩阵将摄像机视图转换为俯视图图像。

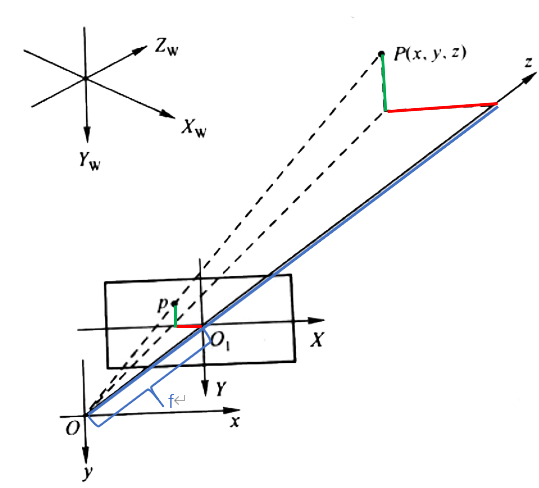
****

图 8 视图转换

2**. 比例因子的估计**

人的身高约为 5.6 英尺（头部和脚部位置的像素点间距）, 通过算法估计比例因子转换以英尺为单位的像素距离。



图 9 人的身高

通过距离校准实现像素距离到现实距离的转换，评估感染风险(试验场景：本校学生活动中心)。

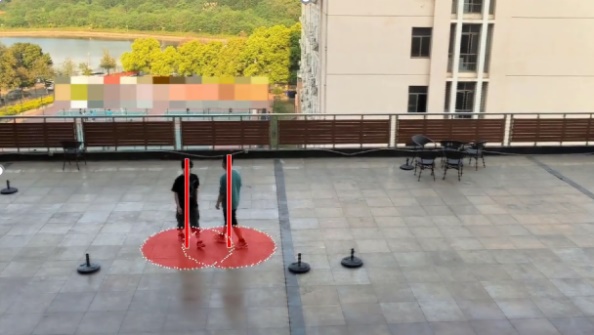


图 10 距离的转换

## 作品特色

**1. 防疫工作覆盖面广**

为YOLO检测与Deepsort多目标追踪算法提供了极佳的鸟瞰图视角，保证AI视觉的有效执行，实现“应检尽检、不漏一人”。



图 11 无人机展示

**2. 立体式作业**

根据像素距离计算社交距离，评估感染风险并以高空自动播报的方式执行宣传任务，突破道路限制，迅速穿梭在城市的每个角落，强机动性使宣传效果和人员安全达到最大化。

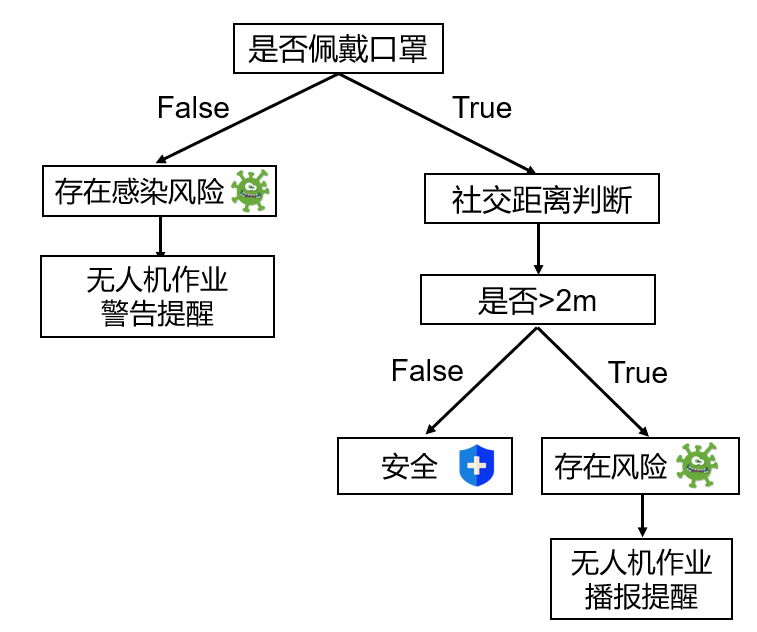


图 12 风险评估流程

详细需求流程设计如下：

主要分为两部分：行人检测聚类与社交距离计算，大体流程如下：

1. 使用YOLOv5模型检测输入视频或视频流中的所有行人，得到行人坐标，并根据此计算位置信息和质心位置；

2. 根据第1步的信息，计算所有检测到的行人的人质心之间的相互距离；

3. 根据上一步计算得到的社交距离，进行人群聚类分析，对比预先设置的安全距离，从而计算每个人之间的距离对，结合口罩佩戴情况，检测两个人之间的距离是否小于N个像素，小于则处于安全距离，反之则不处于安全距离。

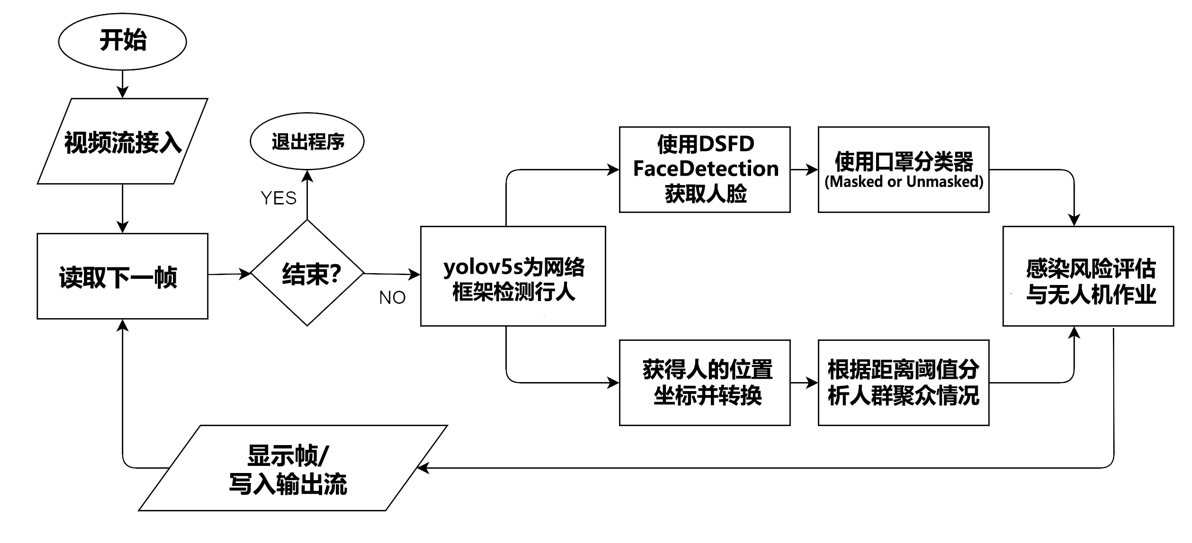


图 13 详细需求设计流程

# 系统实现

我们的作品是基于openbayes与pytorch实现的，首先在openbayes上完成数据集的标注与发布，使用AI市场算法进行训练，训练资源可选Ascend 910或GPU；然后在openbayes (Beta)中转换模型，调试代码，实现行人检测与跟踪以及简单人数统计，整体开发在OpenBayes上完成，从零开始，最终进行Serving 模型部署上，独立运行。

## 数据的加工和扩增

因为有些数据背景的复杂性，相关目标需要标注，网上有可以自动标注的服 务但是由于需要付费，于是我们最终花了几天时间手动标注了数据，并生成了一 些负样本，防止训练的过拟合。

因为实验前期的数据量不太可观，担心数据的数量不能很好的使得网络达到 良好的训练效果，尤其是遇到小目标时，于是我们主要采用了几种数据扩增的方 式：

（1） Mosaic

Mosaic 是把四张训练图片缩放拼成一张图，Mosaic 有利于提升小目标的检 测，这是因为一般在数据集中小目标在图片中分布不均匀，这导致在常规的训练 中小目标的学习总是不太充分。使用 mosaic 数据增强后，在遍历每个张图片包 含了四张图片具有小目标的可能性就很大了，同时，每张图都有不同程度的缩小， 即使没有小目标，通过缩小，原来的目标尺寸也更接近小目标的大小，这对模型 学习小目标很有利。

（2） Cutout

Cutout 是随机选择一个固定大小的正方形区域，然后采用全 0 填充，同时 为了避免填充 0 值对训练的影响，对数据进行中心归一化操作，norm 到 0。

（3） 矩形训练

通常 yolo 算法解决输入图像大小比率不同的方法是直接缩放和填充到固定 的大小，但是这样将导致有很多冗余的信息，会让网络产生很多无意义的候选框。 矩形训练就是将图像 resize 到变成可以被步长整除并且最接近需要输入的大小， 从而实现最小填充。

## 数据管理

将处理好的数据先分成多个目标类文件，再生成对应的正样本和负样本样例， 在训练时将正样本和负样本组合打乱。

## 用户界面

对于该项目，我们采用了前后端分离的整体架构。前端使用Vue响应式框架进行开发，后端采用SpringBoot框架与前端、数据库进行数据交互，数据库采用“MySQL + ElasticSearch”进行数据存储和检索。

**前端**：

百度地图API：百度地图API是一套为开发者免费提供的基于百度地图的应用程序接口,提供基本地图、位置搜索、周边搜索等功能。我们基于百度地图实现被无人机关注感染者的行程轨迹在地图上的动态渲染和静态展示。

Vue响应式框架：Vue是一套用于构建用户界面的渐进式框架，Vue的核心库只关注视图层，不仅易于上手，还便于与第三方库或既有项目整合。使用Vue组件作为基本组件，不仅可以一处编写多处复用，提高开发效率，同时还能方便地集成百度地图。

**后端**：

1）SpringBoot：Spring是一个开源框架，是为了解决企业应用开发的复杂性而创建的，是一个轻量级的控制反转(IOC)和面向切面(AOP)的容器框架。而SpringBoot能够简化Spring的配置，使开发者能够更加专注于业务逻辑。我们利用SpringBoot与数据库进行连接，并将经过处理的数据以接口的方式提供给前端进行调用，达到前后端分离的效果。

2）数据库：

① 数据获取：无人机传回的实时视频流

② 数据存储：

MySQL数据库：MySQL是最流行的关系型数据库管理系统，在 WEB 应用方面 MySQL 是最好的 RDBMS应用软件之一。我们将获取到的数据存储至MySQL中，实现数据的持久化，同时将数据同步至ElasticSearch中。

③ 数据检索：

ElasticSearch：ElasticSearch是基于Lucene的全文检索引擎，它会对数据进行分词后保存索引，擅长管理大量的索引数据，能够提高用户的检索效率。

**作品特色**：

界面设计精巧：

基于百度地图，界面简洁、地图覆盖范围广，用户满意度较高。使用百度地图作为背景更加贴近用户，遵循用户使用习惯，提高数据呈现效果的可信度，更具权威性；

基于Element组件库，Element是基于Vue2.0桌面端组件库，设计简洁美观，交互动效让用户可以清晰的感知自己的操作，即反馈性好；同时用户可以自由地进行操作，根据场景可给予用户操作建议或安全提示，即可控性好；

动态呈现：

基于百度地图API和无人机巡航，通过POI地址解析，生成可疑病例的动态行动轨迹图，其中包括途径地点以及经过时间，数据详细且直观，能够帮助群众根据行程轨迹进行自我排查，或者为群众提供出行参考，利于有效控制疫情扩散。

# 测试分析

## 数据来源

为了确保识别模型的普适性与非针对性,我们采用开源数据集开展数据训练, 即 Kaggle——数据建模和数据分析竞赛平台，其提供了大量开源优质数据集，我们的口罩识别模型所需要数据主要来源于https://github.com/X-zhangyang/Real-World-Masked-Face-Dataset，其中包含了各类常见的佩戴口罩的数据图片，此外我们的行人训练数据集是从WiderPerson,https://wider-challenge.org/2019.html获取。同时我们从网页和现实生活中采集了一些数据，增加其数据的鲁棒性,并且我们也添加了适量的负样本数据,以追求进一步提高识别的精确性。

## 数据规模

我们的口罩识别训练数据包含了 10000 多张的.jpg 后缀图片，保证数据集普适性。行人数据集使用了80000多张图片用于训练。

## 环境配置

我们使用到的环境有如下:

python 3.7

Anaconda3

Pycharm

cuda10.2 cudnn-10.2-windows10-x64-v7.6.5.32 pytorch1.5.1-gpu

我们运用到的工具包以及其建议版本如下:

Keras==> 2.2.4

opencv-python==> 4.2.0.32

numpy==> 1.16.2

tensorflow==> 1.14.0

sklearn==> 0.20.3

scipy==> 1.2.1

我们使用 Anaconda3 来安装所有必要的软件环境：

安装 opencv： conda install opencv

安装 mingw libpython： conda install mingw libpython

安装 theano： conda install theano

安装 tensorflow-gpu： conda install tensorflow-gpu

安装 sklearn： conda install scikit-learn

安装 matplotlib： conda install matplotlib

安装 keras： conda install keras

## 测试过程

我们使用 1660Ti+i7-8750 的笔记本电脑，采用 72000 张图片(包含口罩类 8000张图片，行人类 64000 张图片)输入，训练 1000 轮次对于行人类和口罩类的模型训练。完成训练后会在yolov5文件夹的run下保存有训练模型以及训练参数的图表如下图所示。

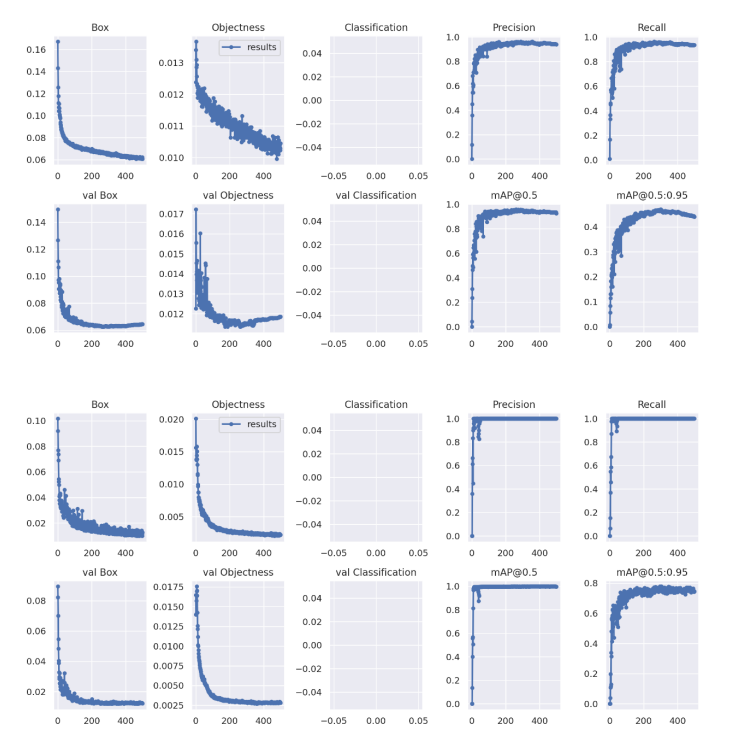


图 14 训练模型与训练参数

完成训练后会在 yolov 文件夹的 run 下保存有训练参数的图表如下图所示：

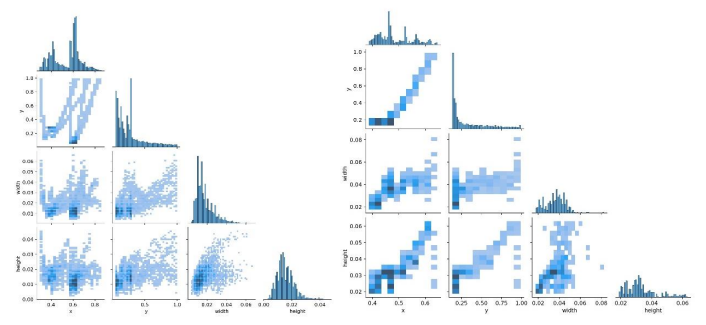


图 15 训练参数

根据图表,结果表明了训练结果具有较高的收敛速度(在 200 个 epoch 后趋 向于收敛)和较高的识别准确性(高达 95%及以上)。

下图为行人检测与社会距离判断的总体精度数据显示。

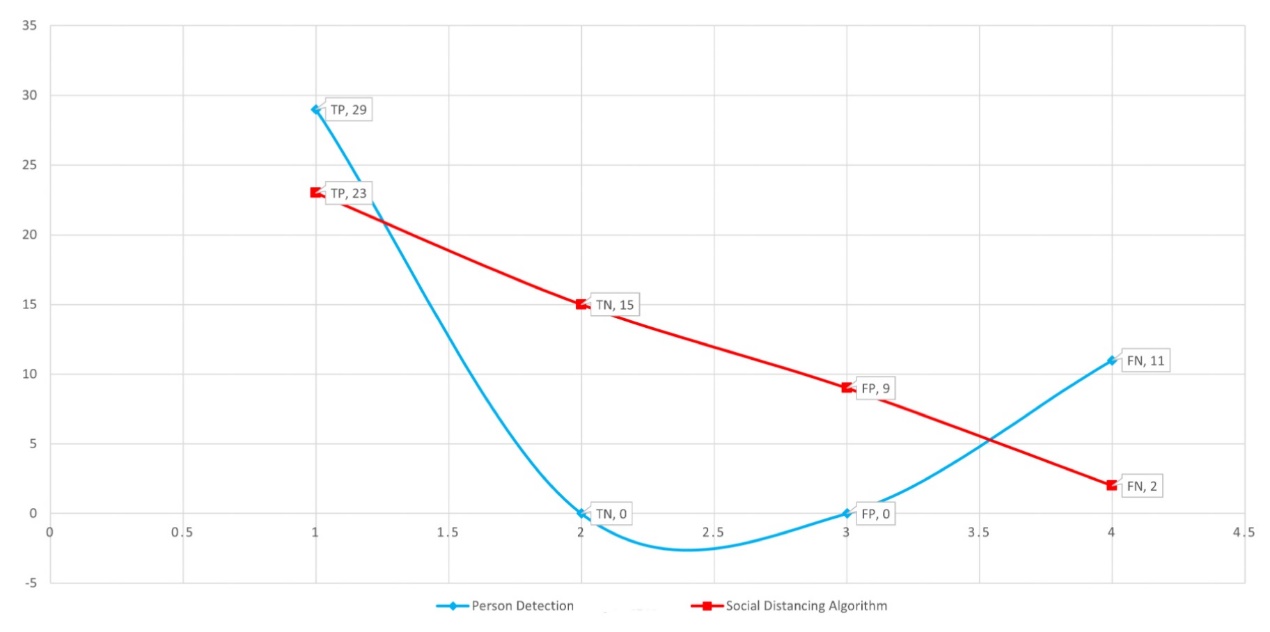


图 16 行人检测与社会距离判断的总体精度数据

# 作品总结

## 作品特色与创新点

* **覆盖面广**

为YOLO检测与Deepsort多目标追踪算法提供了极佳的鸟瞰图视角，保证AI视觉的有效执行，实现“应检尽检、不漏一人”。

* **立体式作业**

评估感染风险并以高空自动播报的方式执行宣传任务，突破道路限制，迅速穿梭在城市的每个角落，进行DBSCAN行人聚类，评估感染风险，强机动性使宣传效果和人员安全达到最大化。

* **自主巡检**

为疫情重灾区做出路径规划，定时自动执行巡检任务，无接触式灵活作业避免了交叉感染的可能性，提供7×24小时无人化巡检。

## 应用推广

本作品解决了人工监督防疫工作耗费人力物力、疏散聚集人群效率低、防疫人员感染风险高等问题，以无人机为载体，通过自动化执勤、随调随飞，实现防疫工作自动化，建立了有效的监管手段，以科技赋能疫情防控，真正实现“科技服务民生”。推广领域包括：

1、疫区源头喷洒消毒；

2、防疫任务路径规划；

3、聚集场所实时巡检和监控；

4、智能识别口罩佩戴；

5、防疫站点筛查异常体温人群；

6、隔离疫区物资投递。

## 作品展望

通过“高空巡查、低空劝返、悬停监控”等方法，DeepSafe致力于搭建起“人防+技防”“空中+地面”的全覆盖、立体式的防疫安全网。

# 参考文献

[1]Joseph Redmon et al. “You Only Look Once: Unified, Real-Time Object Detection” computer vision and pattern recognition (2016): n. pag.

[2]qiu xiaofeng et al. “Pedestrian detection and counting method based on YOLOv5+DeepSORT” 4th International Symposium on Power Electronics and Control Engineering (ISPECE 2021) (2021): n. pag.

[3] Joseph Redmon and Ali Farhadi. “YOLOv3: An Incremental Improvement..” arXiv: Computer Vision and Pattern Recognition (2018): n. pag.

[4] Kamran Khan et al. “DBSCAN: Past, present and future” international conference on applications of digital information and web technologies (2014): n. pag.

[5] 蔡莹莹. “以“人”为本的用户界面设计流程研究” 才智 (2011): n. pag.

[6] Joseph Redmon and Ali Farhadi. “YOLO9000: Better, Faster, Stronger” computer vision and pattern recognition (2017): n. pag.

[7] Mazin Hnewa and Hayder Radha. “Integrated Multiscale Domain Adaptive YOLO” (2022).

[8] Immanuel Jose C. Valencia et al. “Vision-based Crowd Counting and Social Distancing Monitoring using Tiny-YOLOv4 and DeepSORT” ieee international smart cities conference (2021): n. pag.