

中国科学技术大学

专业硕士学位论文

(专业学位类型)



吸烟检测系统的设计与实现

作者姓名： 吴俊佳

专业领域： 软件工程

校内导师： 汪增福 教授

企业导师： 侯锐 研究员

完成时间： 二〇二一年十月二十八日

University of Science and Technology of China
A dissertation for master's degree

(Professional degree type)



**Design and implementation of
smoking detection system**

Author:	Junjia Wu
Speciality:	Software Engineering
Supervisor:	Prof. Zengfu Wang
Advisor:	Prof. Rui Hou
Finished time:	October 28, 2021

中国科学技术大学学位论文原创性声明

本人声明所呈交的学位论文,是本人在导师指导下进行研究工作所取得的成果。除已特别加以标注和致谢的地方外,论文中不包含任何他人已经发表或撰写过的研究成果。与我一同工作的同志对本研究所做的贡献均已在论文中作了明确的说明。

作者签名: _____

签字日期: _____

中国科学技术大学学位论文授权使用声明

作为申请学位的条件之一,学位论文著作权拥有者授权中国科学技术大学拥有学位论文的部分使用权,即:学校有权按有关规定向国家有关部门或机构送交论文的复印件和电子版,允许论文被查阅和借阅,可以将学位论文编入《中国学位论文全文数据库》等有关数据库进行检索,可以采用影印、缩印或扫描等复制手段保存、汇编学位论文。本人提交的电子文档的内容和纸质论文的内容相一致。

控阅的学位论文在解除后也遵守此规定。

☐公开 ☐控阅 (____年)

作者签名: _____

导师签名: _____

签字日期: _____

签字日期: _____

摘 要

吸烟对人体健康造成的危害极大，并且存在着诱发火灾、爆炸等重大安全事故的隐患，尤其是工厂这类易燃易爆场所更是要做到全面禁烟。传统的工厂禁烟方法是设置安全检查人员不断地巡视车间或者通过监控系统来及时发现并制止吸烟行为，存在着效率低下、人工成本高的问题。随着计算机视觉技术的迅速发展，基于深度学习的吸烟检测算法相较于传统吸烟检测算法更具实用性和高效性。因此借助深度学习技术，设计并实现一套针对工厂生产车间的自动吸烟检测系统有着重要的现实意义。主要工作如下：

（1）提出了针对工厂复杂生产环境的吸烟检测算法。首先从机器学习和深度学习两方面对各种目标检测算法进行分析与比较，并在此基础上提出了一种基于 YOLOv5 的两步吸烟检测算法，通过先识别人体再在人体区域内检测香烟的方式来减少复杂背景对检测的干扰，也变相地增大了香烟在检测画面中的比例，而且分别使用了不同大小的 YOLOv5 模型来对人体和香烟进行检测，实现了检测速度与检测精度的平衡。此外，还通过构建针对性的吸烟检测行为数据集的方式，进一步增强了模型的检测能力。

（2）设计并实现了针对工厂生产车间的自动吸烟检测系统。首先采用面向对象分析的方式对用户的需求进行建模，并在基础上完成系统的概要设计与详细设计，然后使用深度学习框架 PyTorch 构建了吸烟检测模型算法，并将其与系统设计进行融合，最终实现了自动吸烟检测系统的搭建与测试。

测试结果表明，该检测系统可以稳定持续地通过多个摄像头对工厂生产车间进行图像采集，使用设计好的吸烟检测算法自动进行吸烟行为检测，及时发现车间内吸烟行为并发出警报，帮助安全检查人员迅速定位吸烟者所处位置，制止吸烟行为并进行警告或处罚，进而实现工厂内全面禁烟的目的，保障了工作人员的生命财产安全和工厂车间的生产环境安全。

关键词：吸烟检测，深度学习，YOLOv5 网络，工厂车间

ABSTRACT

Smoking does great harm to human health, and has risks causing fire, explosion and other major safety accident. Especially in flammable and explosive places like factories, it is necessary to completely ban smoking. The traditional method of banning smoking in factories is to set up safety inspectors to constantly visit the workshop or through the monitoring system to detect smokers, which is inefficient and has high labor costs. With the rapid development of computer vision technology, smoking detection algorithms based on deep learning have the advantages of low cost and good effect compared with traditional smoking detection algorithms. Therefore, it is of practical importance to design and implementation of an automatic smoking detection system for the factory workshop with the help of deep learning technology. The main tasks as follows:

(1) An algorithm for smoking detection in complex factory environment is proposed. Various object detection algorithms are analyzed and compared from the aspects of machine learning and deep learning, and a two-step smoking detection method based on YOLOv5 is proposed on the basis. By first recognizing the human body and then detecting the cigarette in the human body area, the interference of the complex background is reduced, and the proportion of the cigarette in the detection picture is also increased. In addition, YOLOv5 models of different sizes were used to detect human body and cigarettes, which achieves a balance between detection speed and accuracy. By constructing a targeted smoking detection behavior data set, the detection ability of the model is further enhanced.

(2) The automatic smoking detection system for the workshop was designed and implemented. The user needs are modeled by object-oriented analysis, the summary design and detailed design of the system are completed on the basis, and a smoking detection model are constructed using the deep learning framework PyTorch. By integrating detection model with the system design, the automatic smoking detection system was finally constructed and tested.

The test results show that this detection system can steadily and continuously collect images of the workshop through multiple cameras, and use the designed smoking detection algorithm to detect smoking behavior automatically, which detect smoking behavior in the workshop in time and issue an alarm. By the help of this

detection system, safety inspectors can quickly locate the smoker's location, stop smoking acts and issued warnings or penalties, which achieve the goal of a total ban on smoking in the factory, and ensure the safety of the lives and property of the staff and the production environment of the factory.

Key Words: smoking detection, deep learning, YOLOv5, factory

目 录

摘要.....	I
ABSTRACT	III
第 1 章 绪论.....	1
1.1 系统开发背景.....	1
1.2 国内外吸烟检测研究现状.....	2
1.3 本文的主要工作.....	4
1.4 论文的组织结构.....	5
第 2 章 相关技术分析	7
2.1 基于机器学习的目标检测算法.....	7
2.2 基于深度学习的目标检测算法.....	8
2.2.1 卷积神经网络.....	9
2.2.2 基于候选区域的目标检测算法.....	10
2.2.3 基于回归思想的目标检测算法.....	11
2.3 针对工厂生产环境的吸烟检测算法.....	12
2.3.1 YOLOv5 模型	12
2.3.2 基于 YOLOv5s 的人体识别.....	13
2.3.3 基于 YOLOv5x 的香烟检测.....	13
2.4 数据集的构建.....	15
2.4.1 数据的采集.....	15
2.4.2 数据的处理.....	15
2.5 本章小结.....	18
第 3 章 系统需求分析	19
3.1 需求导出.....	19
3.2 分析建模.....	21
3.3 非功能性需求.....	24
3.4 本章小结.....	24
第 4 章 系统概要设计	25
4.1 系统概述.....	25

4.2 系统静态结构.....	25
4.3 系统动态结构.....	27
4.4 系统数据管理.....	28
4.5 本章小结.....	30
第5章 系统详细设计与实现	31
5.1 预处理模块的实现.....	31
5.2 人体识别模块的实现.....	32
5.3 吸烟检测模块的实现.....	33
5.4 画面显示模块的实现.....	36
5.5 数据查询模块的实现.....	38
5.6 本章小结.....	39
第6章 系统测试与分析	40
6.1 测试概要.....	40
6.1.1 测试环境.....	40
6.1.2 概述.....	41
6.2 测试分析.....	41
6.2.1 模块测试需求获取.....	41
6.2.2 配置项测试需求获取.....	42
6.2.3 系统测试需求获取.....	42
6.3 测试用例设计.....	42
6.3.1 模块测试用例设计.....	42
6.3.2 配置项测试用例设计.....	45
6.3.3 系统测试用例设计.....	45
6.4 测试结果及分析.....	47
6.5 测试结论.....	48
第7章 结论与展望	50
参考文献.....	52
致谢.....	56

第 1 章 绪 论

1.1 系统开发背景

吸烟对人体健康造成的危害是众所周知的，尤其是二手烟的危害极大。2017 年，世界卫生组织报告指出：烟草相关疾病是全世界面临的最大公共卫生威胁之一，大约每 6 秒钟就有 1 人死于烟草所致疾病，每年共有近 600 万人死于烟草相关疾病^[1]。香烟在燃吸过程中会产生两部分烟气，其中被吸烟者直接吸入体内的主烟流仅占整个烟气的 10%，而 90% 的侧烟流则弥散在空气中，如果在室内进行吸烟，则会造成室内空气的严重污染。有报道指出，在无人吸烟的室内可吸入颗粒物浓度为 24.4 微克 / 米，而在有 1 人吸烟时，即可增加到 36.5 微克 / 米，而在有 2 人吸烟时，则可增加到 70.4 微克 / 米；而同期室外环境中可吸入颗粒物浓度仅为 1.1 微克 / 米。室内由吸烟引起的颗粒物浓度增高，必使得环境中有害物质的浓度相应增高，并且烟中的有害物质在主烟流和侧烟流中并不是均匀分布的，在主烟流和侧烟流中许多化学物质含量不同，而侧烟流中许多有害物质浓度远高于主烟流。

此外，吸烟也存在着诱发火灾、爆炸等一系列重大安全事故的潜在风险。香烟从点着到吸完大约需要六至十分钟，在这个过程中人可能会不自觉地安全区域走到了危险区域。同时随手乱扔烟头往往是一些吸烟者难以改正的坏习惯，而工厂车间内电风扇或者其他设备运行时产生的风很可能将未熄灭的烟头吹到易燃易爆物品上去，并且需要三十到四十分钟才能产生明显的火苗，如果恰逢下班时间或者无人发现，火情会迅速扩散，难以控制，还可能进一步引发爆炸等特大安全事故。根据国外新闻统计报道，大约有百分之二十的火灾都是因为乱扔烟头导致的，而据我国公安部统计，全国每年的火灾中有 10% 以上系吸烟引发，并且在工业生产环境中由吸烟引起的爆炸也是时有发生，造成了大量的财产损失和人员伤亡。

现在全世界越来越多的国家开始明令禁止在室内公共场所吸烟。自 2003 年我国签署了世界卫生组织《烟草控制框架公约》以来，全国 20 多个城市和地区都已制定或修改了地方控烟法规，力求打造“无烟城市”。2014 年，我国国家卫生和计划生育委员会起草了《公共场所控制吸烟条例（送审稿）》，规定所有室内公共场所一律禁止吸烟^[2]。但是目前缺乏行之有效的禁烟措施，一般都是通过人力监督来对吸烟者进行警告或处罚，这就需要监督人员不断地进行巡查监督，而且不能保障及时地发现并制止每一个吸烟行为。

传统的吸烟检测方法通常是使用烟雾传感器或可穿戴设备等物理方式进行检测^[3]。但是，由于香烟烟雾浓度低、易发散的特点，烟雾传感器的检测效果往往取决于使用环境空间的大小和密闭程度，不适合工厂车间这样的大型环境，而可穿戴设备进行吸烟检测的成本较高，且需要人人佩戴，也不适合有着大量工作人员的工厂使用。因此，设计一套成本低、效率高的吸烟检测系统是十分有必要的。

随着监控系统的日益完善和计算机视觉的不断发展，利用监控摄像头采集的视频图像信息，然后根据预先设计的算法提取图像的特征信息，并以此对视频中的序列图像进行吸烟行为检测的方法越来越受到广大研究人员的重视，也更具实用性和高效性。本文正是基于此思路设计并实现了一套基于深度学习的自动吸烟检测系统，可以及时发现并警告工厂生产车间内发生的吸烟行为，保障了工业生产环境的安全。

1.2 国内外吸烟检测研究现状

目前，国内外常见的吸烟检测方法主要有基于传感器和基于视频图像两种。基于传感器的检测方法通常使用烟雾传感器来检测空气中的烟雾浓度^[4]，当吸烟所产生的烟雾浓度超过了某个设定的阈值就会通过触发蜂鸣器等报警设备发出警告。烟雾传感器根据其原理可分为离子式烟雾传感器和光电式烟雾传感器，离子式烟雾传感器利用烟雾引起烟敏电阻的电压变化进行检测，而光电式烟雾传感器则是通过发光器件和受光器件将烟雾信号转化为电信号，进而检测出空气中烟雾浓度。然而，基于烟雾传感器的检测方法存在着诸多局限：一是吸烟产生的烟雾浓度低，且容易飘散，所以需要高精度的传感器，特别在是较大的生产环境中；二是空气中除香烟烟雾外，还漂浮着许多其他颗粒物，会对传感器的检测产生干扰，因此容易出现误检^[5]。除此之外，基于烟雾检测器的检测方法只能检测区域内是否有人正在吸烟，无法准确地定位吸烟者的位置。

随着可穿戴式检测设备应用的发展，Lopez-Meyer 等人^[6]提出了一种可穿戴射频接近传感器(RF proximity sensor)系统来对吸烟行为进行检测，该传感器对手势具有高敏感性，通过识别在嘴部附近手的姿态来检测吸烟行为，而 Senyurek V 等人^[7]则通过判断肢体多个部位的运动轨迹和速度，与吸烟的动作行为进行模式匹配，进而通过支持向量机(Support Vector Machine, SVM)^[8]等机器学习分类方法对匹配度进行吸烟行为判定。但是基于可穿戴传感器设备的吸烟检测方法成本高且仅能对穿戴设备的人进行监测，因此也没有得到广泛的应用。

近年来，由于监控系统软硬件的不断升级和计算机视觉的飞速发展，利用监

控设备获取视频图像信息,通过提取特定的特征信息,实现图像分类、目标检测等任务的相关研究和技术成果越来越成熟,逐步地代替了传统的人工检测和传感器检测,更具灵活性和高效性。目前基于计算机视觉的吸烟检测方法主要分为识别香烟烟雾、识别吸烟手势和识别香烟三种。

在香烟烟雾检测方面,丁宏杰^[9]使用背景差分法和颜色模型分割可能的香烟烟雾区域,再通过烟雾图元分块和时空特征提取烟雾特征,最后基于 SVM 进行香烟烟雾识别。潘广贞等人^[10]基于混合高斯模型对视频背景进行建模,并结合帧差法提取视频帧中的运动目标,最后在提取出的感兴趣区域中以香烟烟雾的颜色为依据完成吸烟行为的判定。刘远丁^[11]通过提取香烟烟雾的方向梯度直方图(Histogram of Oriented Gradients, HOG)特征以及纹理的特征的方式,进行相关分析后再结合特征融合算法对提取到的特征信息进行分类,进而实现对吸烟行为的检测。I. Kopilovic^[12]则利用小波变换等手段锁定香烟烟雾作为目标,再根据香烟烟雾的飘散方式进行特征分类最终实现对吸烟行为的判定。而叶果等人^[13]提出了一种利用尺度不变特征变换(Scale-Invariant Feature Transform, SIFI)和时空兴趣点(Space-Time Interest Points, STIP)信息进行优化以获取最优分类器的方法。上述香烟烟雾检测算法在狭小的室内场景下均取得不错的效果,但是在开放环境,尤其工厂车间这种大型环境下,检测效果就会变得很差。一方面是因为香烟烟雾浓度低易扩散,工厂背景多为白色不易区分,且光线情况复杂多变容易产生干扰等;另一方面,工厂车间的生产设备在运作时也可能产生烟雾,给香烟烟雾检测带来极大的困难。因此,检测香烟烟雾的方法的不适合工厂车间的环境。

在吸烟手势检测方面, Echebarria 等人^[14]利用人脸识别算法定位嘴部区域,通过在 HSV(Hue, Saturation, Value)颜色空间进行颜色分割和噪声过滤后确定香烟位置,最后根据嘴巴和香烟之间的距离判定是否在吸烟。王超^[15]则根据视频的运动信息结合肤色检测分割出手部区域,再利用 Hu 不变矩和边缘方向直方图提取手势特征,结合 SVM 进行训练和分类,完成对吸烟行为的判定。Bhandari Babin 等人^[16]基于加速度传感器去分析摸头、喝水、吃东西、使用手机以及吸烟等动作所产生的传感器信号,找到吸烟动作与其它动作的不同之处,最终设计了一套非侵入性、自动化的吸烟行为检测框架。林镇涛^[17]基于传统的图像分析处理方法和计算机视觉原理,设计了一种两步吸烟行为检测算法,并针对嵌入式设备进行优化,实现了对户外的施工人员的中距离吸烟行为检测。Zhang D 等人^[18]使用卷积神经网络(Convolutional Neural Network, CNN)通过大量已标注的数据集进行训练,提取出视频图像中表示吸烟行为的高维特征,对抽象的特征向量进行深度学习,实现检测视频中吸烟行为的目的。上述吸烟手势检测算法通常会由于手势复杂、肤色多样、相机角度等原因无法准确地识别到吸烟手势,并且存在着容易和类似

的手势混淆等问题，因此仅仅使用吸烟手势来判断吸烟行为的方法误判率较大。

在香烟检测方面，Wu Pin 等人^[19]基于图像的颜色直方图提取点燃的香烟及吸烟者的特征信息，并使用高斯混合模型实现了香烟的分割和背景像素跟踪，可以检测各种不同尺度、颜色和性质的香烟。Zhao R 等人^[20]先通过 K-Means 聚类算法在训练集上得出香烟的先验框，再在基础的 YOLOv3-tiny 网络结构上添加了一个细粒度的目标检测层，有效地改善了对细小香烟目标的检测效果。李倩等人^[21]使用人脸识别算法将识别到的人脸图像作为香烟的检测区域，大幅减少了需要检测区域，并且使用了 Faster Region-CNN(R-CNN)模型对香烟进行高精度的检测。Artan Y 等人^[22]提出一种基于近红外(Near Infrared, NIR)监视摄像机检测驾驶员吸烟行为的方法，首先通过目标检测技术定位前挡风玻璃和驾驶员头部区域，然后在这两个区域执行双窗口局部异常检测确定 NIR 图像上的因烟头产生的白色热点，并以此来判定驾驶员是否有吸烟行为。程淑红等人^[23]将多个不同的卷积神经网络算法级联并结合级联回归来准确迅速地定位嘴部感兴趣区域(Region of Interest, ROI)，同时采用残差网络对感兴趣区域内的目标特征进行判别，从而实现对香烟的检测。上述香烟检测算法都由于香烟自身目标太小，缺乏足够多的特征信息将其与类似的小物体进行区分，因此检测精度都不是很理想。

纵观上述吸烟检测研究现状，通过基于深度学习的目标检测技术对不同目标进行吸烟检测存在着香烟烟雾浓度低易扩散、吸烟手势复杂和香烟自身太小等问题。本文在综合考虑后选择针对香烟本身进行识别，通过对目标检测算法进行深入研究，提出了一种先识别人再检测烟的两步检测算法，并在此基础上设计出一套针对工厂复杂环境的自动吸烟检测系统。

1.3 本文的主要工作

为了保证工业生产环境的安全，本文从吸烟行为检测的角度入手，设计并实现了一套利用监控设备采集图像并进行自动吸烟行为检测的系统，可以及时发现工厂生产车间内发生的吸烟行为并发出警报，帮助安全检查人员快速定位吸烟者位置。完成的主要工作如下：

(1) 提出了针对工厂车间复杂环境的吸烟检测算法。本文比较并分析了基于机器学习和基于深度学习两方面的各种目标检测算法，阐述了基于机器学习目标检测算法存在的问题，然后针对工厂生产环境复杂，香烟目标太小的问题，提出了一种基于 YOLOv5 的两步吸烟检测算法：第一步快速识别监控画面中的所有人体并进行分割；第二步在分割出的人体区域内进行高精度的香烟检测。该算法在减少检测的目标区域，降低复杂背景对检测的干扰的同时，也变相地提高了

香烟在检测画面中的所占比例，而且分别使用了两种不同大小的 YOLOv5 模型对人体和香烟进行检测，实现了检测速度和精度的平衡。此外，还通过构建针对性的吸烟检测行为数据集的方式，进一步增强了模型的检测能力。

(2) 设计并实现了针对工厂车间的自动吸烟检测系统。本文使用 Python 语言开发了一套 Windows 和 Linux 通用的吸烟检测系统，包括采用面向对象分析的方法对安全检查人员的需求进行建模，进行系统的概要设计和详细设计，并将吸烟检测算法与系统设计进行融合，最终实现了自动吸烟检测系统的搭建与测试。经测试结果表明，本文开发的自动吸烟检测系统能够长时间稳定持续运行，通过多个摄像头对工厂生产车间进行图像采集，使用设计好的吸烟检测算法自动进行吸烟行为检测，及时发现车间内吸烟行为并发出警报，帮助安全检查人员定位吸烟者位置。

本人负责的主要工作有收集吸烟行为数据集并进行标注与数据增强；训练检测人体与检测香烟的两种神经网络模型并进行针对性的优化；实现两步吸烟检测算法并将其与系统设计相结合；测试吸烟检测系统的功能和性能是否满足设计需求。

1.4 论文的组织结构

本论文总共分为 7 章，各章节内容安排如下：

第一章：绪论。本章首先简要介绍了吸烟检测系统的背景和开发意义，然后对国内外吸烟行为检测的两种主要研究方向进行举例分析，进而确定了本文所设计系统的检测思路，最后阐述了本文的主要工作以及论文的组织结构。

第二章：相关技术分析。本章对吸烟检测系统相关的各种技术知识进行分析，包括机器学习、深度学习以及卷积神经网络等相关技术，然后针对工厂生产环境复杂，香烟目标太小的问题，提出了一种先识别人再检测烟的两步吸烟检测算法，并构建了针对性的吸烟行为数据集。

第三章：系统需求分析。本章主要是采用面向对象分析的方法对系统用户的需求进行分析与建模，明确吸烟检测系统的功能需求和性能需求，建立系统需求规格说明书。

第四章：系统概要设计。本章主要是对吸烟检测系统的总体框架进行设计与描述，包括系统的静态软件结构、动态运行流程以及数据存储设计，确立了系统的五个功能模块：预处理模块，人体识别模块，吸烟检测模块，画面显示模块以及数据查询模块。

第五章：系统详细设计与实现。本章主要是对吸烟检测系统的各个功能模块

的内部实现进行设计，确定每个模块的详细算法过程、数据结构和接口细节，并进行具体实现。

第六章：系统测试与结果分析。本章主要是对吸烟检测系统进行单元测试、配置项测试和系统测试，包括测试需求分析和测试用例设计的过程，并依据测试结果，分析吸烟检测系统功能和性能的实现情况。

第七章：结论与期望。本章对全文工作进行总结，简要地回顾了吸烟检测系统的开发背景，介绍了系统的具体设计与实现情况，并指出了当前系统所存在的问题以及未来可能进行改进的方向。

第2章 相关技术分析

从第一章中国内外吸烟检测研究现状可以看出,目前大部分学者们对于吸烟检测的研究都是从目标检测的角度出发,所以本章首先从机器学习和深度学习两个领域比较并分析了各种目标检测算法,指出基于机器学习的目标检测算法所存在的两个问题,然后对深度学习目标检测领域中常用的几种卷积神经网络模型进行了详细说明与分析,最后在此基础上针对工厂复杂生产环境,提出了一种基于YOLOv5的两步吸烟检测算法并构建了针对性的吸烟行为数据集。

2.1 基于机器学习的目标检测算法

目标检测是计算机视觉领域最重要也最具挑战的一个分支,在现实生活中得到了广泛的应用,如监控安全、自动驾驶、无人机场景分析等^[24]任务。目标检测简单来说就是在给定的图片中精确地找出特定的目标,并获取目标的类别信息和位置信息。不同于图像分类任务,目标检测需要对图片的前景和背景有着足够的理解,进而从背景中分离出感兴趣的目标,并确定这一目标的类别和位置信息。

基于机器学习的目标检测一般分为三个步骤:(1)在给定图片上选取候选区域;(2)在候选区域内提取图像特征;(3)设计分类器判定候选区域是否包含目标对象。其基本流程如图2.1所示,具体地说:

(1) 第一步候选区域选择是为了对目标进行大致的定位。由于目标可能出现在图像的任何位置、目标有各种不同的大小以及目标可能有各种不同的形状等原因,这一步多采用滑动窗口的策略对整个图像进行遍历,而且还要设置不同尺度、不同长宽比的矩形框去进行遍历。这种穷举的策略虽然包括了目标所有可能出现的位置,但是缺少针对性,时间复杂度太高,而且会产生冗余窗口,严重影响了后续特征提取和分类器判定的性能。

(2) 第二步特征提取通常采用一些计算机视觉模式识别中的经典特征提取算法,包括:基于颜色、纹理或形状等底层特征的方法,以及一些利用机器学习进行特征挖掘后的中、高层次语义特征,如尺度不变特征(SIFT)^[25]、方向梯度直方图特征(HOG)^[26]。然而由于目标形态的多样性、光照变化的不确定性和背景的复杂性等因素,人为设计的特征对图像场景的认知能力往往是有局限性的,这就会导致最终检测结果的准确性不高,而且缺乏鲁棒性。

(3) 第三步分类器判定是对候选区域内提取出的特征通过一个预先设计好的分类器进行分类判定,常用的分类器有支持向量机(SVM),自适应增强

(Adaboost)^[27]和 Deformable Part Model (DPM)^[28]等, 这些分类器的效果与候选区域所提取的特征密切相关。对于单类别目标检测只需判定候选区域包含的是背景还是目标对象, 而对于多类别目标检测则需要进一步区分所包含对象的类别。此外, 如果经过分类器判定后得到的一系列预测框存在着重叠情况, 还可以通过非极大值抑制算法(Non-Maximum Suppression, NMS)对预测框进行二次筛选, 再将筛选过的预测框作为最终的检测结果。

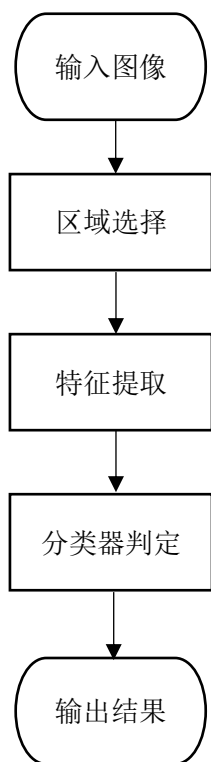


图 2.1 基于机器学习的目标检测基本流程

综上所述, 基于机器学习的目标检测算法存在着以下两个主要问题: 1. 基于滑动窗口的候选区域选择策略时间复杂度高, 而且存在着窗口冗余的情况; 2. 基于手工设计的特征对目标的多样性变化不具有很好的鲁棒性。

2.2 基于深度学习的目标检测算法

现如今, 基于深度学习的目标检测算法逐渐成为了主流研究方法, 主要过程分为图像的深度特征提取和基于深度神经网络的目标识别与定位两个部分, 其中用到深度神经网络模型多为卷积神经网络(CNN)。根据不同的检测思路可将检测算法分为两种类型: 一种是基于候选区域(Region Proposal)的两阶段(Two stage)算法, 检测准确度高但速度较慢; 另一种是基于回归思想的一阶段(One stage)算法, 检测速度快但准确度不高。下面分别对卷积神经网络以及两种类型的经典目

标检测算法进行介绍与分析。

2.2.1 卷积神经网络

卷积神经网络是深度学习领域的经典算法之一，最早由 Yann LeCun^[29]在 1989 年提出并成功应用到手写字符识别系统中。卷积神经网络是一种前馈神经网络，具有局部连接、权重共享和下采样的结构特点，在图像处理领域如图像分类、物体识别等任务上表现优异。

卷积神经网络相比于其他神经网络的优越性主要体现在局部连接与权值共享两个方面。局部连接是指卷积神经网络不像传统神经网络那样，第 $n-1$ 层的每一神经元都与第 n 层的所有神经元连接，而是第 $n-1$ 层的神经元与第 n 层的部分神经元之间进行连接；而权值共享是指同一个卷积核在整张图像中权重参数不变，这使得卷积神经网络的网络结构更加类似于生物的神经网络。以上两个特点降低了卷积神经网络模型的复杂度，减少了权值的数目，更有利于提取图像的深度特征。

卷积神经网络的基本结构由卷积层、池化层和全连接层组成，一般会设置若干个卷积层和池化层，并采用卷积层和池化层交替的方式提取图像特征，最后通过全连接层进行输出。随着卷积神经网络的不断发展，出现了残差模块、空洞卷积模块和深度可分离卷积模块等复杂结构。下面对基础的卷积神经网络结构进行介绍：

(1) 卷积层：主要用于图像特征的提取，是卷积神经网络的核心，产生了网络中大部分的计算量。在一个卷积层中通常包含多个可学习的卷积核，每个卷积核对应一组权重参数被整个输入共享，不同的卷积核对应不同的特征提取，通过反向传播算法可以对每个卷积核参数进行优化。卷积层的计算过程为将上一层输出的特征图(Feature map)与卷积核进行卷积操作，然后使用激活函数对结果进行非线性激活以增强特征的非线性拟合能力，最终得到输出特征图。一次卷积计算过程如图 2.2 所示。

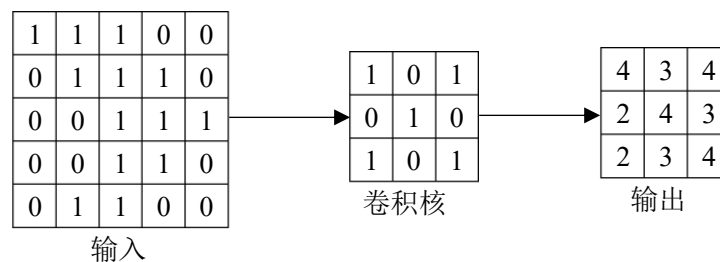


图 2.2 卷积计算过程

(2) 池化层：也称下采样层，通常出现在连续卷积层的中间，对提取出的特征进行抽样，降低了特征的维度，从而减少训练参数，减轻了网络过拟合程度。池化层缩小了特征图的大小但不改变特征图的数量。常用的池化操作有最大值池化和平均值池化^[30]。最大值池化是选择特征图上某一区域的最大值作为该区域的特征，能够减少卷积层参数误差造成估计均值误差的偏移，保留了更多的纹理信息；平均值池化是计算特征图上某一区域的平均值作为该区域的特征，可以减少邻域大小受限造成的估计值方差，保留更多的图像背景信息。最大值池化和平均值池化过程如图 2.3 所示。

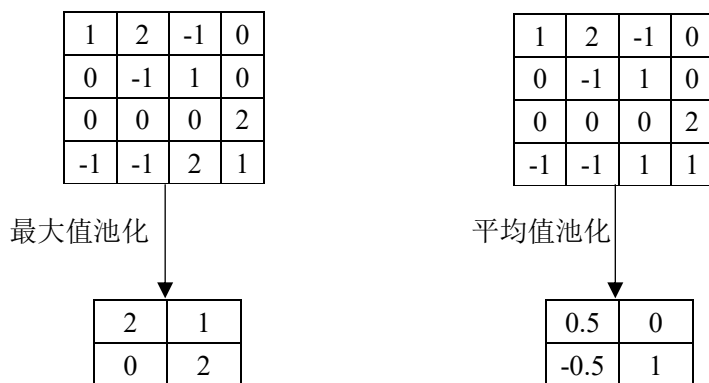


图 2.3 两种池化计算过程

(3) 全连接层：与常规神经网络中神经元一样，通常位于网络的最后一部分，将前面的卷积层和池化层所提取的各种深度特征进行融合，增强了信息表达能力。

2.2.2 基于候选区域的目标检测算法

为了解决基于机器学习的目标检测算法存在的两个主要问题，Ross B. Girshick 等人^[31]使用 Region Proposal + CNN 结构替代传统目标检测所使用的滑动窗口和手工设计特征，设计出了 R-CNN 网络模型，其主要由三个部分组成：

(1) 利用选择性搜索算法(Selective Search)^[32]在给定图像中提取大约 2000 个的候选区域(Region Proposal)；(2) 将每个候选区域缩放至 227×227 并输入到当时最好的分类网络 AlexNet^[33]中提取特征；(3) 将每个候选区域特征输入到 SVM 进行分类。R-CNN 将目标检测任务转化为候选区域上的分类任务，利用选择性搜索算法提取候选区域可以在选取较少区域的情况下保持较高的召回率，大大降低了后续步骤的时间复杂度，并且获取的候选区域也比通过滑动窗口选取的质量更好。此外，从图像分类的结果来看，卷积神经网络自动习得的特征比传统目标检测中手工设计的特征更具鲁棒性、效果更好，因此 R-CNN 在局部候选区域内选用 AlexNet 作为骨干网络(Backbone)进行特征提取，以发挥卷积神经网络强大

的学习能力。

随后, Ross B. Girshick 又针对 R-CNN 每个候选区域的骨干网络计算不共享导致大量耗时的问题进行了改进, 提出 Fast R-CNN 网络模型^[34]。Fast R-CNN 将整个图像先经过骨干网络提取特征得到特征图, 然后在原图上运行选择性搜索算法并将提取到的候选区域映射到特征图上, 再通过 ROI 池化层把不同大小的候选区域转化为固定大小的特征向量, 最后将这些特征向量输入用于分类和边界回归的全连接层中得到最终结果。Fast R-CNN 只需对整个图像进行一次特征提取, 并且可以一次性全部送入后续网络进行计算处理, 与 R-CNN 相比节省了大量用于候选区域特征提取的时间, 也节约了存储不同大小特征所需的磁盘存储空间。此外, 还使用 Softmax 代替 SVM 进行分类任务, 省去了 SVM 的训练过程。R-CNN 的训练过程是多阶段的, 包括预训练、微调、SVM 分类和边界框回归四个训练阶段, 而 Fast R-CNN 是一个单阶段的端到端训练过程, 使用多任务损失函数来训练联合网络。

之后, Ren Shaoqing^[35] 等人在 Fast R-CNN 的基础上继续做出改进, 巧妙地设计了一个卷积神经网络 Region Proposal Networks(RPN)来代替原有的选择性搜索算法提取候选区域, 提出了 Faster R-CNN 网络模型。RPN 网络使用骨干网络提取的卷积特征图作为输入, 在最后一个卷积层的输出结果上通过基于多尺度锚框(Anchor)的滑动窗口过程产生多个候选区域并且细化边界框的位置, 从而快速地产生具有大范围尺度和宽高比的高质量候选区域。Faster R-CNN 成功地将一直以来分离的候选区域提取和整个检测网络融合到了一起, 真正地实现了端到端的目标检测网络框架, 在检测速度和精度上都得到了很大的提升。

2.2.3 基于回归思想的目标检测算法

从 R-CNN 到 Fast R-CNN 再到 Faster R-CNN, 基于深度学习的目标检测算法的精度越来越高, 速度也越来越快, 但是仍然不能满足实时性的要求。于是, Joseph Redmon 等人^[36]提出了基于回归思想的目标检测网络 YOLO, 可以在给定图像上直接回归出特定目标的边界框和类别信息, 省去了生成候选区域的过程, 大大加快了检测速度。YOLO 的检测流程主要包括三步: 1. 将输入图像划分成 $S \times S$ 个网格, 若某个物体的中心坐标在某个网格, 则分配该网格负责预测这个物体; 2. 对于每个网格, 通过基于 GoogLeNet^[37]的网络预测 B 个边界框以及对每个类别预测一个概率, 每个边界框包括四维位置信息(中心点坐标和宽高)以及一维置信度信息(反映该框包含物体的概率和位置的准确性); 3. 根据阈值去除可能性较低的预测框, 再使用非极大值抑制算法对重复的预测框进行过滤后得到最终检测结果。YOLO 使用了提取出的整个图像特征进行预测, 充分地利用了上

下文信息,使得在背景上预测错误目标的概率大大降低。Liu W. 等人^[38]则结合 YOLO 的回归思想和 Faster R-CNN 的锚框机制设计了 SSD 网络模型。SSD 在 VGG-16 网络^[39]后面添加了一些额外的卷积层,这些卷积层的尺寸逐渐减少从而实现在多个尺度特征上进行预测,同时还使用了多个不同尺寸、不同长宽比的锚框,大大地提高了检测精度。而且不同于 YOLO 最后的全连接层,SSD 使用卷积层进行最后的预测,使得检测网络可以适应不同大小的输入,更具有灵活性。

随后,Joseph Redmon 等人对 YOLO 不断地进行改进,设计了 YOLOv2^[40]和 YOLOv3^[41]网络模型。YOLOv2 将骨干网络由 Darknet-17 升级为 Darknet-19,并采用批标准化(Batch Normalization)^[42]、锚框机制、K-means 聚类算法、细粒度特征(Fine-Grained Features)和多尺度训练等方法优化检测效果。YOLOv3 在此基础上使用参考残差网络(ResNet)^[43]设计的 Darknet-53 作为骨干网络,并使用 Feature Pyramid Networks(FPN)^[44]结构进行多尺度预测和交叉熵(Cross-Entropy)损失等技术最终实现了更高的检测精度和更快的处理速度。

后来,Alexey Bochkovskiy 等人^[45]在综合各种前人研究技术并加以适当创新后提出了 YOLOv4 网络模型,实现了检测速度和检测精度的完美平衡。YOLOv4 的改进主要有:1. Mosaic 数据增强,把四张图通过随机缩放、随机剪裁、随机排布的拼接成一张图,变相地增大了最小批(Mini-Batch),更有利于小目标的检测;2. 自对抗训练,训练时让网络模型反向更新图像,在图像上产生扰动影像,然后在更新后的图像上进行训练;3. 引入 Path Aggregation Network(PANet)^[46]的 PAN 结构,并将两个特征图的融合从相加(addition)改为拼接(concatenation);4. 跨最小批标准化,是在跨批标准化(Cross Batch Normal)^[47]的基础上进行的改进;5. Mish 激活函数,DropBlock 正则化,基于 CIoU 的损失函数等技术。

2.3 针对工厂生产环境的吸烟检测算法

考虑到工厂复杂多变的生产环境以及香烟本身目标太小的问题,本文使用最新的 YOLOv5 建立了吸烟检测模型,并设计一种先识别人体再检测香烟的两步检测算法,通过将香烟检测的范围缩小到识别出的人体所在区域来减少复杂背景对检测的干扰,同时也变相地增大了香烟在检测画面中所占地比例。下面对 YOLOv5 模型以及在此基础上设计的吸烟检测算法进行详细说明。

2.3.1 YOLOv5 模型

YOLOv5 是 Ultralytics 团队在 YOLOv4 的基础继续改进并使用 PyTorch 框架实现的,主要改进有:1.使用 Focus 结构对输入图像进行切片,在图像中每隔一

个像素取一个值，将一张图片切分成四张小图片，然后将切分后的图片经过卷积操作即可得到没有信息损失的二倍下采样特征图；2.借鉴 Cross Stage Partial Network(CSPNet)^[48]的思想在骨干网络和特征融合网络中添加 CSP 结构，将基础层的特征映射分为两个部分，然后通过跨阶段层次结构将它们合并，有效地缓解了梯度消失的问题，在保证检测速度和检测精度的同时减少了模型的大小；3.使用 SPP-Net^[49]中的 SPP 结构，将通过多个不同尺度的最大池化操作后得到的特征图拼接起来再输入后续网络进行计算处理，扩大了网络的感受野，实现了局部特征和全局特征的融合；4.通过 FPN+PAN 结构来构建特征金字塔，FPN 把深层的语义特征传递到浅层，增强了特征金字塔的语义表达，而 PAN 将浅层的位置特征传递到深层，增强了特征金字塔的定位能力，实现了多尺度特征的深度融合，最后在三个不同尺度的特征图上使用 YOLOv3 检测头进行预测。

YOLOv5 通过控制网络层数和卷积核数量设计了四种不同大小的网络结构，从小到大依次为 YOLOv5s, YOLOv5m, YOLOv5l, YOLOv5x, 可以灵活地运用于各种不同的目标检测任务。YOLOv5l 是 YOLOv5 的基线网络，其网络结构如图 2.4 所示。

2.3.2 基于 YOLOv5s 的人体识别

针对工厂生产环境复杂多变的问题，本文选择先进行人体识别将检测区域缩小至人体区域内。YOLOv5s 是 YOLOv5 系列中网络最浅、宽度最小的模型，仅有 YOLOv5l 一半的卷积核数量和三分之一的网络层数，检测速度最快，但精度较低，因此使用它来完成人体识别这种较大目标的检测任务是最合适不过的。工厂监控设备所拍摄的图像大小一般为 1920×1080 ，直接输入网络进行检测的话比较费时，因此先将输入图像等比例地缩放至 640×360 大小，又因为 YOLOv5 的骨干网络有五次下采样操作，故需将图像填充至 640×384 大小后才能输入网络进行计算。检测网络最终输出的结果为人体检测框的中心点坐标和宽高，根据图像变化关系将其转换成 1920×1080 的原始图像上的坐标后在原图上进行人体切割，即可得后续吸烟检测的候选区域。

2.3.3 基于 YOLOv5x 的香烟检测

针对香烟本身目标太小的问题，本文选择使用 YOLOv5 系列中网络最深、宽度最大的模型 YOLOv5x 来保证检测结果的准确性，YOLOv5x 拥有 YOLOv5l 模型 1.25 倍的卷积核数量和 1.33 倍的网络层数，检测能力更强但速度较慢。此外，在检测图像上切割出的人体区域通常是大小不一的，为了加快检测速度，设

计了一种可以自适应填充最小灰边的算法来统一人体区域图像的大小。若给定图像大小为 400×300 ，目标图像大小为 320×320 ，则自适应填充算法步骤如下：第一步计算图像宽高缩放比例， $320 \div 400 = 0.8$ ， $320 \div 300 = 1.07$ ；第二步选择最小的缩放比例 0.8 进行缩放， $400 \times 0.8 = 320$ ， $300 \times 0.8 = 240$ ；第三步计算最小灰边填充大小， $320 - 240 = 80$ ， $80 \bmod 32 = 16$ ，故在图像较短边的两侧各填 8 个像素的灰边至 320×256 的大小。将填充后的图像输入检测网络后得到了香烟检测框的中心点坐标和宽高，根据所输入人体区域的坐标和图像变化关系将其转换成监控画面上的坐标后进行显示。

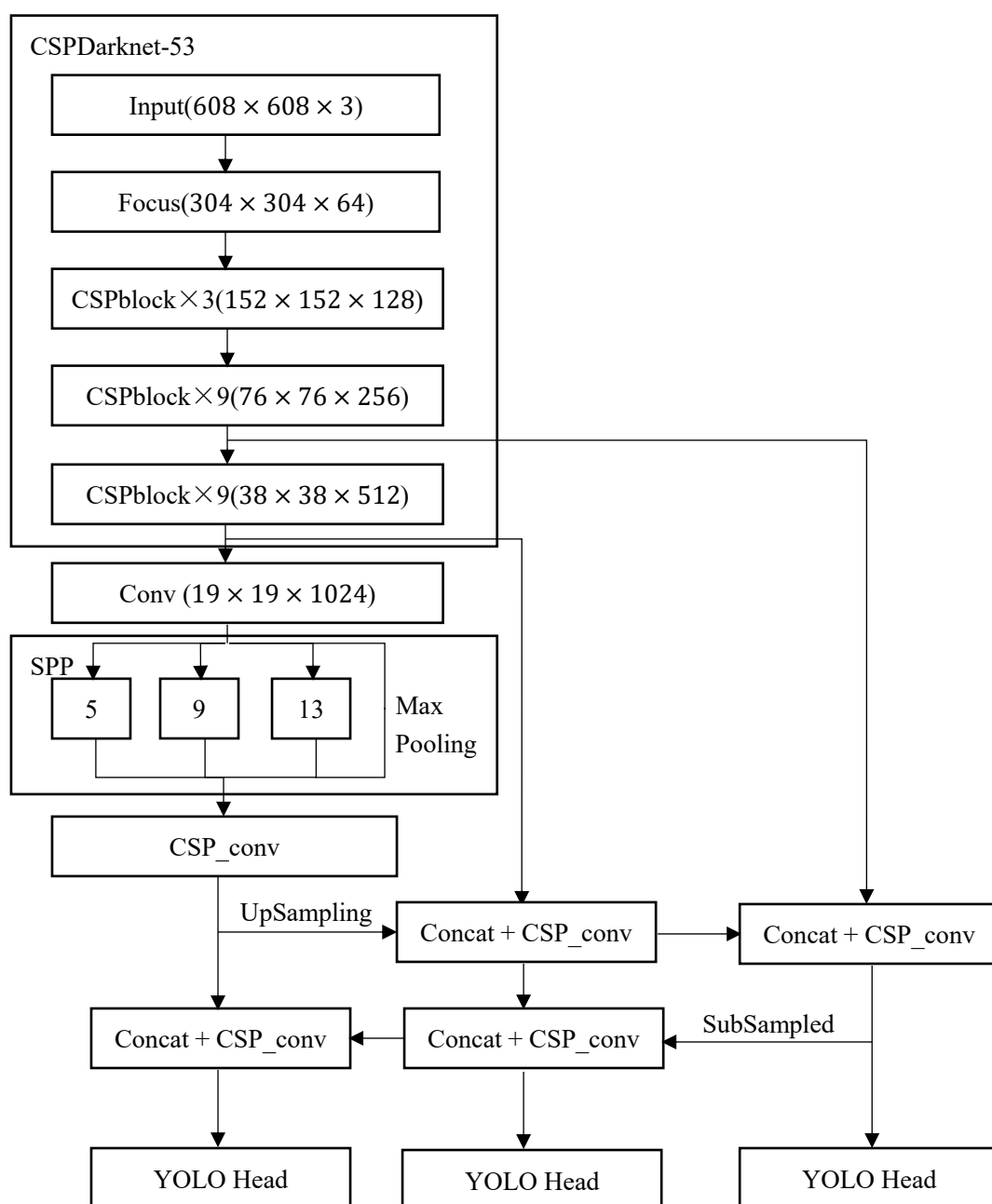


图 2.4 YOLOv5l 网络结构

2.4 数据集的构建

目前针对吸烟行为检测的研究还不太成熟,从不同角度入手的吸烟行为检测方法所用的数据集都不太一样,因此缺少有关吸烟行为的标准公开数据集,所以本文自行收集并构建了针对本项目的吸烟行为数据集,用于吸烟检测模型的训练和测试。

2.4.1 数据的采集

通过采集谷歌和百度图片上吸烟者图像、截取影视图像中吸烟者和自行拍摄针对性图像等手段收集 12400 张图像数据,包括吸烟者图像、手拿类烟物体图像以及正常人体图像。数据集的正负样本比按照统计出的中国 30%的吸烟率来设计,即吸烟与非吸烟图像比例为 3:7。由于图像获取的方式不同,图像的大小和质量差距都较大,所以对图像数据进行了筛选,去除太小和太模糊的图像,然后通过比例缩放和填充灰边的方法将剩余图像统一至 640×640 大小,最后按照 8:1:1 的比例随机分成训练集、验证集和测试集。训练集是输入到检测网络进行训练的数据,验证集是用来在训练过程中不断地对当前训练的模型进行测试验证训练效果,测试集是用来在训练结束后对训练好的检测网络进行测试评价的。

2.4.2 数据的处理

数据集的质量对深度学习模型训练的影响极大,所以要保证数据集的图像尽量清晰,标签尽量准确。本文使用 LabelImg 软件对图像数据进行手动标注来获取目标物体的具体位置,标注内容为人体和香烟两个目标,标签内容为五元组(class, x, y, w, h),其中 class 表示目标类别,0 代表人体,1 代表香烟,(x, y)表示目标物体的中心点坐标,(w, h)表示标注框的高度和宽度,并使用原始图像的宽高对 x、y、w、h 进行归一化,方便后续的操作与计算。图像的 txt 标签文件如图 2.5 所示,对应的标注界面如图 2.6 所示。

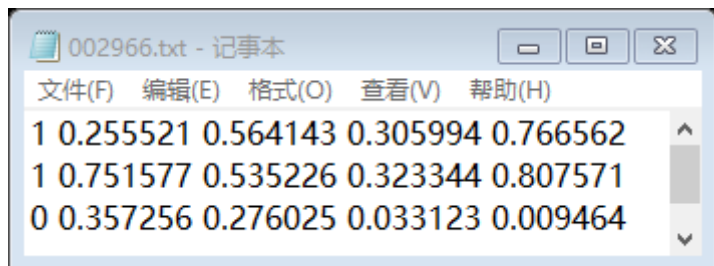


图 2.5 标签文件示例

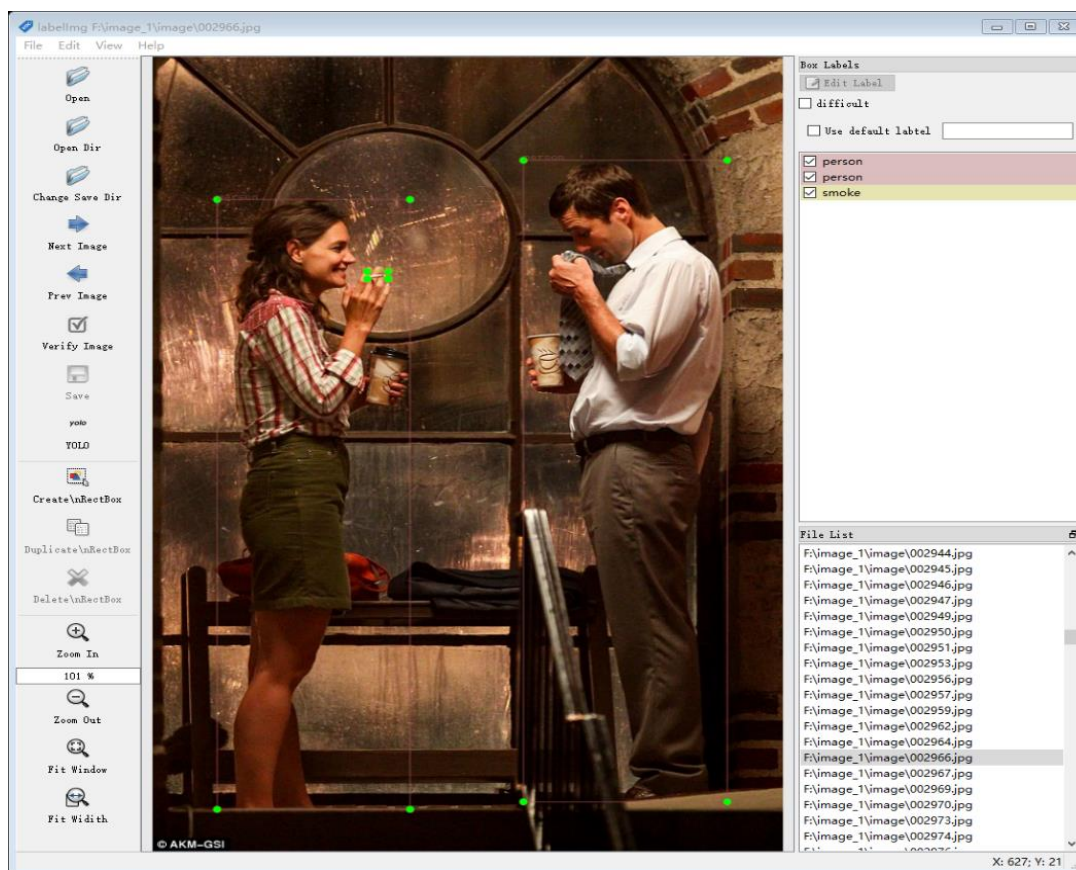


图 2.6 标注界面

经过筛选后的图像数据远远不足以满足深度学习模型的训练，因此通过数据增强的方法来增加训练集图片的数量，主要有 1.构造 2×3 变换矩阵然后使用 OpenCV 的 `warpAffine` 函数对图像进行旋转和平移；2.将图像转化为 HSV 颜色格式，然后对颜色、饱和度和亮度进行随机变换；3.使用 Numpy 的 `flipplr` 函数将图像进行左右翻转。上述数据增强方式都是随机进行的。最后使用 Mosaic 数据增强方法进行图片拼接，先随机选取画面中的一点作为中心点，然后将四张图片分别拼到中心点的左上、右上、左下和右下四个方位，根据计算得到的拼接坐标可以得出每个图像的偏移量，然后对每个图像的标签坐标进行转换，得到拼接后图像的标签。标签转换函数如图 2.7 所示，Mosaic 图像拼接过程如图 2.8 所示，最终得到的一个批次中训练所用数据如图 2.9 所示。

```
def xywhn2xyxy(x, w=640, h=640, padw=0, padh=0):
    # 将原始标签[x, y, w, h]转换为拼接后的标签[x1, y1, x2, y2]
    y = x.clone() if isinstance(x, torch.Tensor) else np.copy(x)
    y[:, 0] = w * (x[:, 0] - x[:, 2] / 2) + padw # 左上角 x
    y[:, 1] = h * (x[:, 1] - x[:, 3] / 2) + padh # 左上角 y
    y[:, 2] = w * (x[:, 0] + x[:, 2] / 2) + padw # 右下角 x
    y[:, 3] = h * (x[:, 1] + x[:, 3] / 2) + padh # 右下角 y
    return y
```

图 2.7 标签转换函数

```

def load_mosaic(self, index):
    labels4 = []
    s = self.img_size
    # 随机的中心点坐标
    yc, xc = [int(random.uniform(-x, 2 * s + x)) for x in self.mosaic_border]
    # 另外3张图片的索引
    indices = [index] + random.choices(self.indices, k=3)
    for i, index in enumerate(indices):
        # 加载图像
        img, _, (h, w) = load_image(self, index)
        # 拼接四张图片
        if i == 0: # 左上
            # 用114像素初始化拼接图像，宽高均为待拼接图像的二倍
            img4 = np.full((s * 2, s * 2, img.shape[2]), 114, dtype=np.uint8)
            # xmin, ymin, xmax, ymax (拼接图像)
            x1a, y1a, x2a, y2a = max(xc - w, 0), max(yc - h, 0), xc, yc
            # xmin, ymin, xmax, ymax (待拼接图像)
            x1b, y1b, x2b, y2b = w - (x2a - x1a), h - (y2a - y1a), w, h
        elif i == 1: # 右上
            x1a, y1a, x2a, y2a = xc, max(yc - h, 0), min(xc + w, s * 2), yc
            x1b, y1b, x2b, y2b = 0, h - (y2a - y1a), min(w, x2a - x1a), h
        elif i == 2: # 左下
            x1a, y1a, x2a, y2a = max(xc - w, 0), yc, xc, min(s * 2, yc + h)
            x1b, y1b, x2b, y2b = w - (x2a - x1a), 0, w, min(y2a - y1a, h)
        elif i == 3: # 右下
            x1a, y1a, x2a, y2a = xc, yc, min(xc + w, s * 2), min(s * 2, yc + h)
            x1b, y1b, x2b, y2b = 0, 0, min(w, x2a - x1a), min(y2a - y1a, h)
        img4[y1a:y2a, x1a:x2a] = img[y1b:y2b, x1b:x2b] # img[ymin:ymax, xmin:xmax]
        padw = x1a - x1b
        padh = y1a - y1b

```

图 2.8 Mosaic 拼接代码

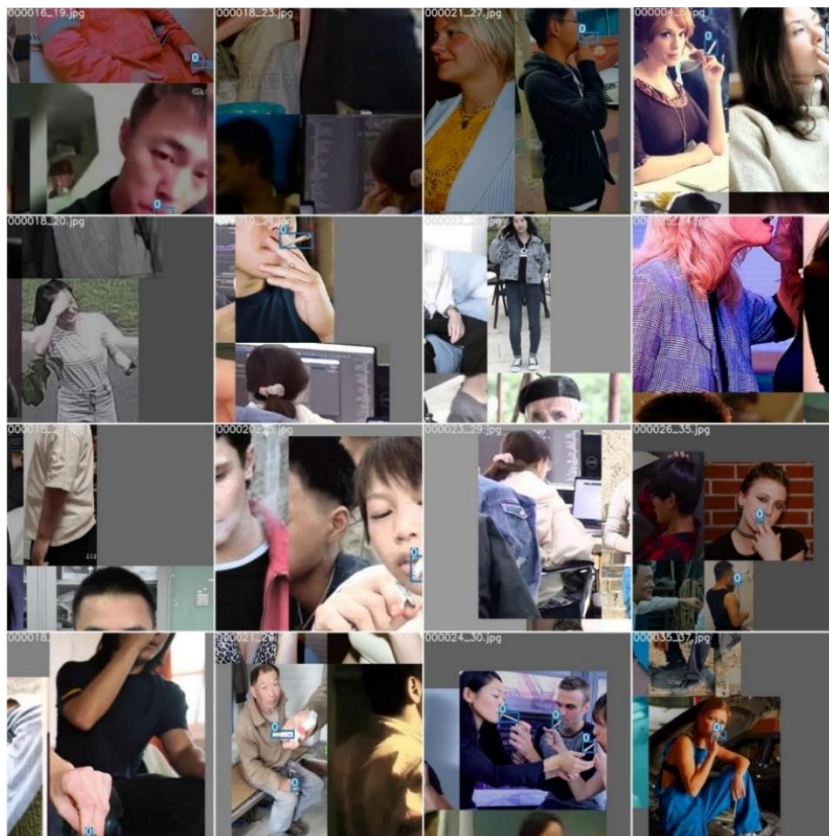


图 2.9 训练数据

2.5 本章小结

本章节从基于机器学习和基于深度学习两方面深入地对各种目标检测算法进行了比较与分析,阐述了基于机器学习的目标检测所存在的问题,并详细分析了几种基于深度学习目标检测的常用网络模型,最后在此基础上针对工厂复杂生产环境设计了一种基于 YOLOv5 的两步吸烟检测算法,分别使用两种不同大小的 YOLOv5 模型对人体和香烟进行检测,可以快速定位人体进而准确地识别出香烟,并且构建了检测模型训练所需要的针对性吸烟行为数据集。

第3章 系统需求分析

需求分析是软件生产周期中的一个重要环节,本章将采用面向对象分析的方法对吸烟检测系统的需求进行具体分析与建模,明确系统所需实现的功能需求以及所需满足的非功能性需求。

3.1 需求导出

工业是国民经济中起着主导作用的物质生产部门,为自身和国民经济其它各个部门提供主要生产资料,也为人民物质文化生活提供工业消费品。它决定着国民经济现代化的速度、规模和水平,是国家财政收入的主要源泉,更是国家经济自主、政治独立、国防现代化的根本保证。此外,工业的发展还是巩固社会主义制度的物质基础,是逐步消除工农差距和城乡差别,推动社会主义社会向共产主义社会过度的前提条件。工业生产活动基本是在工厂生产车间内进行的,因此保障工厂车间的生产环境安全,保证生产活动的有序进行是工业持续发展的基础条件。

工厂车间具有着复杂独特的室内环境,不仅拥有大量的生产资料和机械设备,还存在着大量的工作人员操作设备进行工业生产活动,属于易燃易爆的高风险区域。所以为了保障工作人员的生命安全和工厂的财产安全,全面禁烟是工厂车间的一项基本要求。传统的禁烟方法一般是设置安全检查人员,通过查看监控设备或者不断巡查来发现吸烟者,及时制止吸烟行为并进行警告或处罚。这种方法需要安全检查人员长时间枯燥地盯着多个监控画面或者安排足够多的人员去巡查才能实现禁烟的目的,并且吸烟往往是一个短暂的行为过程,安全检查人员很难及时地发现吸烟者,禁烟效率低下且存在一定的安全隐患。

吸烟检测系统是针对工厂禁烟工作效率低下,人力成本过高的问题设计的一套完整的解决方案,借助监控设备和计算机视觉技术,由传统的人工巡查变成计算机自动检测,也不再需要安全检查人员长时间枯燥地盯着监控画面。该系统不仅提高了检测效率,还减低人力成本,给各种禁烟工作提供了一套行之有效的解决思路。为了达到预期的检测效果,本系统使用了基于 YOLOv5 的两步检测算法,不仅减少了漏报以确保每一次吸烟行为都可以被检测到,保障了工厂车间的安全生产环境,而且避免出现大量的误报而导致保存大量的无效的检测结果,造成糟糕的用户体验。

通过与工厂生产人员和安全检查人员的沟通,以及对市面上的吸烟检测系统

的调研，最终明确了工厂禁烟工作的业务需求和检测流程，并在基础上进行了进一步分析，建立了系统需求模型。

为了保障检测系统的完整性，设计时还应该考虑以下实际问题：

（1）工厂生产车间的室内面积通常很大，远远超过单个摄像头所能监控的范围，一般需要多个摄像头才能覆盖整个生产车间，因此检测系统需要同时接入多个摄像头数据流并行地进行检测任务。

（2）吸烟行为一般持续时间为 5-7 分钟，在这个过程中吸烟者可能走到监控摄像头死角或者易燃易爆的危险区域，因此检测系统需要实时地进行检测并以最快的方式发出报警信息，报警信息可以通过弹窗的方式显示在检测系统界面上。

（3）检测系统目的是将监控管理人员从繁杂枯燥的盯屏幕任务中解放出来，因此系统需要全自动地对监控画面进行不间断分析，无需人为操作，并将检测结果保存到服务器上，包括时间、地点、快照等数据。

吸烟检测系统的核心业务流程如图 3.1 所示。

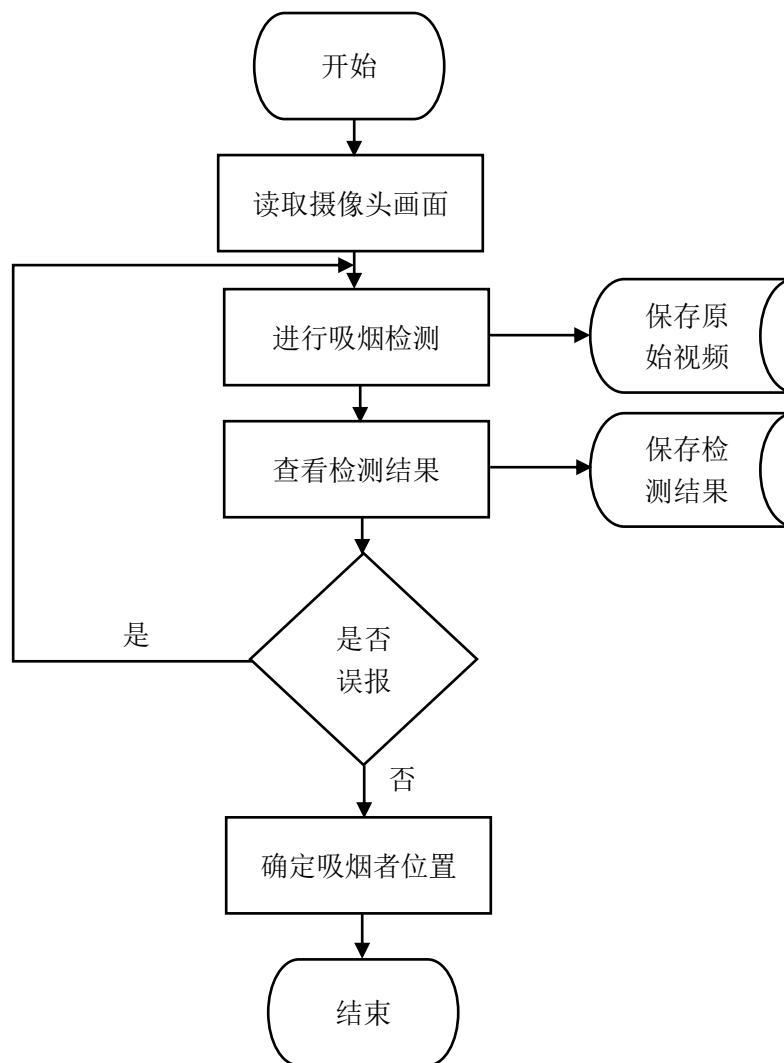


图 3.1 吸烟检测系统核心业务流程图

3.2 分析建模

检测系统的主要参与者为安全检查人员，通过对其需求进行分析整理后得出其所需要的功能有：

（1）对监控摄像头进行管理，包括添加摄像头、删除摄像头和切换目前显示的摄像头，并且如果摄像头出现异常还有对应提示信息。

（2）查看摄像头所拍摄的生产车间内的监控画面，如果检测到吸烟行为，还有查看所检测到吸烟者画面以确定吸烟者。

（3）查询检测系统以往的数据信息，包括查看监控摄像头所拍摄的生产车间视频以验证系统自动检测过程中是否存在漏报情况，以及查看检测系统所保存的吸烟检测结果以达到审核和取证的目的。

综上所述可以得出如表 3.1 所示的用户需求描述表。

表 3.1 用户需求描述

名称	参与者	说明
添加摄像头	安全检查人员	在检测系统中添加新的摄像头信息
删除摄像头	安全检查人员	删除检测系统中存在的某一摄像头信息
切换摄像头	安全检查人员	切换当前系统检测界面所显示的摄像头画面
摄像头异常提示	安全检查人员	提示某摄像头出现异常情况
查看监控画面	安全检查人员	查看当前摄像头所拍摄的监控画面
查看吸烟者画面	安全检查人员	查看系统当前检测到的吸烟者画面并确定吸烟者位置
查看监控视频	安全检查人员	根据时间信息查询以往生产车间的监控视频
查看检测结果	安全检查人员	根据时间信息查询以往的系统吸烟检测结果

根据用例描述表可以得出安全检查人员的用例图如图 3.2 所示

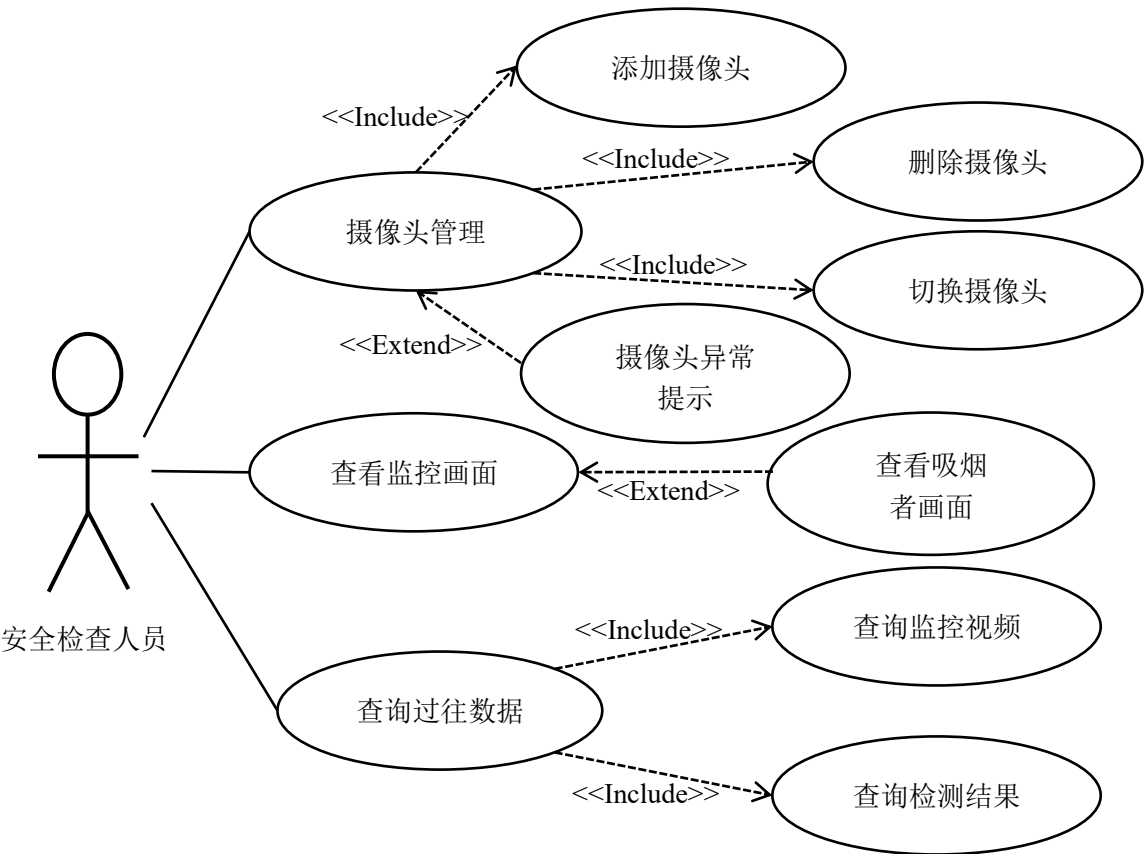


图 3.2 安全检查人员用例图

下面分别对安全检查人员的五个主要用例进行详细描述。首先是添加摄像头的用例描述如表 3.2 所示。

表 3.2 添加摄像头用例描述

用例名称	添加摄像头
用例描述	安全检查人员向系统中添加新的摄像头信息
触发条件	点击添加摄像头按钮
后置条件	系统对新的数据流进行吸烟检测
基本事件流	(1) 输入摄像头信息，包括 IP 地址、端口号、用户名、密码和所在位置 (2) 系统给新摄像头分配一个编号并与摄像头建立连接获取新的数据流
异常事件流	信息填写错误或摄像头未开启导致无法连接

查看监控画面的用例描述如表 3.3 所示。

表 3.3 查看监控画面用例描述

用例名称	查看监控画面
用例描述	安全检查人员查看摄像头所拍摄监控画面
触发条件	获取摄像头数据流
后置条件	持续读取摄像头数据流
基本事件流	(1) 获取摄像头数据流中一帧图像 (2) 更新系统可视化界面中的监控图像
异常事件流	摄像头未开启导致无法获取数据流

查看吸烟者画面的用例描述如表 3.4 所示。

表 3.4 查看吸烟者画面用例描述

用例名称	查看当前结果
用例描述	安全检查人员查看当前检测出的吸烟者画面
触发条件	检测系统报警
后置条件	继续检测下一帧
基本事件流	(1) 查看吸烟者画面 (2) 确定摄像头位置 (3) 关闭报警信号
异常事件流	错误的检测结果导致的报警

查看监控视频的用例描述如表 3.5 所示。

表 3.5 查看监控视频用例描述

用例名称	查看监控视频
用例描述	安全检查人员查看原始的监控视频
触发条件	点击视频查询按键
后置条件	进入视频查询界面
基本事件流	(1) 选择摄像头 (2) 选择日期和时间 (3) 查看监控视频
异常事件流	未找到对应的查询结果

查看之前结果的用例描述如表 3.6 所示。

表 3.6 查看检测结果用例描述

用例名称	查看之前结果
用例描述	安全检查人员查看系统以前的检测结果
触发条件	点击结果查询按键
后置条件	进入结果查询界面
基本事件流	(1) 选择摄像头 (2) 选择日期和时间 (3) 查看检测结果
异常事件流	未找到对应的查询结果

3.3 非功能性需求

该检测系统的非功能性需求有：

（1）可靠性：可靠性是系统稳定运行的保障，因此检测系统要能够 7×24 小时持续稳定运行，自动地进行吸烟行为检测任务，在检测吸烟行为发生时要在 2 秒内发出报警信息。如果遇到异常情况，检测系统能够在恢复正常情况后继续运行，且检测系统中数据保持不变。

（2）准确性：检测系统的性能需求首先是对吸烟行为检测的准确性。系统检测结果的精确率要大于 85%，避免出现大量的误报而导致保存大量的无效的检测结果，造成糟糕的用户体验。并且，系统检测结果的召回率应该大于 80%，以确保每一次吸烟行为都可以被检测到，尽量减少漏报情况的发生。

（3）实时性：检测系统的另一个性能需求的检测结果的实时性。检测系统传入的是摄像头实时数据流，该数据流的帧数一般为 25FPS，因此检测系统的处理速度至少需要达到 10FPS，即处理一帧画面的时间要在 0.1 秒左右，这样通过多线程的方式可以使检测系统的检测画面和摄像头实时数据流保持一致，避免出现检测画面的延迟情况。

（4）兼容性：既可以在 Windows 系统上正常运行，也可以在 Linux 系统上正常运行。

（5）持久性：检测系统需要保存的数据有原始的监控视频和检测到的吸烟者画面。为了方便后续对这些数据进行查询验证，需要将其进行持久化保存，保证系统可以正确地对至少 1 年以内所保存的数据信息进行查询与验证。

（6）友好性：检测系统需要拥有一个结构清晰的可视化系统界面，方便用户进行操作，给用户带来良好的使用体验。

3.4 本章小结

本章主要是采用面向对象分析的方法，通过与工厂生产人员和安全检查人员的沟通将其提出的需求进行分析与建模后，确定了吸烟检测系统需要实现的功能和需要满足的非功能需求，最终形成了系统的需求规格说明书，为后续概要设计、详细设计以及最终的实现与测试工作奠定了基

第 4 章 系统概要设计

4.1 系统概述

本系统是针对工厂复杂生产环境开发的自动吸烟检测系统，通过多个监控摄像头对生产车间进行图像信息采集，使用基于 YOLOv5 的两步检测算法进行吸烟行为检测，及时发现生产车间内吸烟行为并发出警报，使安全检查人员可以快速定位吸烟者所处位置，制止吸烟行为并进行警告或处罚，进而实现工厂全面禁烟的目的，保障了工作人员的生命财产安全和工厂车间的安全生产环境。

4.2 系统静态结构

吸烟检测系统的整体功能模块如图 4.1 所示，主要包含五个功能模块：预处理模块，人体识别模块，吸烟检测模块，画面显示模块和数据查询模块。其中，预处理模块包括流视频读取与保存和图像变换功能；人体识别模块包括人体检测和图像分割与变换功能；吸烟检测模块包括香烟检测和结果保存功能；画面显示模块包括可视化界面和弹窗报警功能；数据查询模块包括视频查询和结果查询功能。

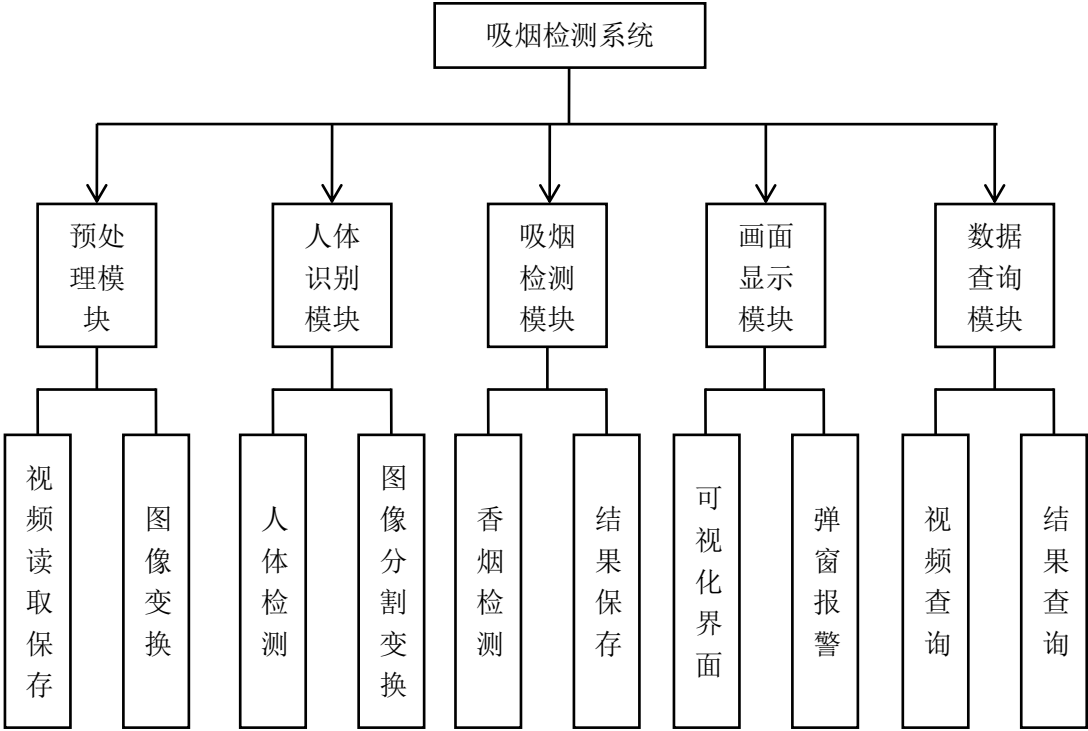


图 4.1 系统功能模块图

根据检测系统各个功能模块的特点以及性能需求，确定了系统的整体架构设计如图 4.2 所示，分别由硬件层、数据层、逻辑层、表现层和用户层组成。硬件层主要是检测系统所需的硬件设备，包括用于获取生产车间图像的监控摄像头、用于数据计算和逻辑控制的服务器以及用于储存系统数据的储存设备。数据层主要由摄像头信息、原始监控视频和吸烟检测结果组成，使用 MySQL 数据库对检测系统的数据进行持久化存储。逻辑层包括了这个系统的业务处理逻辑，由预处理模块从摄像头数据流中读取检测画面并进行处理，然后由人体识别模块分割出各个人体区域图像，再由吸烟检测模块进行处理，最后将得到的检测结果传入表现层。表现层主要由系统检测界面和结果查询界面组成，用于展示系统实时的检测结果以及所保存的原始监控视频和吸烟检测结果。最终在用户层向用户提供完整的吸烟检测系统。

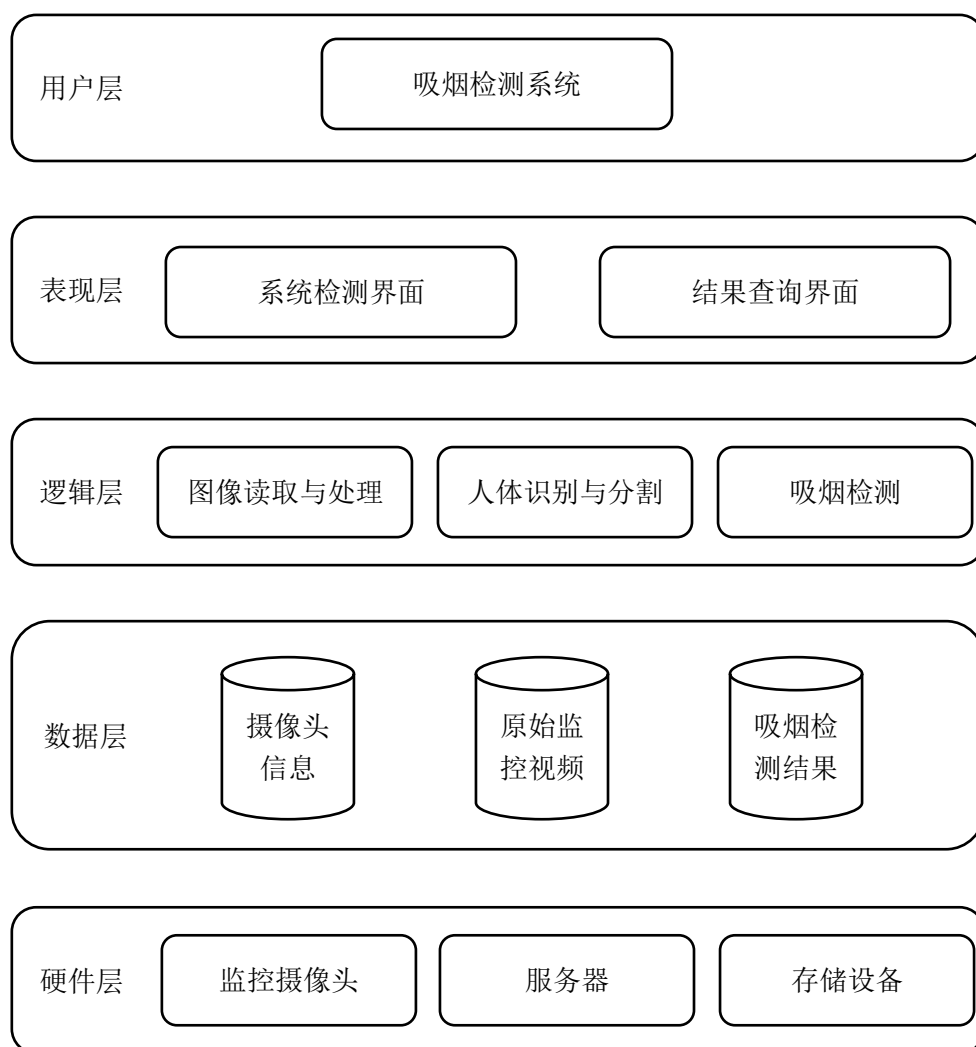


图 4.2 系统整体架构图

4.3 系统动态结构

吸烟检测系统的总体检测流程设计如下：首先，从摄像头实时数据流中取出一帧画面，将原始画面保存后传入画面显示模块，并且将图像处理至 640×384 大小后传入人体识别模块；然后，对处理后的画面进行人体识别，将识别出的人体区域进行分割，并进行自适应填充后传入吸烟检测模块；之后，并行地对传入的人体区域图像进行香烟检测，如果检测到香烟存在，就将检测框画到对应的人体区域图像上并作为检测结果保存后传入画面显示模块；最后，在可视化界面中显示摄像头原始画面，如果有检测结果传入，就进行弹窗报警并显示检测结果，否则检查数据流是否存在下一帧画面，存在就重复检测过程，不存在就停止显示画面并提示没有数据流传入。吸烟检测系统的活动图如图 4.3 所示。

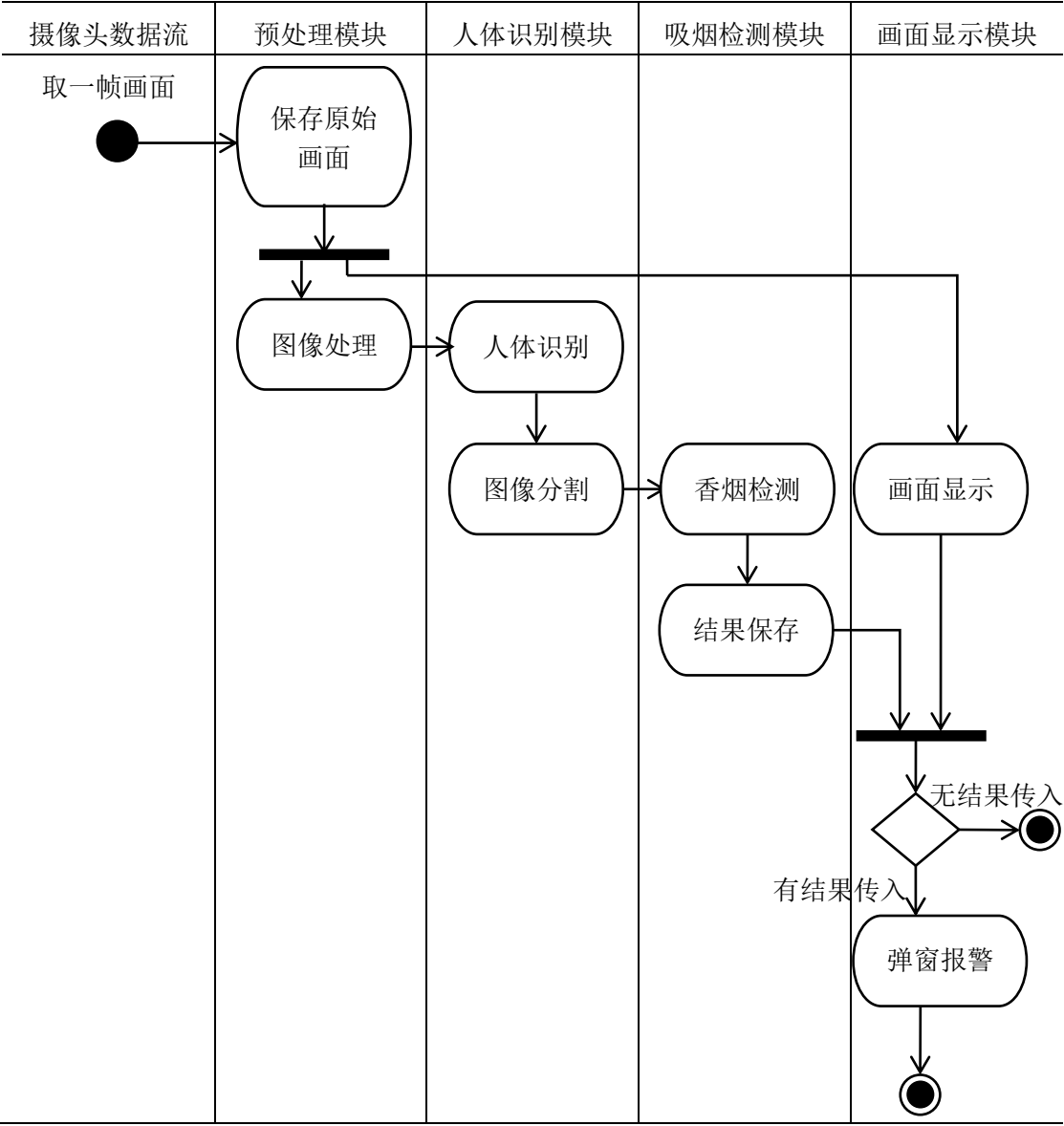


图 4.3 吸烟检测系统活动图

根据检测系统的总体检测流程可以得出检测画面数据在输入到输出过程中所经历的变换如图 4.4 所示。

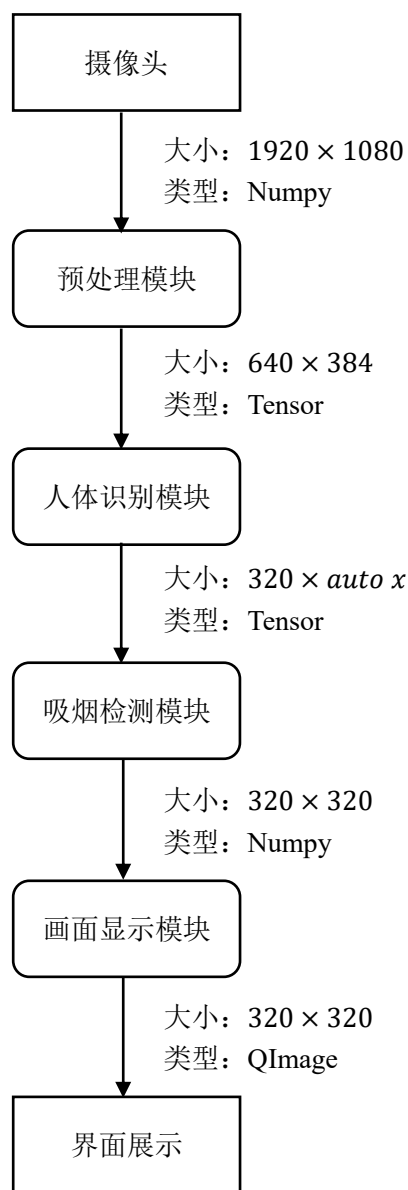


图 4.4 检测画面数据流图

4.4 系统数据管理

数据的存储与管理是一个系统能够持久运行的关键所在,因此良好的数据库设计对整个系统的设计和之后的数据分析都具有非常重大的意义。本文首先从 E-R 图 4.5 出发,对本系统所涉及到的不同实体的不同属性以及相互之间的联系进行分析。

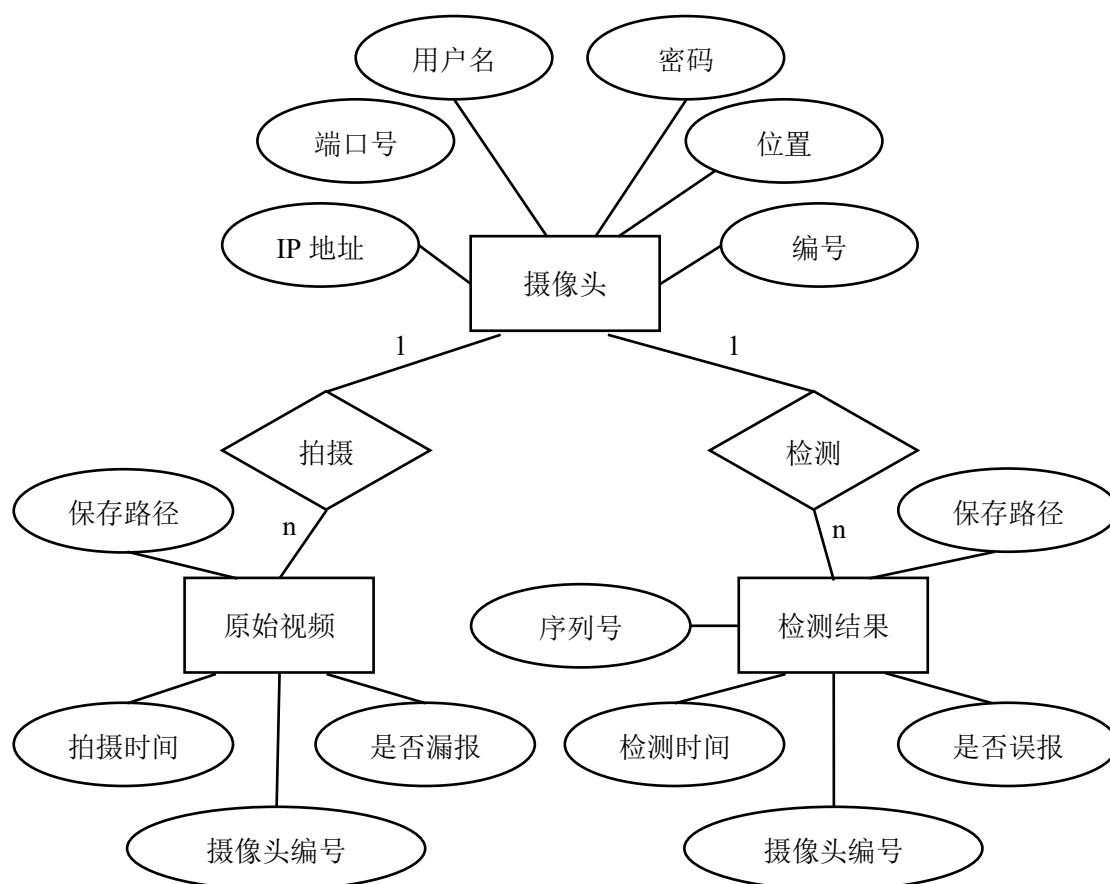


图 4.5 系统整体 E-R 图

从图 4.5 可知，吸烟检测系统主要涉及到三个实体：摄像头、原始视频和检测结果。摄像头具有六种属性，分别为 IP 地址、端口号、用户名、密码、位置和编号；原始视频具有四种属性，分别是拍摄时间、对应摄像头编号、是否有漏报和保存路径；检测结果具有五种属性，分别是序列号、检测时间、对应摄像头编号、是否为误报和保存路径。系统包含的关系是将摄像头拍摄画面保存为原始视频，以及对摄像头画面进行吸烟行为检测后保存下检测结果。安全检查人员可以添加或删除摄像头信息，查看原始视频及检测结果。最终设计了以下三个数据库表结构。

表 4.1 用来储存原始视频数据，主键为 Shoot_time。

表 4.1 原始视频表

字段名	数据类型	长度	说明
Shoot_time	varchar	20	拍摄时间
Camera_id	int	4	对应摄像头编号
Miss	bit	1	是否有漏报
Save_path	varchar	50	保存路径

表 4.2 用来储存检测结果数据，主键为 Result_id。

表 4.2 检测结果表

字段名	数据类型	长度	说明
Result_id	int	4	检测结果序列编号
Detection_time	varchar	20	检测时间
Camera_id	int	4	对应摄像头编号
Error	bit	1	是否为误报
Save_path	varchar	50	保存路径

表 4.3 用来储存摄像头信息，主键为 Camera_id。

表 4.3 摄像头表

字段名	数据类型	长度	说明
Camera_id	int	4	摄像头编号
IP_address	varchar	15	IP 地址
Port	int	1	端口号
Account	varchar	10	用户名
Password	varchar	20	密码
Address	varchar	50	摄像头所在位置

4.5 本章小结

本章主要是根据系统的需求规格说明书，对吸烟检测系统进行概要设计，确定了系统的五个功能模块：预处理模块，人体识别模块，吸烟检测模块，画面显示模块以及数据查询模块，描述了系统的主要检测流程，并对数据库存储方式进行设计，明确了吸烟检测系统的总体架构、动态流程以及数据库储存方式。

第 5 章 系统详细设计与实现

系统的详细设计主要是对概要设计中各功能模块的细节逻辑进行详细的设计,本章节将分别对吸烟检测系统中五个功能模块的详细流程以及实现细节进行阐述与说明。

5.1 预处理模块的实现

预处理模块主要用于从摄像头实时数据流中取出图像并保存以及对图像进行统一处理。由于吸烟检测系统需要同时接入多个摄像头数据流进行检测,故选用实时流传输协议(Real Time Streaming Protocol, RTSP)来实现检测系统与多个摄像头之间的数据传输。RTSP 协议是 TCP/IP 协议体系中的一个应用层协议,定义了一对多应用程序如何有效地通过 IP 网络传送多媒体数据,使用 TCP 或 UDP 完成数据的传输。RTSP 是类似 HTTP 协议的流控制协议,可以建立并控制一个或几个时间同步的连续媒体流,并且客户机和服务器都可以发出请求,是全双工通信。RTSP 的控制流和数据流是分开的,而且是基于文本的协议,具有较强的可扩展性,非常契合本系统的业务环境。

本模块通过多进程队列来读取多个摄像头数据流,并使用 OpenCV-Python 函数库实现了监控图像的预处理过程。因为读取摄像头数据流时会每帧画面都存入缓冲区,通常每秒钟会有 25 帧画面存入,然后需要一帧一帧地依次进行处理,而吸烟检测系统的处理速度是小于图像的存入速度,这就导致了缓冲区中待处理的图像越积越多,系统检测到的图像为几秒前的堆积数据。为了解决检测画面延迟的问题,本模块中每个读取进程都设置了两个线程,第一个线程通过 RTSP 协议从实时数据流依次读取出每一帧画面放入缓冲队列并保存到服务器,如果发现缓冲队列中存在两张图像(即处理速度跟不上写入速度),就将队列中的旧图像删除,写入新图像;第二个线程用来从缓冲队列中取出一帧图像,先使用 `Resize` 函数将图像的宽高都缩放至原来的三分之一,再使用 `copyMakeBorder` 函数在图像上下两侧分别填充 12 个像素的灰度值使图像达到 640×384 大小,然后把图像的颜色格式从 BGR 格式变为 RGB 格式,最后除以像素的最大值 255 进行归一化处理并转化为张量(Tensor)后传入人体检测模块。预处理模块的协作图如图 5.1 所示。

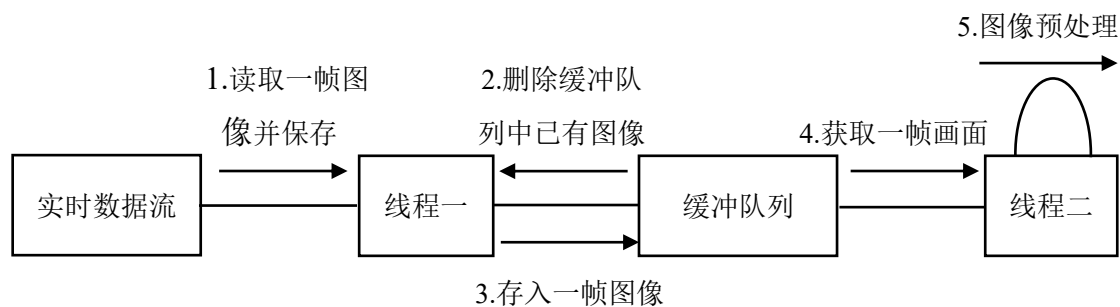


图 5.1 预处理模块协作图

5.2 人体识别模块的实现

人体识别模块主要用于从检测画面中识别人体区域并进行分割。将预处理模块传入的张量图像输入到训练好的人体识别模型中进行推断，得到了一组人体的检测框坐标($x1, y1, x2, y2$)，分别为左上角坐标的 $x1$ 值、 $y1$ 值和右下角坐标的 $x2$ 值、 $y2$ 值。而且，为了获得更加清晰的人体图像以取得更好的吸烟检测效果，本模块每次都只选择置信度最高的 8 个人体检测框并在原始图像上对相应区域进行分割，这就需要将所得到的检测框的坐标转换到原始图像的坐标系下，转换过程如下：首先将检测框的 y 轴坐标值减去预处理模块中 12 个像素的填充值，然后将检测框的各个坐标值除之前的缩放比例 0.33，即得到原始图像上的人体区域坐标，并以此切割出各个人体区域图像。人体切割结果如图 5.2 所示。



图 5.2 原始图像



切割后人体图像

切割出人体图像通常大小不一，如果依次对其进行吸烟检测的话需要花费大量时间，因此需要对所有切割出的图像进行统一处理，具体的处理过程如下：第一步，计算人体图像宽高缩放至 320 大小的比例，选择两者中较小值对图像进行等比例缩放；第二步，计算图像的较短边填充至 320 大小所需像素值，再使用 Numpy 中的 mod 函数取模，得到所需要填充的像素值后将图像的宽高填充至 32 的倍数，具体实现代码如图 5.3 所示。经过统一处理后的人体图像转化为张量后就可以一起传入吸烟检测模块进行检测了。

```
def letterbox(img, new_shape=(640, 640), color=(114, 114, 114), stride=32):
    # 当前图像的高宽 [height, width]
    shape = img.shape[:2]
    if isinstance(new_shape, int):
        new_shape = (new_shape, new_shape)
    # 计算缩放比例 (new / old)
    r = min(new_shape[0] / shape[0], new_shape[1] / shape[1])
    # 计算自适应填充距离
    ratio = r, r
    new_unpad = int(round(shape[1] * r)), int(round(shape[0] * r))
    dw, dh = new_shape[1] - new_unpad[0], new_shape[0] - new_unpad[1]
    dw, dh = np.mod(dw, stride), np.mod(dh, stride)
    # 计算其中一侧的填充距离
    dw /= 2
    dh /= 2
    # 图像宽高变换
    if shape[:2] != new_unpad:
        img = cv2.resize(img, new_unpad, interpolation=cv2.INTER_LINEAR)
    # 边界填充
    top, bottom = int(round(dh - 0.1)), int(round(dh + 0.1))
    left, right = int(round(dw - 0.1)), int(round(dw + 0.1))
    img = cv2.copyMakeBorder(img, top, bottom, left, right, cv2.BORDER_CONSTANT, value=color)
    return img, ratio, (dw, dh)
```

图 5.3 自适应填充代码图

5.3 吸烟检测模块的实现

吸烟检测模块主要用于在分割出的人体区域中进行香烟检测并保存吸烟者图像。该模块需要实现训练好的模型进行吸烟检测，所以本文使用了 PyTorch 进行吸烟检测模型的训练工作。由于构造的针对性数据集图像总量不够大，训练出的模型精度可能达不到吸烟检测系统的性能需求，因此选择使用微调(Fine-tune)的方式训练模型。微调是指基于别人训练好的预训练模型使用自己的数据集训练

更适合的模型，使用预训练模型中的参数作为网络的初始参数，省去了从头开始训练网络的过程，节约了大量的计算资源和训练时间，甚至可能提高了准确率。本文使用的是 Ultralytics 团队在 COCO2017 数据集上训练 300 个轮次(Epochs)后得到的预训练模型，其中包含有 80 种目标类别，而本文仅对香烟这一个类别进行检测，因此最后预测头的输出结果需要由(80+5)维变为(1+5)维。由于是基于预训练模型进行的微调，所以初始学习率设置为一个较低值，并且使用 WarmUp^[50]和 OneCycleLR^[51]策略更新学习率，优化器选择的是 SGD，批大小为 32，具体超参数设置如 5.4 所示。

```
lr0: 0.0032 # 初始学习率
lrf: 0.12 # OneCycleLR 比例
momentum: 0.843 # SGD 动量
weight_decay: 0.00036 # 衰减比例
warmup_epochs: 2.0 # 预热轮次
warmup_momentum: 0.5 # 预热动量
warmup_bias_lr: 0.05 # 预热偏置学习率
```

图 5.4 超参数设置

本文使用 NVIDIA GeForce RTX 3090 进行训练，训练 50 个轮次共耗时 10 个小时，最终训练结果如图 5.5 所示。

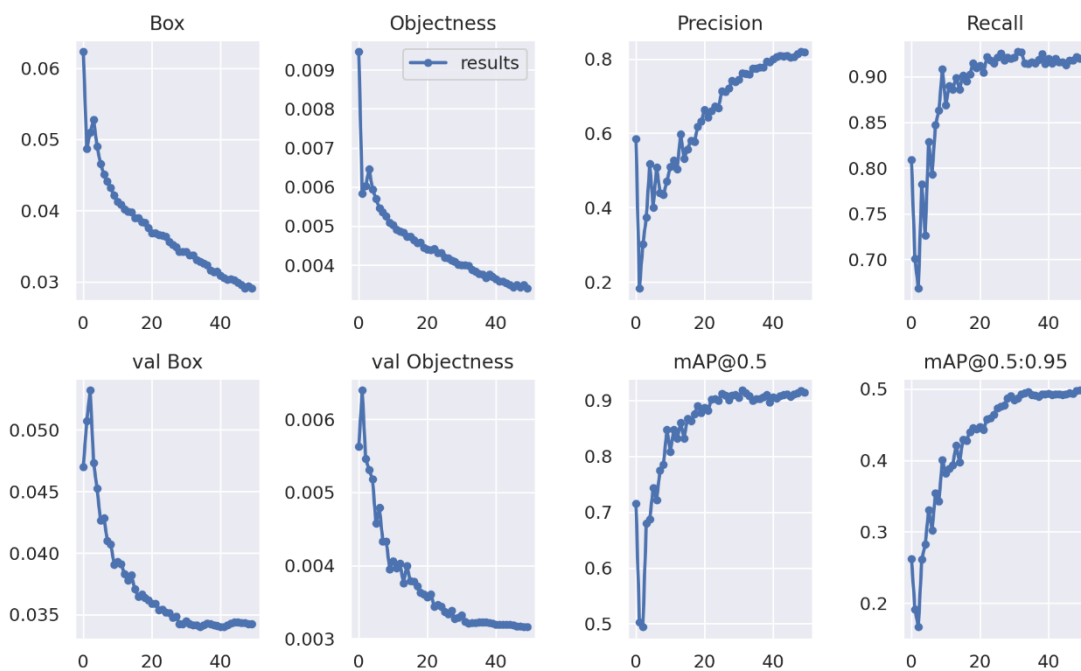


图 5.5 训练结果图

训练结果图中第一行前两个指标为训练集上的边框损失和目标损失，后面的分别为验证集上的准确率、召回率、边框损失、目标损失和两种 mAP 指标，可以看出由于使用预训练模型的原因，模型收敛的速度非常快，最终也满足了吸烟检测系统的性能需求。

将人体检测模块处理好的各个人体图像张量一起输入到训练好的香烟检测模型中，可以得到了一组香烟的检测框坐标及其对应的人体图像序号。为了进一步增强吸烟检测的准确性，本模块对置信度在 0.55 以下的香烟检测框进行了二次检测，具体操作为基于摄像头的软件开发工具包(SDK)实现控制摄像头向指定区域移动的功能，对可疑区域拉近后进行二次检测。

通过 SDK 控制摄像头移动的流程如图 5.6 所示，第一步调用 NET_DVR_Init 函数对整个 SDK 系统进行初始化；第二步调用 NET_DVR_Login_V40 实现设备用户注册功能，注册成功则返回用户 ID 作为其它功能操作的唯一标识；第三步调用云台控制模块的 NET_DVR_PTZControl_Other 函数通过之前返回用户 ID 控制摄像头移动以及 NET_DVR_PTZPreset_Other 函数对摄像头进行复位；最后调用 NET_DVR_Logout 函数注销设备用户并调用 NET_DVR_Cleanup 函数释放 SDK 系统相关资源。

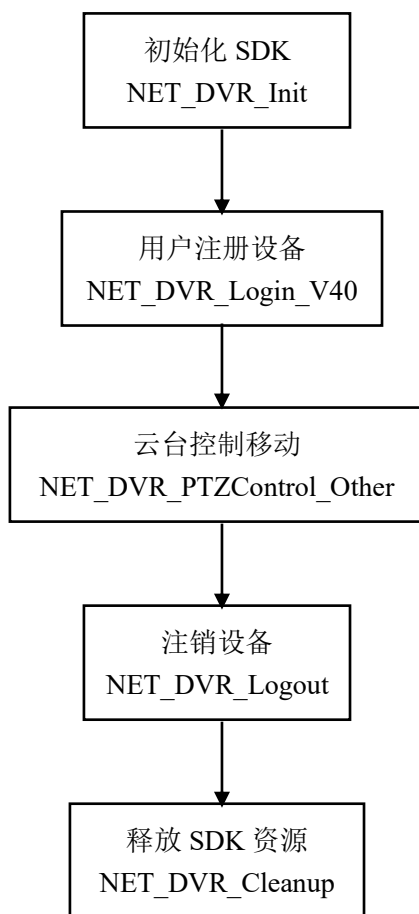


图 5.6 SDK 控制流程图

摄像头的 SDK 包中函数均为 C++ 编写的, 因此本文通过 Pybind11 将编译好的一系列 SDK 函数接口导入到本模块中, 实现了输入置信度在 0.55 以下的香烟检测框坐标后即可控制摄像头向对应区域移动的功能。在摄像头移动到可疑区域后调整摄像头焦距拉近画面进行二次检测, 经过 1.5s 后再将摄像头复位结束二次检测过程。二次检测过程的顺序图如图 5.7 所示。将最终确定的香烟检测框画到对应人体图像上并作为检测结果保存到服务器上, 同时将检测结果传入画面显示模块触发弹窗报警功能。

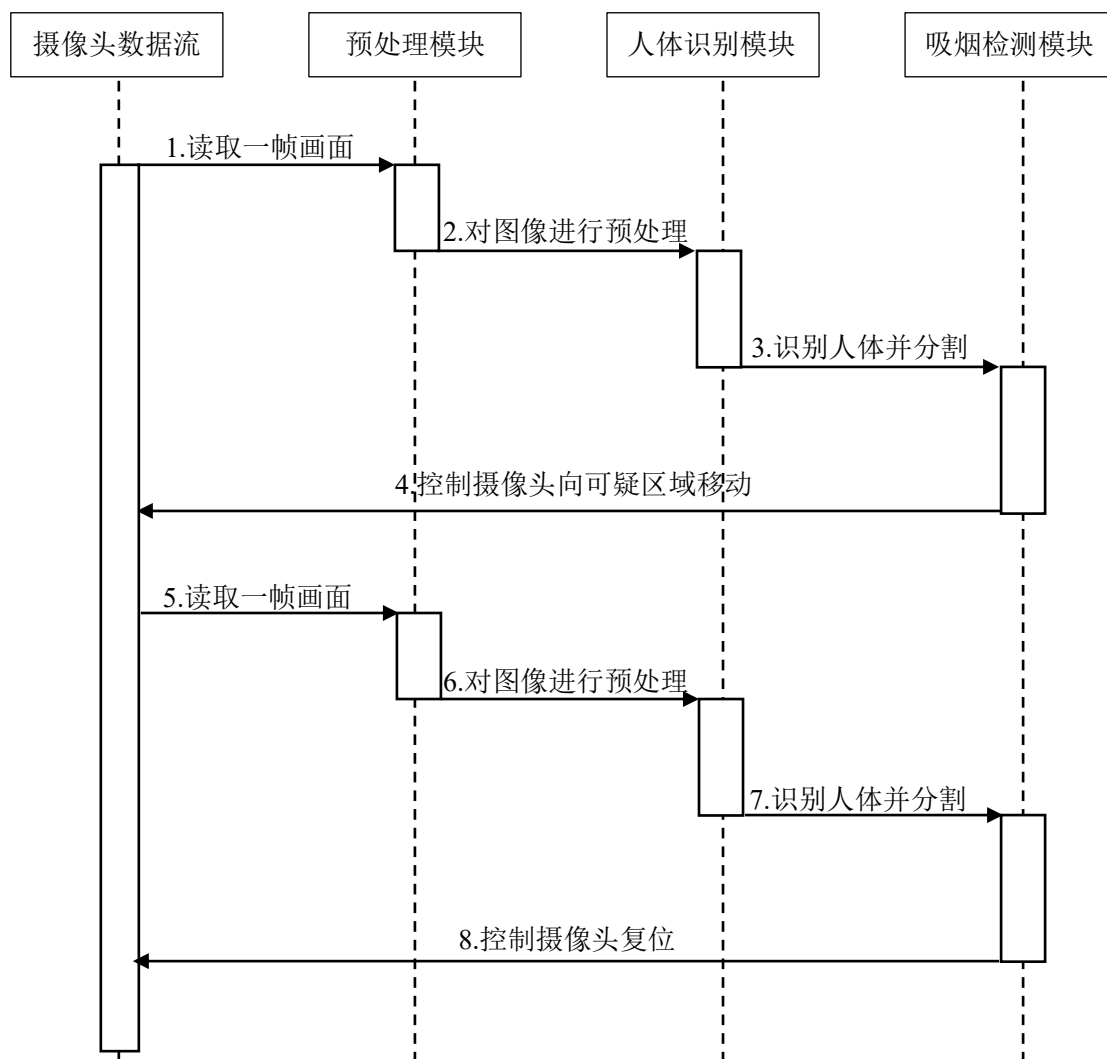


图 5.7 二次检测顺序图

5.4 画面显示模块的实现

画面显示模块主要用于实现检测系统的可视化界面以及快速弹窗报警功能。检测系统的可视化界面包括检测界面和查询界面, 检测界面中包括了实时画面显示区域和检测结果展示区域。本模块使用 PyQt5 进行系统可视化界面的开发,

PyQt 是 Qt 框架的 Python 实现，是最优秀的 GUI 图形界面架构之一，拥有高性能的窗口控件集合和丰富的 API 接口，能够跨平台运行在 Window、Linux 以及 Mac OS 系统上。主要是通过 QtDesigner 软件灵活地进行可视化界面的设计，在拖拽摆放各种窗口控件并进行属性设置的同时可以实时查看界面的整体效果，并且最终生成 ui 文件也可以通过 pyuic 工具自动转换为可以直接导入的 py 文件。

系统界面包含有检测界面和查询界面两个界面。检测界面主要由摄像头信息区域，摄像头画面查看区域和检测结果展示区域四部分组成，导航区域用来切换吸烟检测界面和结果查询界面，摄像头信息区域用来切换当前显示的摄像头以及添加新的摄像头，摄像头画面查看区域用来显示当前检测的摄像头实时监控画面，检测结果展示区域用来显示当前检测的摄像头最近的检测结果，省去了自行查询的过程。而查询界面主要由原始视频查询和检测结果查询两部分，通过选择或输入摄像头信息以及时间信息查询对应的原始视频或检测结果。检测界面如图 5.8 所示。



图 5.8 系统的检测界面

为了避免系统界面主窗体的卡死，弹窗报警功能需要使用另外一个线程进行实现，并通过信号量机制来实现报警窗口的快速弹出。当画面显示模块的主线程检测到有检测结果传入，就立刻发出一个弹窗信号，弹窗线程接受到这个信号后立即启动，弹出对应窗体显示检测结果和对应摄像头信息，帮助安全检查人员快

速定位吸烟者位置。弹窗报警界面如图 5.9 所示。

图 5.9 弹窗报警界面



5.5 数据查询模块的实现

数据查询模块主要用于根据指定的摄像头信息和时间信息查询对应摄像头拍摄的原始视频以进行复审和对应的检测系统的检测结果以进行取证。本模块使用 MySQL 建立数据库，根据概要设计中设计的数据库表储存摄像头信息、对应原始视频以及检测结果，为了方便查询和存储，数据库中存储的是原始视频和检测结果的保存路径。原始视频查询和检测查询过程基于一致，都是先确定所要查询的摄像头，检查数据库中是否存在该摄像头信息；再确定具体的日期和时间，最后在数据库中找到对应的原始视频和检测结果信息。此外，通过拍摄时间与检测时间来将原始视频和检测结果进行对应查看，并进行漏报与误报的标记，便于吸烟检测系统后期的维护升级，使吸烟行为检测算法可以更好地满足用户需要，完成算法的局部更新迭代。原始查询界面如图 5.10 所示，检测结果查询界面如图 5.11 所示。

视频查询

摄像头编号	拍摄时间	保存路径	是否误报

结果查询

摄像头IP:

时间:

查询

查询

图 5.10 视频查询界面

视频查询

序号	检测时间	摄像头编号	保存路径	是否误报

结果查询

摄像头IP:

时间:

查询

查询

图 5.11 结果查询界面

5.6 本章小结

本章主要是在概要设计的基础上对吸烟检测系统中五个功能模块的详细流程和内部算法进行了设计与描述，同时使用自己构造的数据集对吸烟检测模型进行了针对性的训练，最后完成了对各个功能模块的代码实现并给出了对应的演示效果图。

第 6 章 系统测试与分析

6.1 测试概要

系统测试是系统开发过程中必不可少的重要阶段，是确保系统开发质量的关键，它贯穿于系统开发的整个过程，尽可能地发掘系统中存在的问题和错误，可以及时地进行纠错，促进系统开发工作高效有质量地进行。

6.1.1 测试环境

吸烟检测系统的测试环境分为软件运行环境和硬件运行环境，软件环境主要是深度学习框架及其相关支持库，硬件环境主要是需要配置支持 CUDA 的 NVIDIA 显卡。具体的系统测试软件环境如表 6.1 所示，硬件环境如表 6.2 所示。

表 6.1 系统测试软件环境

名称	说明
Python 3.8	基础语言环境
PyTorch 1.7.1	深度学习框架
MySQL 8.0.20	数据库管理系统
Numpy	科学计算库
OpenCV-Python	图像处理库
Threading	多线程库
PyQt5	GUI 框架

表 6.2 系统测试硬件环境

名称	型号
CPU	Intel Core i7-6700 CPU@3.4GHz
操作系统	Windows 10
内存	16 GB
显卡	NVIDIA GeForce RTX 3090
硬盘	HDD 1TB

6.1.2 概述

吸烟检测系统的测试采用测试驱动开发(TDD)方式进行, 主要过程如下:

(1) 根据需求分析中确定的安全检查人员需求和系统概要设计中划分的五个功能模块, 采用黑盒法设计对应的测试用例, 当各个功能模块代码实现后立刻进行测试。

(2) 根据需求分析中的需求规格说明, 使用场景法对检测系统进行配置项测试, 测试检测系统的业务功能是否满足需求。

(3) 最后进行系统测试, 重点测试系统吸烟检测的精确度和系统处理的速度是否满足性能需求。

6.2 测试分析

测试分析主要是对吸烟检测系统的测试需求进行分析, 包括可以直接获取的显性需求和系统中隐含的隐性需求。在系统开发的不同过程, 测试的需求也有所不同, 本文所分析的测试需求有: 模块测试需求、配置项测试需求和系统测试需求等。下面对主要的测试需求展开分析。

6.2.1 模块测试需求获取

根据系统的概要设计, 吸烟检测系统共分为五个功能模块, 各个功能模块是否达到设计要求便是模块测试的需求。下面分别对各个功能模块的测试需求进行分析:

(1) 预处理模块: 该模块用于从摄像头数据流中取出图像并进行预处理, 依据概要设计要求, 主要测试该模块能否持续地从摄像头实时数据流中一帧一帧地获取图像, 并进行预处理转换为张量后输出, 而且还需要测试当摄像头实时数据流异常时该模块能否正常结束。

(2) 人体识别模块: 该模块用于在检测画面中识别出人体区域进行分割, 依据概要设计要求, 主要测试该模块能否正确识别出人体区域并分割出各个人体图像, 而且还需要测试当传入非张量数据时和未能从检测画面中识别出人体区域时该模块能否正常运行。

(3) 吸烟检测模块: 该模块用于在分割出的人体图像上检测香烟位置并保存检测结果, 依据概要设计要求, 主要测试该模块能否正确识别出香烟位置并在对应人体图像上画上检测框后进行保存, 而且还需要测试当传入非张量数据和未能从人体图像中检测出人香烟时该模块能否正常运行。

(4) 画面显示模块：该模块用于可视化界面的运行和实现弹窗报警功能，依据概要设计要求，主要测试该模块能否持续稳定地运行可视化界面程序并在检测到有检测结果传入时及时触发弹窗进行报警显示，而且还需要测试当没有摄像头实时画面传入和没有检测结果传入时该模块能否正常运行。

(5) 数据查询模块：该模块用于根据指定的摄像头信息和时间信息查询摄像头拍摄的原始视频和查询检测系统的检测结果，依据概要设计要求，主要测试该模块能否正确查询到指定的原始视频和检测结果，而且还需要测试当输入错误的查询条件时和当未找到所查询的数据时该模块能否正常运行。

6.2.2 配置项测试需求获取

根据吸烟检测系统的需求规格说明书，对安全检查人员所需功能进行配置项测试，主要是对摄像头管理功能的测试，包括添加摄像头、删除摄像头和切换摄像头。安全检查人员使用此功能管理吸烟检测系统中所接入的摄像头信息，通过填写摄像头的 IP 地址、端口号、用户名和密码，检测系统可以与指定摄像头建立 RTSP 连接并进行管理，获取摄像头实时监控画面进行吸烟行为检测。

6.2.3 系统测试需求获取

根据系统的需求规格说明书，吸烟检测系统需要进行可靠性测试和性能测试。检测系统的可靠性可以通过对稳定性、容错性和易恢复性的测试来衡量，而性能测试是测试系统检测的准确率和召回率以及处理速度是否满足说明书中指定的需求。

6.3 测试用例设计

针对各个阶段不同的测试需求，测试用例的设计和所采用的设计方法都有所不同，下面分别对各个阶段的测试用例进行设计：

6.3.1 模块测试用例设计

依据模块测试的需求，对吸烟检测系统的五个功能模块进行测试，各个模块的测试用例设计如下：

(1) 预处理模块：根据测试需求，该模块的功能是从摄像头数据流中取出画面并进行处理，测试目的是验证该模块可以持续地输出图像张量并保存对应的原始视频，以及当摄像头数据流异常时该模块的反应，具体测试用例设计如表 6.3

所示。

表 6.3 预处理模块测试用例

序号	输入	预期输出	说明
1	正确的摄像头 RTSP 地址	处理好的图像张量，并在数据库添加对应的原始视频	验证功能模块的有效性
2	异常结束的摄像头 RTSP 地址	提示无数据传入	测试功能模块的健壮性
3	错误的摄像头 RTSP 地址	提示无法连接摄像头	测试功能模块的健壮性

（2）人体识别模块：根据测试需求，该模块的功能是从检测画面上识别出人体区域并分割，测试目的是验证该模块可以正确分割出人体图像，以及当输入非图像张量和未能识别出人体区域时该模块的反应，具体测试用例设计如表 6.4 所示。

表 6.4 人体识别模块测试用例

序号	输入	预期输出	说明
4	包含人体区域的图片张量	分割后的人体图像张量	验证功能模块的有效性
5	不包含人体区域的图片张量	空张量	测试功能模块的健壮性
6	Numpy 数组等非张量数据	提示输入数据不是张量	测试功能模块的健壮性

（3）吸烟检测模块：根据测试需求，该模块的功能是从人体图像上检测香烟的位置，测试目的为验证该模块可以准确地定位香烟位置并保存检测结果，以及当输入非图像张量和未能检测出香烟时该模块的反应，具体测试用例设计如表 6.5 所示。

表 6.5 吸烟检测模块测试用例

序号	输入	预期输出	说明
7	包含香烟的人体图像张量	带有香烟检测框的人体图像，并在数据库添加对应的检测结果	验证功能模块的有效性
8	不包含香烟的人体图像张量	空张量	测试功能模块的健壮性
9	Numpy 数组等非张量数据	提示输入数据不是张量	测试功能模块的健壮性

（4）画面显示模块：根据测试需求，该模块的功能是运行可视化界面和触发报警弹窗，测试目的为验证该模块可以稳定运行可视化界面并快速完成弹窗报警过程，以及当无实时监控画面传入和无检测结果传入时该模块的反应，具体测试用例设计如表 6.6 所示。

表 6.6 画面显示模块测试用例

序号	输入	预期输出	说明
10	正常的摄像头画面	在界面对应区域上显示输入画面	验证功能模块的有效性
11	系统的检测结果	弹窗显示检测结果	验证功能模块的有效性
12	无摄像头画面输入	可视化界面正常运行	测试功能模块的健壮性

（5）数据查询模块：根据测试需求，该模块的功能是根据指定的摄像头信息和时间信息查询对应的原始视频和检测结果，测试目的是验证该模块可以准确找到所查询原始视频和检测结果，以及当查询输入不合法以及未找到查询数据时该模块的反应，具体测试用例设计图表 6.7 所示。

表 6.7 数据查询模块测试用例

序号	输入	预期输出	说明
13	正确的摄像头信息和正确的日期时间	对应的原始视频和检测结果	验证功能模块的有效性
14	不存在的摄像头信息或不存在的日期时间	提示未找到查询结果	测试功能模块的健壮性
15	错误的摄像头信息或非法的日期时间	提示查询条件错误	测试功能模块的健壮性

6.3.2 配置项测试用例设计

依据配置项测试需求,采用基于实际业务的场景设计法对吸烟检测系统的摄像头管理功能进行测试用例设计。该功能设计的基本流有:1.输入摄像头的 IP 地址、端口号、用户名、密码以及所处位置后,吸烟检测系统为摄像头分配一个编号后启动读取进程获取摄像头数据流;2.删除检测系统中某一摄像头信息;3.切换当前检测系统所显示的摄像头。该功能的备选流有:1.输入的摄像头信息错误造成检测系统无法添加摄像头信息;2.所指定的摄像头未开启导致检测系统无法获取摄像头数据流。基本流和备选流可以组合成各个场景,进而对每个场景设计设计用例,具体测试用例设计如表 6.8 所示。

表 6.8 配置项测试用例

序号	操作	预期输出	说明
16	输入正在运行的摄像头信息	摄像头 RSTP 流地址	基本流 1
17	删除系统中某一摄像头信息	删除成功	基本流 2
18	选择系统中某一摄像头信息	显示所指定摄像头画面	基本流 3
19	输入错误的摄像头信息(如 IP 地址不合法等)	提示摄像头信息格式错误	基本流 1, 备选流 1
20	输入未开启的摄像头信息	提示无法连接摄像头	基本流 1, 备选流 2
21	选择未开启的摄像头信息	提示无法连接摄像头	基本流 3, 备选流 2

6.3.3 系统测试用例设计

依据系统测试需求，系统测试主要是对吸烟检测系统进行稳定性测试、容错性测试、易恢复性测试和性能测试，下面分别对各个测试进行测试用例设计：

(1) 稳定性测试：系统测试需求中要求吸烟检测系统能够不需人为操作的、不间断地稳定运行，因此稳定性测试设计为在一台服务器持续运行吸烟检测系统三天，预期结果是检测系统可以一直持续运行，并能够自动保存摄像头原始视频及检测出的吸烟者结果。

(2) 容错性测试：依据系统测试需求，选择从以下面三个方面进行测试用例设计：某个已添加摄像头与吸烟检测系统连接异常断开，测试检测系统能否继续正常检测其它摄像头；保存原始视频或检测结果时检测系统与数据库连接断开，测试当重新连上数据库后检测系统能否在指定时间内重新将要保存的数据写入数据库；当添加摄像头信息输入错误或进行其它异常操作，检测系统能否给出提示，并且继续正常运行。

(3) 易恢复性测试：依据系统测试需求，选择采用人工干预的方式模拟硬件故障，从而测试吸烟检测系统的恢复能力，具体做法是：在一台服务器上正常运行检测系统一段时间，突然断电，然后重新通电，预期结果是检测系统可以在来电后 5 分钟内恢复正常运行，并且数据库信息保持不变。

(4) 性能测试：依据系统测试需求，对吸烟检测系统的检测精确率、召回率和处理速度进行测试，具体做法为使用提前录制好的 20 段吸烟测试视频输入检测系统，具体测试视图如图 6.1 所示，所有视频的时长均在 10s-20s 内，视频帧率均为 25 FPS，与常见的摄像头实时数据流帧率保持一致。预期结果是检测系统的精确率达到 85% 以上，召回率达到 80% 以上，处理速度至少达到 10FPS。



图 6.1 系统测试数据集

6.4 测试结果及分析

使用各个测试阶段所设计的测试用例，对吸烟检测系统展开了单元测试、配置项测试和系统测试，具体测试结果如下：

（1）在单元测试中对吸烟检测系统的五个功能模块都进行了测试，经过五次回归实验后，各个功能模块实际输出与预期输出达成一致，各个模块的功能和健壮性都得到了验证，达到了需求规格说明书中的需求。

（2）在配置项测试中对吸烟检测系统的摄像头管理功能进行测试，各个测试用例的实际输出与预期输出一致，所测业务功能满足设计需求。

（3）在系统测试中对吸烟检测系统进行稳定性测试、容错性测试和易恢复性测试。稳定性测试时系统正常运行情况如图 6.2 所示，容错性测试和易恢复性测试都达到了预测的结果。从测试结果可以看出，吸烟检测系统的可靠性满足需求规格说明书中的需求。

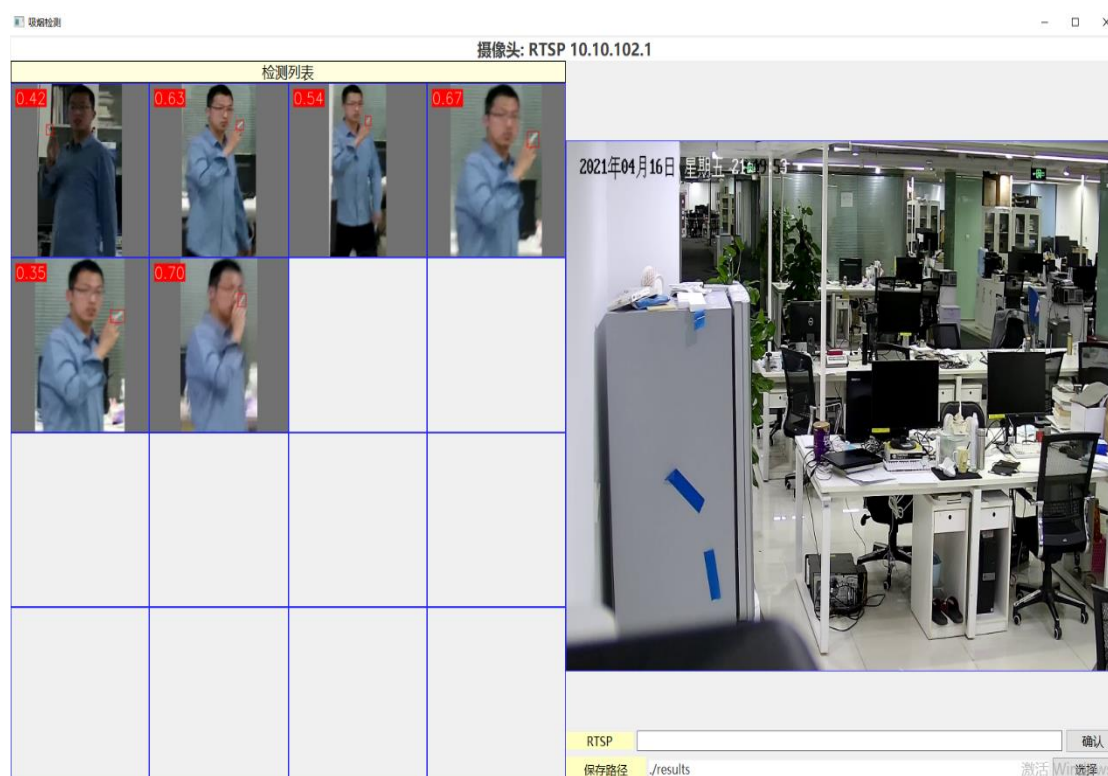


图 6.2 系统正常运行图

（4）在系统测试的性能测试中，20 段吸烟测试视频中总共有 114 人，其中吸烟者 33 人，非吸烟者 81 人，吸烟检测系统的最终检测结果如表 6.9 所示。

表 6.9 吸烟检测系统检测结果

	检测到吸烟行为	未检测到吸烟行为
吸烟者	27	6
非吸烟者	4	77

根据系统测试结果表，可以计算出检测系统的检测精确率为 87.1%，召回率为 81.8%。10 段时长 10s 的测试视频处理时间分别为 27.06s、26.42s、23.96s、25.40s、25.62s、24.81s、24.54s、25.64s、23.94s、24.56s，计算可得检测系统的平均处理时间为 25.19s；10 段时长为 20s 的测试视频处理时间分别为 51.83s、49.82s、50.46s、50.51s、49.87s、49.42s、49.98s、50.75s、48.17s、52.15s，计算可得检测系统的平均处理时间为 50.29s，最终可以得出检测系统的平均处理速度达到了 10 FPS，满足了需求规格说明书中的性能需求。

为了验证吸烟检测系统所使用的两步吸烟检测算法的有效性，本文使用 YOLOv5x 模型和 Faster R-CNN 模型在同样的数据集上进行训练与测试，最终测试结果如表 6.10 所示，可以看出本文提出的两步吸烟检测算法实现了检测速度与检测精度之间的平衡。

表 6.10 吸烟检测系统检测结果

算法模型	准确率 (Precision)	召回率 (Recall)	F1 分数 (F1-Sorce)	检测速度 (FPS)
YOLOv5x	0.81	0.86	0.83	28
Faster R-CNN	0.88	0.85	0.86	6
两步吸烟检测模型	0.87	0.82	0.85	10

6.5 测试结论

经过一系列的测试表明，吸烟检测系统能够长时间地稳定持续运行，通过摄像头收集工厂车间的图像信息，自动地进行吸烟行为检测并且能快速发起警报，帮助安全检查人员及时定位吸烟者位置。此外，如果发生断网、断电等异常情况造成系统宕机，在恢复正常情况后检测系统可以快速恢复到宕机前运行状态，并且数据库中数据不变。

但是在测试过程中发现，本吸烟检测系统中只针对常见的白色香烟进行检测，并且存在着对移动中的人体检测效果不佳的问题。此外，当检测画面中人数较多

时,检测系统的处理速度会变慢,导致可视化界面显示的实时监控画面出现卡顿,因此检测系统仍存在着一些改进的空间。

第7章 结论与展望

吸烟不仅危害人体健康，而且还存在着诱发火灾、爆炸等一系列重大安全事故的潜在风险。工厂车间不仅拥有大量的生产资料和机械设备，还存在着大量的工作人员操作设备进行生产活动，属于易燃易爆的高风险区域，因此全面禁烟正是保障工业安全生产环境的基本要求。本文针对工厂生产环境复杂多变、香烟本身目标过小难以检测的问题，提出了一种基于深度学习的两步吸烟检测算法，并在此基础上设计并实现了针对工厂生产车间的自动吸烟检测系统，将安全检查人员从频繁地巡查车间和枯燥地盯监控屏幕的工作中解放了出来，有效地解决了工厂禁烟工作效率低、人工成本高的问题。本文完成的主要工作如下：

(1) 提出了针对工厂复杂生产环境的吸烟检测算法。本文比较并分析了基于机器学习和基于深度学习两方面的各种目标检测算法，阐述了基于机器学习目标检测算法存在的问题。并且针对工厂生产环境复杂、香烟目标过小的问题，提出了一种基于 YOLOv5 的两步吸烟检测算法，通过先识别人体再检测香烟的方式，将香烟检测的范围缩小到每个识别出的人体所在区域，减少了复杂背景对检测的干扰，也变相地增大了香烟在检测画面中所占地比例，而且分别使用了不同大小的 YOLOv5 模型来对人体和香烟进行检测，实现了检测速度与检测精度的平衡。此外，还构建了针对性的吸烟行为数据集，对检测模型进行训练，进一步增强了模型的检测能力。

(2) 设计并实现了针对工厂生产车间的自动吸烟检测系统。本文首先采用面向对象分析的方式对安全检查人员的功能和性能需求进行建模，然后对吸烟检测系统进行了概要设计与详细设计。最后将系统设计与吸烟检测算法实现进行融合，完成了自动吸烟检测系统的开发与测试，极大地提高了工厂禁烟工作的效率。

本文开发出的自动吸烟检测系统基本能够满足安全检查人员的需求，但是在后续测试中，发现其仍存在一些不足之处，需要未来不断地改进和优化，主要包括以下两点：

(1) 本文设计的吸烟检测系统只针对最常见的白色烟卷黄色烟蒂型香烟进行检测，但是实际生活中香烟的品种多样而且样式不一，本文所构造的数据集数据量有限且香烟类型单一，难以实现对不同类型香烟的检测。所以未来需要收集并构造更加全面、多样的吸烟行为数据集，进一步增强检测模型的泛化能力。此外，本文所构造的数据集仅含有少量从网络上收集到具有工厂吸烟的行为特征数据，缺乏对工厂环境的针对性，未来会继续收集此类数据。

(2) 本文实现的吸烟检测系统对移动中的人体检测效果不佳, 主要原因是摄像头实时数据流在传输过程中需要进行编码, 而由于网络带宽的限制, 包含移动物体的图像无法使用更大的码率进行传输, 导致图像清晰度较差, 检测效果也随之降低。因此, 在本文提出的吸烟检测算法基础上, 结合更多的吸烟行为特征如烟雾、手势等, 或者利用传感器获取温度等多模态信息, 增强对移动中吸烟者的检测能力, 是未来改进的重点方向。

参 考 文 献

- [1] 《中国职业医学》编辑部. 吸烟危害健康, 远离吸烟[J]. 中国职业医学, 2016, 043(004): 486-486.
- [2] 中华人民共和国国家卫生和计划生育委员会. 《公共场所控制吸烟条例(送审稿)》起草说明[J]. 中国实用乡村医生杂志, 2015, 000(010): 1-2.
- [3] Senyurek V, Imtiaz M, Belsare P, et al. Cigarette Smoking Detection with An Inertial Sensor and A Smart Lighter[J]. Sensors, 2019, 19(3): 570-588.
- [4] 何媛, 程旭锋, 章珊伟, 等. 家用烟雾报警器设计研究[J]. 机械设计, 2016, 33(9): 105-108.
- [5] 阮颐, 王甲, 阮景. 烟雾报警器的市场应用综述[J]. 集成电路应用, 2018, 35(10): 52-56.
- [6] Lopez-Meyer P, Patil Y, Tiffany T, et al. Detection of hand-to-mouth gestures using a RF operated proximity sensor for monitoring cigarette smoking[J]. The open biomedical engineering journal, 2013, 9(1): 41-49.
- [7] Senyurek V Y, Imtiaz M H, Belsare P, et al. Smoking detection based on regularity analysis of hand to mouth gestures[J]. Biomedical Signal Processing and Control, 2019, 51: 106-112.
- [8] Cortes C, Vapnik V. Support-vector networks[J]. Mach Learning, 1995, 20: 273-297.
- [9] 丁宏杰. 基于视频烟雾的吸烟行为识别与研究[D]. 燕山大学, 2013.
- [10] 潘广贞, 元琴, 樊彩霞. 基于混合高斯模型和帧差法的吸烟检测算法[J]. 计算机工程设计, 2015, 33(5): 1290-1294+1336.
- [11] 刘远丁. 基于多特征融合的室内吸烟烟雾识别算法研究[D]. 燕山大学, 2018.
- [12] Kopilovic I, Vagvolgyi B, Sziranyi T. Application of panoramic annular lens for motion analysis tasks: surveillance and smoke detection[C]. Proceedings 15th International Conference on Pattern Recognition(ICPR-2000), Barcelona, 2000, 4: 714-717
- [13] 叶果, 程洪, 赵洋. 电影中吸烟活动识别[J]. 智能系统学报, 2011, 6(5): 440-444.
- [14] Echebarria I T, Imtiaz S A, Peng M. Monitoring smoking behaviour using a wearable acoustic sensor[C]. Proceedings of the Annual International Conference of the IEEE Engineering in Medicine and Biology Society, EMBS, 2017: 4459-4462.
- [15] 王超. 针对吸烟行为的手势识别算法研究[D]. 燕山大学, 2013.
- [16] Bhandari B, Lu J, Zheng X. Non-invasive sensor based automated smoking activity detection[C]. 2017 39th Annual International Conference of the IEEE Engineering in Medicine and Biology Society: Smarter Technology for a Healthier World, 2017: 845-848.

- [17] 林镇涛. 中距离户外施工人员吸烟行为检测方法研究[D]. 电子科技大学, 2020.
- [18] Zhang D, Chen J, Wang S. Smoking Image Detection Based on Convolutional Neural Networks[C]. 2018 IEEE 4th International Conference on Computer and Communications (ICCC), IEEE, 2018: 1509-1515.
- [19] Wu P, Hsieh J, Cheng J, et al. Human Smoking Event Detection Using Visual Interaction Clues[C]. 2010 20th International Conference on Pattern Recognition, IEEE, 2010: 4344–4347.
- [20] Zhao R, Wang M, Zhai Z, et al. Indoor Smoking Behavior Detection Based on YOLOv3-Tiny[C]. 2019 Chinese Automation Congress (CAC), 2019: 3477-3481.
- [21] 李倩. 基于深度学习的烟支检测技术研究与应用[D]. 西安邮电大学, 2020.
- [22] Artan Y, Balci B, Elihos A, et al. Vision Based Driver Smoking Behavior Detection Using Surveillance Camera Images[C]. ICIAP, 2019.
- [23] 程淑红, 马晓菲, 张仕军, 等. 基于多任务分类的吸烟行为检测[J]. 计量学报, 2020, 41(5): 538–543.
- [24] Jiao L, Zhang F, Liu F, et al. A Survey of Deep Learning-Based Object Detection[J]. IEEE Access, 2019, 7: 128837-128868.
- [25] Lowe D G. Distinctive Image Features from Scale-Invariant Keypoints[J]. International Journal of Computer Vision, 2004, 60(2): 91-110
- [26] Dalal N, Triggs B. Histograms of oriented gradients for human detection[C]. 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05), 2005: 886-893.
- [27] Freund Y, Schapire R E. A Decision-Theoretic Generalization of On-Line Learning and an Application to Boosting[J]. Journal of Computer and System Sciences, 1997, 55(1): 119-139.
- [28] Felzenszwalb P F, Girshick R B, McAllester D A, et al. Object Detection with Discriminatively Trained Part-Based Models[J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2010, 32(9): 1627-1645.
- [29] LeCun Y, Boser B, Denker J S, et al. Backpropagation Applied to Handwritten Zip Code Recognition[J]. Neural Computation, 1989, 1(4): 541-551.
- [30] 刘万军, 梁雪剑, 曲海成. 不同池化模型的卷积神经网络学习性能研究[J]. 中国图象图形学报, 2016, 21(09): 1178–1190.
- [31] Girshick R B, Donahue J, Darrell T, et al. Rich Feature Hierarchies for Accurate Object Detection and Semantic Segmentation[C]. 2014 IEEE Conference on Computer Vision and Pattern Recognition, 2014: 580-587.
- [32] Uijlings J, Sande K V, Gevers T, et al. Selective Search for Object Recognition[J].

- International Journal of Computer Vision, 2013, 104(2): 154-171.
- [33] Krizhevsky A, Sutskever I, Hinton G E. ImageNet classification with deep convolutional neural networks[J]. Communications of the ACM, 2012, 60: 84-90.
- [34] Girshick R B. Fast R-CNN[C]. 2015 IEEE International Conference on Computer Vision (ICCV), 2015: 1440-1448.
- [35] Ren S, He K, Girshick R B, et al. Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks[J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2015, 39: 1137-1149.
- [36] Redmon J, Divvala S, Girshick R B, et al. You Only Look Once: Unified, Real-Time Object Detection[C]. 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2016: 779-788.
- [37] Szegedy C, Liu W, Jia Y, et al. Going deeper with convolutions[C]. 2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2015: 1-9.
- [38] Liu W, Anguelov D, Erhan D, et al. SSD: Single Shot MultiBox Detector[C]. European Conference on Computer Vision (ECCV), 2016.
- [39] Simonyan K, Zisserman A. Very Deep Convolutional Networks for Large-Scale Image Recognition[J]. Computer Science, 2014.
- [40] Redmon J, Farhadi A. YOLO9000: Better, Faster, Stronger[C]. 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2017: 6517-6525.
- [41] Redmon J, Farhadi A. YOLOv3: An Incremental Improvement[J]. 2018.
- [42] Ioffe S, Szegedy C. Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift[J]. 2015.
- [43] He K, Zhang X, Ren S, et al. Deep Residual Learning for Image Recognition[C]. 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2016: 770-778.
- [44] Lin T, Dollár R, Girshick, K, et al. Feature Pyramid Networks for Object Detection[C]. 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2017:936-944.
- [45] Bochkovskiy A, Wang C Y, Liao H. YOLOv4: Optimal Speed and Accuracy of Object Detection[J]. 2020.
- [46] Liu S, Qi L, Qin H, et al. Path Aggregation Network for Instance Segmentation[J]. 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2018: 8759-8768.
- [47] Yao Z, Cao Y, Zheng S, et al. Cross-Iteration Batch Normalization[J]. 2020.
- [48] Wang C, Liao H, Wu Y, et al. CSPNet: A New Backbone that can Enhance Learning Capability of CNN[C]. 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW). 2020: 1571-1580.

- [49] He K, Zhang X, Ren S, et al. Spatial Pyramid Pooling in Deep Convolutional Networks for Visual Recognition[J]. IEEE Transactions on Pattern Analysis & Machine Intelligence, 2014, 37(9):1904-16.
- [50] He K, Zhang X, Ren S, et al. Deep Residual Learning for Image Recognition[C]. 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2016: 770-778.
- [51] Smith L N, Topin N. Super-Convergence: Very Fast Training of Neural Networks Using Large Learning Rates[J]. 2017.

致 谢

不知不觉中已经在科大度过了两年多的研究生生涯，时间转瞬即逝，在两年多的时间里我收获了很多，在此我要对指导过我的老师们，帮助过我的同学们，照顾过我的家人及朋友们表示最真诚的感谢！

首先，我要感谢我的导师汪增福教授在我毕业论文写作期间的精选指导和耐心督促，给予了我很多建设性的修改意见，让我对论文的不足之处有了更加深入的理解，并最终完成了毕业论文的写作。

其次，我要感谢在中科院信息工程研究所实习期间的导师侯锐教授、各位师兄和同事。在信工所实习的一年多时间里，我在深度学习图像处理领域收获颇丰，这离不开侯锐教授的指导，也离不开各位师兄和同事提供的帮助，正是如此我才能够完成各项科研任务以及项目实践。

然后，我要感谢软件学院各位老师，在研一时期对我的教导，尤其是郭燕老师在工程实践中指导我们如何钻研知识，学习技术，为以后的学习生涯奠定了基础。

最后，我要感谢我的家人一直以来对我的照顾，在这二十四年来陪伴着我度过了整个求学生涯，在背后不断地支持着我，给予我前进的动力。

2021 年 10 月 28 日