

# 武汉工程大学

## 毕业论文



题	目：	公共场景的人流密集度预警算法应用
姓	名：	徐平江
学	号：	1505120425
指导教师(职称)：	徐国庆（副教授）	
评阅老师：	郑更生（副教授）	
专	业：	物联网工程
班	级：	2015 物联网工程 1
所在学院：	计算机科学与工程学院	

2019 年 6 月



# 公共场景的人流密集度预警算法应用

## The Public Scene Crowd Density Early Warning Algorithm and Application

学生姓名 徐平江

指导教师 徐国庆（副教授）



# 目 录

摘 要.....	III
Abstract.....	V
第一章 绪论.....	- 1 -
1.1 选题来源及研究背景及意义.....	- 1 -
1.2 人流量统计系统的需求分析.....	- 4 -
1.3 主要研究内容及章节安排.....	- 6 -
第二章 视频处理概述.....	- 6 -
2.1 类型转换.....	- 7 -
2.2 轮廓检测.....	- 8 -
2.3 噪音处理.....	- 8 -
第三章 软件部分设计.....	- 11 -
3.1 opencv 简介.....	- 11 -
3.2 环境搭建.....	- 11 -
3.3 设计过程.....	- 15 -
3.4 框架设计.....	- 15 -
3.5 图片处理.....	- 17 -
3.6 前景检测.....	- 17 -
3.7 人数统计.....	- 18 -
3.8 模型的使用.....	- 20 -
3.9 小结.....	- 21 -
第四章 实验与测试.....	27
4.1 所需硬件设备.....	27
4.2 软件调用硬件.....	27
4.3 算法设计预期.....	28
4.4 高斯背景建模结果.....	30
4.5 前景检测结果.....	31
4.6 人与物的智能化判断.....	32
4.7 算法的复杂性判断.....	33
第五章 总结.....	35
致谢.....	- 33 -
参考文献.....	- 35 -
在校期间取得的成果.....	- 36 -



## 摘 要

随着社会科技的不断进步，人们的需求不断变大，这鞭策着现代技术的不断更新。各种智能化设备及系统不断得到改进，其中视频处理技术是对公共场所进行管理最核心的技术支持。为了更好的对公共场所进行监控，可以利用 `opencv` 设计一个公共场所的人流量预警系统。本系统利用 `opencv` 这个库来对从公共场所拍摄到的视频进行处理，利用人头来对人和物进行识别，利用高斯算法进行背景建模，利用像素的变化对人进行统计计数。利用流处理技术对图片数据进行转换，再将字符串进行处理。同时对于人与物可以做到识别区分，以保证此次计数的智能性和有效性。同样，系统还可以自动退出来，以保证程序的灵活性。本次设计将采用 `vs2017` 和 `opencv` 库的组合利用，使程序简单易懂，便于实现。

**关键词：**人流量统计；视频处理；高斯算法





## Abstract

With the continuous progress of social science and technology, our demand is constantly growing, which urges our technology to be constantly updated. All kinds of intelligent equipment and systems have been constantly improved, among which video processing technology is the most core technical support for the management of public places. In order to better monitor public places, I used opencv to design a public traffic warning system. This system USES the library of opencv to process video captured in public places, USES human heads to identify people and objects, USES gaussian algorithm for background modeling, and USES pixel changes to count people. Stream processing technology is used to convert image data into lines, and then strings are processed. At the same time for people and things can be done to identify the distinction, in order to ensure the intelligence and effectiveness of the count. At the same time, the system can also automatically exit to ensure the flexibility of the program. This design will use the combination of vs2017 and opencv software to make the program simple and easy to understand and realize.

**Keywords:** Traffic Statistics; Video Processing; Gaussian Algorithm



# 第一章 绪 论

## 1.1 选题来源及研究背景及意义

### 1.1.1 选题的来源

本次设计选题来源于生活,随着生活质量的提高,各种活动不断的出现在人们的视野中,一般性的大众活动几乎每时每刻都有,而且活动数量不断在攀升。包括各个人群群体,各个年龄阶层。他们有着不一样的追求,但做着一样的一件事,那就是集会。不管是偶然还是必然,对于他们来说都是活动,但对于这些人流量统计分析者来说,只有一种结果,那就是人流量的增长与减少。人流量的增长来源很广泛,广场演出、各种聚会等各类型活动都有,而且人流量时时刻刻都在发生变化,一般集会的人基本上从来不关心这个地区集会的数量究竟有多少,只是哪儿热闹哪儿去,而研究者需要关注。研究者需要利用现代的科技手段去服务人类。而视频处理就是人流科技的必要手段之一,利用某个地方的有利条件去测出人数,并做相应处理,从而解决商业和公共安全等领域的相关问题。

据相关信息来源显示,每年中国举办的各种大型文化体育集会达到了1万4千多场,累计拥有3亿人次参加。这些数据是商业和安全部门的数据基础。要利用好这些数据,并且可以根据这些数据进行统计分析。通过数据了解因团体活动引起的商业价值和安全隐患,要通过人流量的流通情况分析每一个时间段人流量流通的数量,分析在这个阶段每个街道,每个社区的公共场地的人员进出的情况。这些大众活动不仅使人类的精神生活更加丰富,而且可以使国家经济的发展进一步加快,也表明生活质量的提高,不再追求简单的温饱问题了,而是在追求更高的精神和物质追求。但集会必定导致人员密集度的暴涨,这会给公共安全带来很大的隐患,因此必须要处理好人群的流动走向,如果一旦发生意外,将引发严重后果。因此,必须要消除此类的公共安全问题。

纵观安全事故的案列,促使对公共安全加以重视,对于每一次安全事故的发生,都是一次惨痛的教训,公共安全部门和相关管理部门都有一定的责任,必须杜绝安全事故的再次发生。对于研究者分析,这些各种类型的活动拥有大量的人员参与,如果提前制定防止踩踏事件的机制化,就可能会避免因为人群拥挤,推倒,慌乱导致人员死伤事故的发生。因此安全机制和必要的人群疏导是至关重要的,是需要细密的思考去想出来的。

目前制定了一系列对公共安全的管理办法,以加强对公共安全的管控,这些管理办法对大型活动规划了工作,管理办法对责任进一步划分,从法律层面对公共管理人员进行职能强调,让公共管理人员明确自己的职能和作为,降低了因管理人员不重视安全事故而引发踩踏事故的风险,对相关活动提供了预警机制,对管理人员进行了指导。因此为了配合公共管理部门对人流量的提前了解,为了防止大型活动中踩踏事件的再一次的发生,设计了此次公共场所人流量预警设计,对人流量实时预警,达到对安全管理人员的提示作用。

一些商业老板会喜欢在人流流量流通大的地方设立实体商店,而对于外地来的老板并不知道一个城市哪些地方比较适合建立商店,因此他们会到处看,这是相当浪费时间的。因此,需要共享一些数据,将数据发布出去,这时候就可以服务更多的企业老板,为他们减少一些不必要的调研时间。

论文选题来源于商业和公共安全的关注。目前研究的地点包括广场,十字路口,商业街,公司门口,小区,学校公共活动场等。选题的因素包括商业的,安全的,也包括个人熟悉领域。安全领域是最迫切需要人流流量预警的,只要达到相关要求,才能保证人流流量安全预警的有效应用。

### 1.1.2 选题背景

在中国城镇化路径转型和高科技技术快速发展的背景下,人流流量统计及预警机制越来越受到有关部门和商家的重视,因此获得多源而维度丰富的人流流量信息至关重要。对于城市需求来说,视频处理领域供不应求,对于不断增加的城市化发展,商业发展越来越受到重视。视频处理不仅对于城市建设,国防安全,社会稳定和保障有作用,而且对于下一代人工智能的研究也是可以起到辅助作用。因此,鉴于视频处理领域的需求的增加,对视频处理必须要有进一步的设计和升级。对于功能也会进一步完善。而国内外对此项研究也比较少,在百度官网案例里面,只有撞针计数,对于一条路上是人流流量没有一次新的统计,而是一个个计数,到达指定地点才计数,而真正的人流流量不一定非要经过某个地方不可,人流流量代表某个地方人群的数量,不是往来经过的总数,因此,此类设计并不能达到统计某个地区人群数量,这种设计是达不到需求的。而本次设计

就是希望能大范围的统计人群数量,从设计初一直到现在,能参考的资料相对较少,对于研究的对应的参考资料来说,更相对比较较少。国外研究也有案例,对于小区人流采取人头计数,对于人头进行识别,这种人物识别具有智能性,可以区分人和物,因此,有参考的意义。但不管怎么样,对于该领域的发展前景较为乐观。

机器视觉是一个全新的领域,发展较晚,但需求量大。从目前一直在发展的智能小车,京东无人机,无人小车配送快递,无人飞机送快递,以及智能物流分拣系统上来看,视频处理起步较早,实验结果较多,但很多还是处于实验室研究阶段,很多还不能应用到社会生活中。视频处理在各个方面都有实际的需求,比如在医学领域,还有远程操控手术系统;在安防领域,有特定的人物跟踪识别。因此视频处理在诸多领域前景都是可观的,产生的市场价值也是巨大的。而现代摄像头的发展也是快速的,不仅可以做到摄像头白天识别,也可以做到晚上识别。不仅可以二维,还可以做到三维。本次研究的题目为公共场景的人流流量预警系统算法应用。人流流量还可以做到红外预警,红外线比较稳定,不管白天,雨天还是雾天都是可以用的。现代人工智能的发展,本系统设计人流流量对于智能领域是个不错的数据来源。人流流量可以提供人的走向,人群的密集度,对于深度学习有一个好的训练基础。人流流量对于商业来说是重要的,也是有预见性的。人流流量多的地方,相应的经济发展必定较快。因此人流流量密集度对于经济而言是一个重要的参考依据,不仅隐藏着商业价值,也隐藏着这个区域的市场规划。对于物流的发展,

无人运输车一直是一个重要的发展环节,无人分拣系统也是正在迫切需求着,总之在不管什么领域,无人化,高效率化是最终发展方向。因为人是容易疲惫的,而机器是可以不断的工作,如果坏了只需要换一个就好了,没电了换电池,摄像头坏了可以直接换摄像头,因此,智能化是本次研究的最终目标。本次研究就是基于视频处理的智能化,对于视频处理的发展还有很大的提升空间。一般而言,只要需要分析前方有没有人的地方,距离多远的地方,就需要视频处理技术,需要对视频图像进行处理。目前对于城市化研究中需要的人流量数据包括经过某个地方人数,某个场所的人流量通行速度,人流量通行的方向。人流量的计算很简单,划定一个时间段,通过的人数除以单位时间就是这个地方的人流量。

为了数据更可靠,可以取平均人流量,你可以在一天的多个时段进行计算,比如上、下班高峰时候。计算一次,取平均值。一般来说,不论何时,只要商业地带店面的人都是潜在顾客,但真正能成为顾客的,是无法预判的,要从店里每日光顾的人数来统计店里的进出状态以及消费记录。因此需求是大的,应用是广泛的,必须要不断的发展,以满足人日益提高的需求量。广泛的来讲,资料有很多,而且训练模块也有很多,但正如前面所述,对于人流量预警这块资料确实是太少,这为研究带来了一些难题。对于算法多种多样,目前对于解决群体量化分析的问题,一般使用基于密度检测算法去估计群体人数,不过群体量化分析对于行人个体信息的行为特征是分析不出的。论文的主要目的是对人流量化分析,根据目前研究热点来推动城市化的发展。通过以个人为本,来进行城市量化研究。为了充分利用城市摄像头来对行人信息进行统计,本文是基于实时处理视频来实现的。为城市各种规章制度以及城市规划提供了有效的数据支撑。

### 1.1.3 选题意义

对于此次设计,国内外研究比较少,而且各个公司对于这块的研究没有展示,有一些公司虽然做了相关研究,但也是比较传统的算法,而且效果也只是一般,因此此次选题的意义也就不言而喻。提取到这些数据对城市量化研究有重大意义。通过对出入口的人流量统计,可以对出入口的宽度设计进行修改;通过对人流量方向的判定,可以了解建筑物的构架设计是否合理;人流量的滞留时间,可以有效评估商场活动的吸引力;根据人流量变化情况,更有效对物业管理、维护人员进行调度;有些商店需要了解顾客是否在购买商品,因此,对消费者的购买率也是需要了解的。人流量对于商店来说是至关重要的,人就是消费者,消费者是商店收入的主要来源,也可以说是唯一来源。因此,不管这个城市发展水平怎么样,人流量就是唯一的指标。视频的处理不仅可以用来识别人群数量,而且可以识别出人,对公安有着不可忽视的作用。当犯罪分子逃跑时,或者某个作案嫌疑人在没人的地方作案时,监控是最重要的识别手段。只有监控才能分析出某个人从某个地方逃跑,才能分析出某个人是否在这个地区拥有不在场证据。而现在广场舞的兴起,集会的不断发生,街道的不断调整,这个都需要知道人流量数据,人流量数据与这些因素相互影响着。

从科技的角度出发,对于科技方面,不管是探索什么,都在一定程度上是有用的,技术不等于市场,也不等于经济收入。比如华为海思之前的技术储备,一旦美国发生政治制裁,就可

能导致华为整体业务停滞，储备技术就表示着一旦某个技术被垄断，将不能再使用了，就可以用另一个技术代替。因此，技术储备是一个企业乃至国家发展的必要手段。不可能全部受制于人，虽然可以暂时借鉴，但必须能够自己独立。而本次研究的技术，对于人流量的统计有一个新的思路，对于以前的研究有一个新的方案。以前的帧差法和噪音消除法不够准确，因此对于此次研究的意义，是对新方案的一个挑战，对技术发展是一次贡献。技术不等于商业，也可以说技术是走在商业的前面的。人流量统计这个设计对行人是有帮助的，对商业是有帮助的，对政府职能部门是有帮助的。首先在公共安全领域是一个大的需求，不管哪种场合都可以用到人流量统计，对于经费不足的公司或者个人不在考虑范围内。对于个人价值的实现也是有必要的，对于一个领域，一种新的尝试，这也是一种成就感。政府职能部门可以让这个技术用于发展安全预警机制，安全永远放在第一位，人流量对于地区人数具有预警作用。对于科技的意义，有了进一步的探索，有了技术的进一步提升。

## 1.2 人流量统计系统的需求分析

### 1.2.1 国内外研究现状

对于人流量统计分析，对于处理人流预警问题，传统有很多方式，比如用 `matlab` 或者 `opencv` 处理图像，也有用 `python` 解决人流量预警的问题。这个技术发展了几十年，有了一定的技术积累。

红外检测。红外检测是基于红外线技术，在某一个必经之地设置一条红外线，当有人通过设定好的红外线后，红外线会被暂时阻挡，此时光敏电阻因红外线的变化导致了电阻的变化，从而去改变了电路电流，此时计数表会加 1，从而触发计数功能，以此达到计数的目的。不过这种办法只能一个一个的计数。

基于人头检测，识别黑色人头之后，会选中人头这个像素变化的整体，利用边缘检测计数，选中所有的人头，通过选中的个数来计数。因此对于人头计数是可以大面积进行计数，不过缺点是当人头被遮挡，会导致计数不成功，从而影响计数的准确性。

卷积神经网络算法，通过将人群分成多维图像，利用每一维的不同来读出人群的位置，通过标记分开的人来进行计数。

人流量计数的方法还有很多。人群计数与人群密度分析算法在传统上可以分为三种：行人检测，视觉特征轨迹聚类，基于特征的回归。主流算法也有很多种，其中高斯模型算法实现起来比较简单，占用资源小，算法时间相对较快。国内外有较多的人研究人流量统计，其中目标识别方面的有人脸识别和头肩部识别。`opencv` 里也有多种分析算法，包括 `camshift` 算法对人头进行统计，直接对彩色图像精度分析。还有轨迹分析算法，这作用是对目标实时移动方向和移动过后的轨迹进行分析，同时判断目标是否已经移开指定区域，从而对区域内的目标进行数目统计。实现人流量统计的方法有很多，编程语言也有很多，不仅可以利用 `python` 实现人数统计，而且可以利用 `C++` 进行统计，而本文就是利用 `opencv3.4.4` 和 `vs2017` 进行人数的统计，利

用高斯背景建模来实现功能函数。利用图形来代替数值,使人群密度与位置更加直观。利用摄像头实时采取视频进行分析,不受人工误差和天气影响,可以随时采取某个街道或者小区内的人流量数据。高斯背景建模已经相对比较成熟,利用起来方便。国内外视频处理技术进展越来越快,提出的算法越来越多,现代视频领域需求越来越大,算法方式发展实现也越来越快。

### 1.2.2 生产需求现状

不可否认的是,社会上各个地方都安装了摄像头,大到最高安全部门,小到公交车,路边,酒店走廊等。每个地方都需要处理视频,每个地方都需要安全,每个地方都需要记录,这样大众才会拥有一定的监督和安全感。视频处理是关键技术。如何进行视频处理呢?处理到什么地步呢?这是一个双方的问题,供求双方的问题。首先看旅游等这第三四产业的问题。对于景区的客流量滞留问题一直没有一个有效的解决办法,大多数景点都是通过网络售票来解决景点门票的长时间排队等候问题。但这种管理办法只能解决买票排队的部分问题,而且不能从根本上解决买票地方的人流拥挤问题。只要有人流拥挤的地方,就有可能有踩踏事件的发生,就有可能有商业机遇,就有可能需要人工疏导,就有可能有犯罪问题。因此,视频处理的需求是广大的,来源于任何地方,任何时间。并且景区又不是只有一个地方的人员众多,多个地方同时会有大量的人群聚集,如果处理不好将出现踩踏事故,会造成安全事故,安全责任,甚至人员伤亡。这不仅仅降低了景区的运营效率,降低了人员购买力,提高了人员流通程度,而且给景区带来了脏乱差的坏印象。并且研究者需要根据人流的多少去改变景区的设施位置。有些人流量密集度高的项目应该错开来搭建,这样就可以防止人群过于拥挤,从而避免安全事故的发生。对于平常做法,旅游景区可以采用客流分析设备进行人流分析,曾经有过红外线、闸门机、感应装置等手段,在对客流进行管控的同时,这些解决措施的使用也会不方便。例如,红外线也会自身受到自然环境的温度等条件干扰,不是特别准确,对于客流量统计算不上精准;闸门机虽然精准,但一对一的通道限制游客自由通行,客流量过大时其效率非常低,反而阻塞了人流量的运行;感应装置识别差,造成误差数据太大。因此,提供一个优异的人流量统计系统迫在眉睫。

现在很多实体门店不知道人流量怎么样,而大型商店更需要人流量统计技术,对于人流量就是商店收入的基础,客源就是经济收入来源的核心。而小的商店老板要设立商店时都是凭自己一个人去看市场,希望能找到人流量多的地方去设立门店。这样既浪费了人力,也浪费了财力,这样的方式往往效率低下。不管什么时候,设计的人流量统计技术可以实时的统计人群数量,也可以积累人数,多个时间段来统计分析。也可以让人群最大时,统计数量和时刻,这样就会让人知道什么时候,客流量最好,什么时候,生意可能最佳。因此,人流量统计的需求是大的,对于商业需求最为迫切。相对商业领域,安全领域也一样重要,对于大型广场或者楼道间需要进行人流量监控,因为一旦发生踩踏事件,后果不堪设想。人流量统计分析系统对于人流量统计来说是一个很好的辅助工具。对于商场来说,人流量的走向将决定商场物品的摆放顺序,人流量的密集度将决定今天的营业额,防止小偷等措施也必须用到监控系统。对于管理者,

人流量预警系统可以更好的为其在安全和决策分析上提供帮助。

对于公共安全部门，现在虹膜分析技术，声音分析技术，指纹分析技术的出现，越来越感觉到了安全形式的变化。对于安全部门的责任来说，公共场所需要秩序有序，人员不做犯法的事，对于特别危险的人需要时时刻刻关注着。这些凭一个人用肉眼有些是看不到的，有些是看不过来的，因此，民生和安防领域需要视频处理技术，而且需求量广，社会意义大。

### 1.3 主要研究内容及章节安排

本文主要对公共场景进行人流量检测，通过高斯算法，边缘检测算法等对视频进行处理，通过边缘检测得出变化的东西，通过前景检测得出人和物体，同时需要不断的进行模型更换，以达到适用于任何突发状况下，模型的健壮性仍然完好。本系统主要工作方式是利用得出来的轮廓进行像素统计。每次得出来的改变作为新的物体的变化，适用对象为公共场景的人流，为了实验效果好，每次检测完后进行一次膨胀，目的是为了更准确的检测人流量的数量。利用容器来不断的存储得出来的图片，每次计算的人数显示在图片上。论文共为五章和两个附录。各个部分的简单介绍如下所示。

第一章，绪论。主要讲了设计的题材来源，本次设计的背和意义以及国内外研究现状。不同设计的意义不同，选题背景不同，选题来源不同，国内外研究不同，因此对于各个领域的意义有了一个比较详细的了解是必要的。同时，对国内外现状有一个简单的说明，作为现代商业化领域，人们对人流量预警的需求有了一个简单的了解。对商业和安全领域的需求等相关内容做了一个简单的介绍。

第二章，视频处理概述。主要讲如何获取视频以及如何去处理视频，如何对视频图像进行编码，如何进行类型转换，对接收到的视频如何分析利用，如何处理图片，如何输出图片等。同时包括如何进行轮廓检测，噪音处理，前景检测，背景建模等。

第三章，软件部分设计。主要讲如何设计程序，利用什么工具，如何进行框架设计，利用高斯算法背景建模，前景检测。模型的更新，如何进行人数统计等。

第四章，实验与测试。这个处理视频需要一定规格的cpu和显卡，还需要摄像头等硬件设备，包括WiFi组网模块的芯片，还有对一些其他所需要的硬件资源进行描述和要求。

第五章，总结。预期和结果都能达到满意的效果，对此次设计前前后后进行了一次技术分析和难度小结。

后面再就是总结，致谢，以及在校期间取得的成绩。



## 第二章 视频处理概述

### 2.1 类型转换

#### 2.1.1 目标视频提取

代码调用了接口函数，对于接口函数，声明某个库后，可以直接调用，因此直接利用打开摄像头函数 `open`，利用 `open` 函数打开所需要的摄像头，以此来进行对目标区域的拍摄。程序可以自动识别摄像头，安装一个摄像头，然后调整打开摄像头的数值，当数值为 0 时，默认打开电脑摄像头。也可以通过网络连接摄像头，实现远程开启，远程接收，远程控制等功能。拍摄的位置可以不断变换，动态调整。而且可以手动安装可拆除，对摄像头利用比较充分，可以减少不必要的浪费。因此，视频提取可以不断变化，不是一成不变的，而且视频提取具有随时行，多样性。当摄像头不存在时，也可以下载其他视频进行人数统计分析，或者帮别人分析数据。

此次利用了 `videocapture` 类，这是 `opencv` 库特有的类，可以进行视频提取和处理，其中用到了一些特别的函数，`imshow("image", img)`用来显示想要得到画面。`waitkey()`表示时间延迟。这些是提取目标时候的特征。目标提取的方式有很多种，用到的设备也不一样，在程序设计中，读取视频的时候，一定要加异常判断，检查打开是否成功，这对程序的鲁棒性有了一个基础。判断当前视频是否为空是程序是否能够继续运行的必要条件。读取视频也是此次设计的前提。

#### 2.1.2 图片输入

读取视频后，需要将视频转化为图像，通过每分钟一定的帧数，来分割视频，通过每次截取的视频来进行比较，然后通过一帧帧图像来分析人数。而图像的输入需要用到大量的技术。首先需要对图像的编码，一般是采用数字编码技术，但也可采用数模转换技术。图像编码技术发展了接近 100 年，到目前还在不断的探索过程中。编码方案为预测编码和变换域转换编码。图像预测编码一般有三种方法，分别为一维固定预测，二维固定预测，条件传输帧间预测。本次设计用的是条件传输帧间预测，其特点是前一帧同一平面的像素作为预测估值。对于图像的分析，高清图像用来分析视频肯定是好的，在像素上更清晰，对于利用像素为基础分析视频的研究者来说，肯定是越清晰越好，模拟的效果也会比较好。但问题是高清视频内存占用大，传输速度慢，目前高清图像压缩还需要进一步发展，但随着集成电路的发展，数字图像技术必定有一个质的飞跃。因此需要处理好效果，时间，资源上的关系。对于图片的显示和输入也用到了 `opencv` 的基础库，用 `imshow` 函数去显示图像，这也是 `opencv` 库所特有的，也是此次设计的优势之一。对于大视频怎么办呢？不可能限定一个时间去分析，也不可能用数组去存图片，因为数组的大小是有限的，不能控制视频的长短，也不能控制图片数量的多少，因此本次设计

用到了一个特有的东西，叫做容器。容器是一个可以动态调整内存的东西，对于视频的长短不在有所顾及。图片发输入以及存储不在拥有困难。

### 2.1.3 图片输出

为了能实时看到效果，检测背景模型是否建立正确，需要进行图片的输出。图片的输出还是要用到 `videocapture` 类，获取摄像头，得到视频并成功解码后，需要对图片进行输出以达到正常显示的效果。图片的输出需要显示出来，而显示器就是输出图片的最终效果的硬件基础。一般是每读取一帧就显示一次，而且会保持原图像的大小。其中利用到了数据转换，`stringstream` 主要是利用传过来的图像进行转换，将图像转换为 `string` 类型的变量，`string` 类型的变量优点是可以不断的增加图像的像素，达到动态存储的目的。`string` 就是双引号，可以存储中文，英文，字符等。`string` 拥有很多接口，其对象在于稳定，最开始便将值封装好了。不过如果没用声明 `std` 名字的话，使用 `string` 都会报错。C++ 中 `string` 类是对于图像处理来说特别好用的，相对于 `stringstream`，`string` 类型的字符串更加直观。而且 `string` 有着不限长度的优点。因此图片的输出具有不间断的效果。

## 2.2 轮廓检测

轮廓检测是前进检测的基础，首先需要预处理图片，对图片基本背景有一定的设置，这和前景检测的原理是一样的。对于运动变化的物体，如果像素差存在变化，那么就可以说明这个物体是一个独立的物体，这个物体就可以定成一个整体。也就说如果找到变化的范围后，就把他存起来。而轮廓检测的方法有很多种，通常采用的是二维高斯模板。还有一些情况，对于一个形状在另外一个形状的内部的问题，轮廓检测仅仅只能检测出外围，至于里面不是视频可以穿透的，那需要一些低频段的电磁波做辅助。而只有部分重叠的话，这个问题是好解决的。因为重叠必定意味着像素范围和轮廓的增大，这对于后面的图像处理来说并不是什么难事。对于 `opencv` 中层次结构，不管层次机构是什么样的，每一个轮廓都包含自己的信息：上一个轮廓是谁，下一个轮廓是谁，父轮廓是谁，子轮廓是谁。`opencv` 中检测图片轮廓用 `findContours` 函数，一般传入图片是用上面所说的二值图像。对于传值向量，每两个元素之间可以构成一个集合向量，轮廓就是由这些集合向量组成。每个人物就是由这些不同的轮廓来组成的。`hierarchy` 向量由 4 个 `int` 型变量组成，分别包括后轮廓、前轮廓、母轮廓和嵌入轮廓的索引数。如果轮廓与后一个轮廓或前一个轮廓、父轮廓和嵌入轮廓不对应，则层次结构的对应阶段将设置为默认值-1。定义轮廓检索模式的 `InMode` 检测所有轮廓，但仅为所有轮廓建立两个层次关系。外部轮廓为顶层。如果外轮廓包含其他轮廓信息，则内轮廓中的所有轮廓都属于顶级。

## 2.3 噪声处理

去掉噪声的方式有很多种，噪声的来源也有很多，光线的变化，水生波纹，影子的变化等。

对于细小的差别并不需要灵敏度太高。中值滤波器可以去除脉冲噪声和盐和胡椒噪声，有效地保护图像边缘信息。这些优良特性在线性滤波中不可用。中值滤波必须首先生成一个滤波模板，该模板根据命令模板中的像素值单调地生成二维数据序列，然后输出二维中值滤波。通过对图像二维模板中的奇数数据进行排序，将待处理的数据替换为排序后的中值。前面已经提到了中值滤波与线性滤波的优点。获得良好的滤波效果的代价是耗时的改进，可能是中值滤波的几倍。要合理的利用噪声处理算法，其中高斯噪声处理算法用得最多，也是本次设计处理噪声的算法基础，检测出噪声效果还算不错。



## 第三章 软件部分设计

### 3.1 opencv 简介

opencv 和其他的工具是不一样的，是一个更高级的语言工具。主要功能是进行视频获取和处理，对于一些像素点的误差有自带的算法进行处理。对于模型的建立方面，拥有比其他库无法比拟的优势。对于基础 API,opencv 拥有更多的种类。opencv 自项目成立以来获得了来自英特尔和谷歌的大力支持，尤其需要感谢 Itseez，该公司完成了早期开发的大部分工作。此后，Arraiy 团队加入该项目并负责维护始终开源和免费的 [opencv.org](http://opencv.org)。opencv 设计用于进行高效的计算，十分强调实时应用的开发。它由 C++ 语言编写并进行了深度优化，从而可以享受多线程处理的优势。opencv 的一个目标是提供易于使用的计算机视觉接口，从而帮助人们快速建立精巧的视觉应用。opencv 库包含 500 多个从计算机视觉各个领域衍生出来的功能，包括工业产品质量检测、医学图像处理、安全领域、交互操作、摄像机校准、双眼视觉和机器人。由于计算机视觉和机器学习经常一起使用，这些功能对于实现计算机视觉算法非常有效（包括对高级目标检测的最基本的过滤）。opencv 是用 C/C++ 开发的，它为 Python、Java、Matlab 等语言提供了接口。如果你不懂 C/C++，请阅读 C 语言教程和 C++ 教程。opencv 具有广泛的应用领域，包括图像拼接、图像去噪、产品质量检测、人机交互、人脸识别、运动识别、运动跟踪、无人驾驶等。opencv 还提供机器学习模块。可以使用机器学习算法，如正规贝叶斯、K-最近邻、支持向量机、决策树、随机森林、人工神经网络等。

### 3.2 环境搭建

#### 3.2.1 外部库的组建

主要有 C/C++ 语言编写，只需要关注“build”文件夹即可，对于现在的电脑，一般是安装 x64 位操作系统的文件。因此，配置环境路径的时候，需要配置 x64 文件夹的配置文件。如图 3.1 各系统文件。

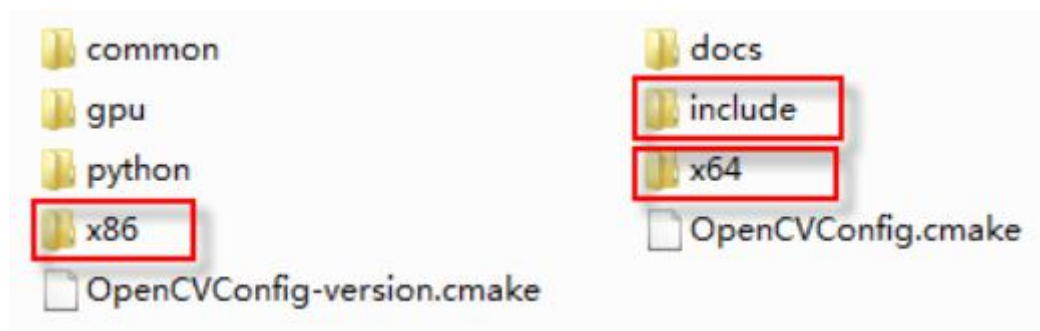


图 3.1 各类系统文件

“build\x64\vc15\bin”目录下放的是 opencv 运行时所需的动态运行库，“build\x64\vc15\lib”目录下放的是编译 opencv 程序时所需的动态链接库，“build\x64\vc15\staticlib”目录方式的静态编译时 opencv 所需要的静态链接库，如果是静态编译，运行时则不需要“build\x64\vc15\bin”目录的 DLL 文件，但编译后的文件较大。对于不同的库，适用的方法和效果是不一样的。

如果想一次配置对所有的 C++ 项目都适用，可以按如下配置：

1. 在 VS2010 打开任何一个 C++ 项目，然后选择“视图”=》“其他窗口”=》“属性管理器”，如下图 3.2 配置属性管理器。对于其他配置，最初没必要做过多的了解，一般时不需要配置的，等用的时候再配置也是可以的。



图 3.2 配置属性管理器

2. 打开的属性管理器窗口如下图 3.3 系统用户窗口，点击进去后，设置系统用户属性，将所需要的库目录和包含目录，附加依耐性等设置完毕。

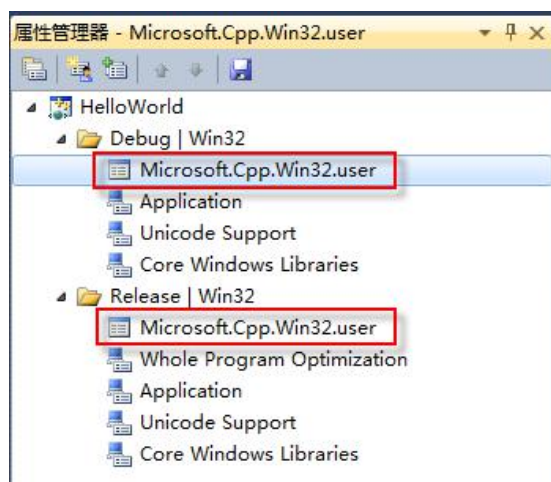


图 3.3 系统用户图

3. 在打开的属性管理器窗口中，展开“Debug|Win32”节点，双击“Micorsoft.Cpp.Win32.user”会打开全局属性设置窗口。

对“Release|Win32”节点下的“build\x86\vc10\bin”也做同样配置。

5. 对于系统路径，这个是从本机电脑上配置的。首先打开电脑属性界面，找到高级系统设置，点击环境变量，然后找到用户系统，从用户系统的路径里面配置路径。设置如下图所示，添加 opencv 路径。



图 3.6 变量设置

### 3.2.2 程序对库的调用

程序能不能利用好类和相关函数，就要看程序对 opencv 库的调用。不管是什么类型的库，都可以为编程的人减轻很多工作压力，不再继续自己去写面向过程的语言就是方便，这也是 c++ 和 c 语言不同的地方。opencv 核心库头文件是最重要的一个类库。d 代表调试版本，两者缺一不可。由于使用 VS2010，所以只关注目录 E:\Soft\opencv\build\x86\vc10。为了使 VS2010 可以编译 opencv 程序，需要对 VS2010 做一些设置，主要是让 VS2010 能找到 opencv 的头文件和链接库。opencv 是开源计算机视觉库的全名。也就是说，它是一个面向计算机视觉的开源 API 库。这意味着：

- (1) 科学研究和商业应用都可以使用它来开发；
- (2) 所有 API 函数都是开源代码，它只为一些常用的、经典的和流行的算法提供 API。

程序调用库就是在调用库里面的函数，一般来说，调用就会利用申明一个对象，利用对象调用相关函数，以及利用对象来对各个功能函数进行关系确定。



### 3.3 设计过程

本系统对行人目标进行跟踪操作，最初想法是先准备好硬件：摄像头，处理器，显示屏，硬盘等，然后利用摄像头拍摄要进行人流预警的地方，再对获得的人群进行统计计数，最后分析效果并不断对算法进行改进。因此本次设计提出的是利用 `opencv` 对收集的视频进行图像处理，利用 SSD 网络进行行人检测，利用 kcf 算法对目标人群进行跟踪，可以检测近大远小的头像问题。利用全卷积的网络，对网络结构进行融合，使融合的网络比单一的比单一的网络有更好的性能。可以在 `opencv` 上处理图像，也可以在 `matlab2015` 上处理图像，也有些人习惯于用 `matlab`。因此，可以利用 `matlab2015` 进行图像处理。从传统摄像头照出来的图像可以看出，人的形状更符合椭圆。这些问题，处理办法是：对于近大远小的问题，一般可以利用线性回归近似，利用人头的位置来确定人的位置，从而使图像内的人数更准确。因为每个场景都有其特有的属性，可以使用数据驱动的方法对 CNN 模型的目标场景进行检测。对于人像遮挡问题：我们可以准备候选场景作为参考，提取类似密度分布的图像，经过预训练模型图像，有相似图像的人群数量是一致的。对于夜间检测，就需要夜间模型进行单独检测，同时还要跟踪模型进行弥补，利用不同的帧对同一人进行匹配。自己提出的方案和解决的问题为：计算出目标区域的人流数量，分析出目标区域的人流动向以及行为特征因此，在测试阶段支持各种尺度的图片的检测。卷积神经网络模型算法主要针对人流预警问题开展研究，目前的算法进展为正在调试过程中，存在的问题主要包括：卷积神经网络模型算法的运行效率有待提高，算法的精确度有待加强。

### 3.4 框架设计

对于此次框架设计，无论从理论再到实践，无论从工具再到算法，都是经过很长时间讨论后确定的。计划是首先必须了解 `opencv` 的特有属性，然后对属于 `opencv` 的 `api` 进行调用。其次，必须建立多个窗口，对不同模块的函数进行分析对比，对不同效果的实验进行选择。对于算法设计，同时对多个算法进行选择，对每个分析结果进行展示。其次，每个人经过的地方必须有个标记，用框框标记行人，以达到锁定的目的。对于物体与人的区分，必须智能化。整体框架不能模糊，必须清楚整体的构架。对于计数和跟踪构架，可以用到一些技术去完成。而且必须去噪，每个噪点都可能对结果产生影响。首先对用到的模块进行分析，对于此次框架的设计需要用到大量的 `opencv` 模块，每种模块都有其特殊的功能。对于框架的搭建，需要花费不少功夫，这必须对如何处理视频以及如何对数进行实时接收有充分的了解。下面就是 `opencv` 的所有模块，按照宏定义的顺序依次介绍。

#### 3.4.1 [contrib]定标与三维的组合缩写模块

`opencv` 有很多板块是新增的，在官网上下下载的没有 `contrib` 板块，需要另外下载。这个模块对于图像处理也是比较重要的模块，本模块主要涉及机器定标和三维重建，包括基本的多视

图几何计算方法、单立体摄像机定标、物理状态估计、立体相似性计算方法、三维信息重建。

### 3.4.2[core]核心功能模块

这个板块有很多功能，基本上每次用 `opencv` 都会用到这个板块。里面有很多类，各种画图功能和处理图片的类都有。包括以下内容：`opencv` 基本数据结构、动态数据结构、绘图功能、数据组操作相关功能、辅助功能、系统功能和宏、与 `openGL` 的交互。此次应用中首先将核心功能模块里面的 `mat` 类存放一些图片，为读取到的视频开辟一些内存地址，然后为图片传递图片像素，为图片定义一些点。基本上显示了图片的坐标信息。

### 3.4.3[Feature2d]二维功能框架

它包含以下内容：特征检测与描述、恐惧探测器通用接口、描述提取器通用接口、描述特征通用接口、通用描述访问器通用接口、关键点绘制功能和分配功能绘制功能。

### 3.4.4[flann]近邻快速搜索算法库

近邻搜索技术正在广泛的使用，主要包括两部分：快速近邻搜索和聚合类。这个模块对于图像处理的好处很大，能够快速的找到某一空间中，离某一特定一点的最近的一点。而对于一般的邻近搜索技术，只能搜索一维的点，而对于 `flann` 最邻近搜索算法，精准度和时间上都是好很多的。

### 3.4.5 gpu 模块

是由 `gpu` 加速的计算机视觉模块，可以通过上传数据到 `gpu` 来增加图像处理的速率。初始下载 `opencv` 时，并没有开启 `gpu` 功能。虽然 `gpu` 运算速度很快，但上传到 `gpu` 很费时间，因此对于少量图像处理不建议使用 `gpu`。

### 3.4.6[Legacy]代码库

虽然这个库不怎么用，但还是包含很多接口，它保持向下兼容，包括运动分析、期望最大化、直接图、特征检测和描述、描述符提取程序的通用接口、通用描述匹配公共接口端口等。

### 3.4.7[stitching]图像拼接模块

图像拼接模块又分为很多子模块，拼接流水线，特点寻找和匹配图像，估计旋转，自动校准，图片歪斜，接缝估测，曝光补偿，图片混合等。主要通过一些特征点进行匹配，利用一些

算法对特征点进行提取。

### 3.4.8 模块框架的使用

设置框架的前提是设置头文件，取 `opencv` 图片处理库，将头文件设置为 `#include<opencv2/opencv.hpp>` 利用 `sstream` 对数据格式的转换，`sstream` 引入了 C++ 串流的输入输出操作。而数据流之间需要转换，本次设计用 `stringstream` 和 `string`。`string` 是字符串类型，可以无限提供长度，而 `stringstream` 主要功能是将字符串分割。利用对象把资料输入。输入后设定窗口，灵活设置窗口，将参数设为灵活变动的大小，这对于视频处理以及观察范围有好处。`namedWindow` 设成四个，一个作为前景检测的，一个用来提取高斯模型建立的视频处理结果，一个用来得到计数统计窗口，一个用来深度学习。而深度学习这个板块没有继续做下去，由于设计的难度大，时间有限，只做到了基于高斯模型的前景计数，检测效果但还是达到了预先的要求。对于每个功能模块，都需要熟练的使用。

前面进行背景建模后，再经过前景检测后，基本整体框架已经出来了，然后就是计数和分类。分类环节必须清楚人群的叠加，有可能人后面还有人，必须达到分析人后面的人的效果。计数框架要对每一个像素变化区域进行计数，无论是一个人还是两个人，都必须清楚的了解。框架的作用就是必须设计到每一个模块。每一个步骤，如果一个环节出了错误，那么整体上的结果就会出现错误，所以每一个框架必须用对。`opencv` 的一些功能模块是建立此次框架的重点，有些东西是可以用 C++ 写出来的，但有些库要花大量的时间，一般人不可能完成。一个好的框架设计决定一个好的毕业设计。而且好的效果也需要找到一个更合适的场景和好的算法。

### 3.5 图片处理

对于图片处理就需要进行数据传输，大多基于 `tcp/ip` 模式传输数据的信息，都可以通过结构体来进行传输和赋值。对于结构体加上指针之后，就可以变成节点，每个节点相互链接，就可以编程链表，而链表是数据结构的一种存储方式。

视频可以通过手机摄像头，或者其他厂家的摄像头，或者直接读取视频。视频通过 `tcp/ip` 传输给服务器，通过服务器对视频进行处理。其中主要用到 `opencv`，这是一个库，专门处理视频图片等，不仅可以实时处理图片视频，还可以处理下载的视频图片等。利用获得的视频进行流转换，成功成为流之后再进行字符串处理，转为 `string`，并对 `string` 变量进行分析。

### 3.6 前景检测

在视频处理领域中，运动对象的分割是一项基本的技术。例如人机互动，人脸识别，多媒体索引，人脸检测与跟踪，手机视频的编辑，照相机的美颜功能等。精确的分割可以极大的对跟踪对象进行提取分析。对于目标提取，主要用到的是高斯算法。高斯算法不在乎每个小变化的区别，是对于大致人流的提取。

高斯算法中涉及到了大量的参数，参数的不同会对检测的结果产生很大的影响。参数总共分为两类，形参和实参。形参和实参的使用方法如下：

形式参数是在定义函数名和函数体的时候使用，是为了接受调用该函数时，该函数传过来的参数。主调和被调函数之间会有一些值的赋予，这时候需要形式参数来当中介。实际参数是调用时传给函数的实际参数，拥有实际的值。最终目的是将实参传给形参，然后传给下一个函数。形参是在调用时被赋予内存结构，而实参时在定义时便有了内存单元。实参可以是各种类型的量，包括常量，变量和函数，函数调用的时候，实参必须有确定的值。因此实参因预先给其初始化。在函数调用的时候，只能把实参传给形参，而不能反过来。参数传完之后，就要利用参数来构造高斯模型算法。在图形上，越接近模型的中心，效果越好。也是尽量吻合实际人数的地方。尽可能多的除去一些无关噪声的影响。所谓的高斯模型也是二维分布曲线。曲线的最高点也是中值点。也是我们取出背景和前景的点。

前景是图像或视频场景中更为可见和突出的部分，是出去背景后留下来的部分，可以看到物体实实在在的出现在图片中。重影区域是指当原始静止物体开始移动时，后静态检测算法可能会错误地检测到原始物体所覆盖的运动区域。一般来说，输入视频图像后，可以根据前景检测算法的需要对图像进行预处理。同样，在前景检测之后，图像也可以根据需要进行处理。

通过提取图像中二维模板中的奇数数据，可以将数据按有序中值排序，而不是按待处理数据进行排序。前景检测的目标是在相对静态的背景之下检测出相对动态的前景目标。但是实际应用场景中因为光照条件的复杂性，摄像机的曝光性能的制约，以及背景并不是理想的静态所以区分出前景和背景是相当困难的事情。详细来说，存在以下几种难题：

背景的变化：包括光线强度的变化、背景中物体的变化、阴影等。场景中光线强度的变化会使得其中的图像也发生变化，可能使得这些变化部分被误检为前景目标。背景中可能存在摇曳的树枝、波动的水面这样类似的背景的位置变化。阴影甚至也可能因为与背景的不相似而被误判为前景目标。

遮挡：场景中可能会出现目标被其他物体全部或者部分遮挡的情况，会对前景的提取造成一定程度的干扰。例如，树木、车辆、路灯、阴影等的遮挡。

前景目标的非时间连续性运动：前景目标有时是最开始就出现在画面中的，而有时又是中途进入画面的。前景目标有时还会静止或者离开画面。当这样的情况发生时，都会给背景建模的质量带来一定的影响。

实时性要求：前景检测是基础步骤，一般的应用当中还需要许多其他后续步骤，所以采用的前景算法要求具有较好的实时性。

## 3.7 人数统计

### 3.7.1 计算像素面积

从前面前景检测结束之后，会出现一些动态变化的像素，如果能将这些变化的像素计算出

来,那么就可以直接检测出流动的物体的总像素,同时检测出人与物后,就可以直接检测出。利用矩形将这些动态的像素标记起来,便于计算改变的像素和。对于不同位置的人或者摄像头发生偏转之后,也可以调整矩形的角度。而像素的总和就是像素变化的面积。也就是说,必须了解像素变化的大小。像素是一个很小的单位,每一个摄像机都会有一定大小的像素,每张图片也会有一定的像素大小,相机的像素改变后,拍摄的照片像素一般也会随之改变。对于数码相机,像素越高代表摄像机的分辨率越高。对于图像,像素越大,代表图像的内存越大,代表图片的内容越多,代表图片存储的信息越大。因此,掌握好像素的变化是计算变化面积的基础,也是为判断是否是一个人的基础。

矩形表示,在计算像素变化轮廓的时候,计算由轮廓弧包围的区域和连接两个端点的弦的总面积。本次设计会用到 `RotatedRect` 类,他是 `opencv` 的一个比较基础的类,依靠调用库函数来计算变化区域面积,通过变化的面积来估计人数的多少。`rotatedRect` 会有一些默认构造函数,会让程序设计出现一些算法。意思是计算像素的时候,会设置一些参数,输入点通常是图像的轮廓点。对于第二个参数,默认值为假,表示轮廓在某个方向上的面积值。字符周围像素的计算。

### 3.7.2 实现计数

可以建立人体模型,通过摄像头的俯视图,检测出人的头部,有效地抵抗遮挡。对于人体检测,通过检测人体各个部位,其创新之处在于在特征分析中增加各种特征,在分类中改进 SVM,或者使用助推分类器,或各种分类器的组合。

在出入口分析中,首先对行人进行检测识别,然后进行跟踪,通过交叉线或轨迹分析,实时检测出当前区域内的行人数量。人形提取已基本完成,接下来就是识别环节了,这个时候,我们还要最后对我们处理的图像做除去噪点的操作。这个时候,需要用 `arcLength()` 函数获取每个轮廓的长度,对于长度比较小的轮廓,将其剔除,不让其参与识别。当然,现在也有非常多的识别方法,最多的就是基于深度学习方式。这些深度学习最大的优点就是识别稳定,识别效果好,准确性高。但是这些深度学习方法也有一些缺点,对硬件的要求高,特别是显卡,不然视频处理速度就很慢。二是深度学习需要大量的训练集,而对于一些特殊的应用,劣势就明显了。再往下说,识别出来后,可以把人给框起来,然后用绿色的点把人中心画出来。而对于人流量的计数,就是要利用这个绿色的点。当用户划一条线后,把线上的值设置为 BGR 为 (0,0,255),再在这条直线上设置一个直线的迭代器,就可以把直线上的点的信息统计下来,当人穿过直线时,绿色的点会穿过直线,这时,直线上部分点的绿色通道将由 0 跳变到 255,通过收集这种上跳沿的信号,就可以实现人流量的计数了。

通过前景检测的效果,对改变的像素进行统计。对边界区域进行统计,满足条件的进行计数。每个人的像素会有一个范围。对于不同距离的人,像素不同。因此,对于不同距离的人进行分别统计。对于像素变化比较大的部分,认为有重叠,对于有重叠的部分,可以将其设为两个人或者三个人。对于模型可以不断更换。当人消失在视野中的时候,可以判定人不在这个公共区域。对于人物的移动,我们对比每一帧图片,不跟踪某一个人,只对这个区域的人计数。

当区域变化帧数大于 800 而小于 3000，可以断定是一个人，当区域帧数 `area` 大于 3000 而小于 5000，判定是两个人。

## 3.8 模型的使用

### 3.8.1 模型的工作原理

模型的作用是来源于高斯背景建模后，要对人物进行处理，而每次进行处理后，图片会再一次更新，因此，模型也会不断的更新。用到的函数是 `createBackgroundSubtractorMOG2`，选的是默认帧数 500 帧，帧数设置越高，检测速度越慢，因此为了切合行人流动速度，选择的默认的 500 帧。方差阈值也选的默认，用于判断当前模型是前景还是背景，默认为 16。因为方差越大，灵敏度越低，对于公共场景，灵敏度效果还是高一点好。这个具体还是看训练效果。

对于是否要检测出影子的问题，模型的工作原理总而言之就是利用一定的帧数去检测出动态物体，从而达到区分出人还是物的工作方法。如果没有出现动态物体的情况，认为没有前景的，全部都为背景。高斯背景算法的一个重要背景特征是为每个像素选择合理的高斯分布，模型是基于高斯背景建模，每一个参数都会对提炼出来的前景效果带来很大影响。

### 3.8.2 模型的初始化方法

一个优秀的模型是需要不断更新的，而对于模型的初始化，是从第一张图片开始。首先利用一系列原样大小的图片，结合空间分布特性，将相邻的动态变化像素点结合起来。优点是可以处理突发情况，缺点是容易引起拖影区域。对于刚刚进入屏幕区的物体，我们会舍弃掉相同区域之前的部门，以此次图片作为初始对象，将此次物体作为初始值。因此，模型的初始化灵活多变。

### 3.8.3 模型的更新策略

模型的更新伴随着旧模型的消失和新模型的建立，对于此次更换过程的方式有很多种，包括无记忆更新策略，时间取样更新策略，空间领域更新策略。对于旧模型的消失不是简单的 `delete`，也不是简单的 `clear` 操作，模型的更新以及新的背景对象的建立跟以往的代替过程是不同的。会用到腐蚀，会计算像素点的梯度以设定一个程度判断需不需要背景更新。腐蚀从字面上理解就知道是对图像缩小的操作，腐蚀了就意味着图像中物体的边界被侵蚀了，轮廓向内收缩，体积变小了。

要注意的是这里的膨胀和腐蚀的概念是针对图像中高亮区域而言的（二值图像中对应像素值 255，白色），变大和缩小也是针对图像中高亮的部分。所以对图像执行膨胀，对高亮区域是胀大，面积增大，对暗区域相当于是腐蚀，面积缩小。

### 3.9 小结

这次设计主要功能实现在于图像处理，也就是说主要的时软件部分，同时软件设计部分也是花的时间最多的一部分，是花精力最多的一个部分，也是最重要的一个部分。硬件需要的东西不是特别复杂，摄像头，处理器够用就行。对于软件部分花比较多的时间是指针，结构体以及算法部分。一开始想将图片通过链表的形式存起来，希望存起来能够很方便，而且方便找被赋值的对象，但后来发现查找图片还是太麻烦，每次需要从头开始，这样对比图片太过于费劲。由于一开始接触的 C++ 比较少，有些指针对象没有释放导致了电脑死机，也一直没有找出问题，甚至后来考虑过换代码。因此，对于指针需要重新看定义。首先指针变量是一个变量，对象也可以看成变量，他们之间的共同点是什么？他们都是一段内存，该内存上存着另一个变量的内存空间。而结构体也是一样的内存分块，内部其实没什么不同，只不过多几个变量的内存而已。通过这种思维方式，解决了很多关于指针的 bug。因此，对于每个问题，都会重头开始看定义。

对于是否用高斯算法来建模我一直在犹豫，在对比效果。虽然帧差法有一定的时代了，但最终模拟出的效果还是没有高斯算法好。所以为了更加准确，虽然时间复杂度高了一点，但一般来说，时间上还是能容忍的。其他的算法还有 VIBE 算办法，基于颜色信息的背景建模方法，统计平均法，中值滤波法。对于很多后来的算法，速度都偏慢。

对于此次设计，比较精准的处理了人流量问题。而对于设计过程也是特别详细，对每一个对象的考虑都很到位，不存在内存泄漏的问题，也不存在造成大量的内存浪费，对于算法也是尽量减少时间和空间的占用。对于视频的处理，做到了可以随时检测，实时分析，这也是 opencv 的优点。算法用到了 opencv 自带的高斯算法，效果还可以。但是对于人物的检测，做的还不够精准，用像素来判断一个人的效果不是太好，对于人物重叠的人流量检测，做到不是很到位，需要进一步改进，对于深度学习案例，做的还不够。

对于框架设计，还是比较直观的，能够比较清楚的看到那个是高斯背景建模的效果，哪个是人流量检测的结果。以及人物可以比较精准的刻画出来。目前存在的问题主要包括如何让代码稳定的在程序上运行，每个厂商都无法保证程序不会因为某个故障导致电脑奔溃。C++ 的稳定性要强于 Java，所以用的 C++ 来写这个程序。本程序对硬件组装和视频编辑功能的双实现难度比较大。是先用 vs2017 先对程序基本框架实现出来，建立多个功能的窗口，对每个窗口进行分析。选择工具都会初始化，包括一些线，圆，矩形。空白区用来充当绘图的地方，每画完一个形状都会对画笔进行刷新，以保证画面的连续性。利用 settimer 对时钟函数进行控制。每画完都会保存。利用粒子算法对图画进行去模糊步骤，以保证画出来的东西能够清晰的展现。背景多样性，功能切换实时性，代码运行高效性，简洁性是本次设计最基础的。





## 第四章 实验与测试

### 4.1 所需硬件设备

本次设计需要用到摄像头，而摄像头用的是手机拍摄的，也可以用电脑直接带的摄像头，可以直接调用，也可以直接处理下载的视频。对于 cpu，用的是 Inter(R)Core(TM)i5-6200U,64 位操作系统。I5 6200u 采用的是 3M 的高速缓存，14nm 工艺制作的，热功耗只有 25W 的处理器，它的主频为 2.3G，最高睿频可以达到 2.8G。在对于内存的支持上，它可以最大支付 32G 的内存，最大内存频率支持 DDR42133MHz。其集成显卡是英特尔最新的高清显卡 520 显示芯片。

### 4.2 软件调用硬件

#### 4.2.1 摄像头驱动

同时 Index 设置 0。因为开启摄像头后一直在进行读取，所以需要等 waitKey()返回值判断退出预览。如果只是读取摄像头不进行处理的话，那毫无意义，现在以 canny 边缘检测以及颜色空间转换为目标进行处理。作为一个对公共场所处理类型的系统设计，调用摄像头是关键的一步操作，不管用到哪个公司的产品，都必须匹配和调用。硬件上的资源都是由操作来管理分配的。因为在内核中，很多的驱动或者别的一些应用程序可能会公用一个资源，这样会导致发生资源冲突。这个时候需要一个管理人员，来分配所有的资源。不管硬件如何调用摄像头，首先第一步，申请资源，软件可以消耗所有硬件资源。进行映射，映射后，返回虚拟地址，这个虚拟地址是 MMU 要是映射的，以后操控硬件都是使用这个虚拟地址。最后函数推出后需要释放相应的资源。

#### 4.2.2 API 的调用

opencv 会常用一些 API 函数，此次用了 API 函数，包括 waitkey 函数，putText 函数，视频的获取以及摄像头的打开等这些都是 opencv 的 API 函数。对于 opencv 函数的调用，首先 opencv 函数对于一般库函数是不一样的，opencv 有很多独特的功能，不能将函数名以及用法等同于其他类型的库。首先这次设计的案例，有很多的接口调用，主要通过继承一些基类来实现自己想要的功能。每一个类都必须定义个对象，以此来对函数接口的调用。而也有一些接口被隐藏了，只能通过继承来调用所需要的函数。在本设计中，采用了简单的视频采集、图像和视频编解码、用户界面功能的简单接口模块；同时包含了基本的数据结构、Mat 类和其他模块调用的一些基本功能；编译 VS 时，它们都是以函数调用的方式实现的。对于每一个 api 函数，都经过了函数的对比以及参数的设计，将参数合理化设置是对此次设计的负责。对于硬件函数的调用，首先需要用到 VideoCapture 类，其功能主要是对视频的读取以及摄像头的调用，

## 4.2.2 功能模块的分析

软件部分如果没有 opencv2 以上是做不出来的，而且不会出现高斯背景建模这个库。配置环境花了很长时间，基本前前后后花了一个月的时间，主要是没配置过，而且不知效果怎么样，网上找的版本太老，对应不上。vs 也需要一定的版本，其中 vs 越高版本，那么对应的 opencv 库用起来就越智能，好多东西不需要自己去定义。硬件部分要求不高，处理器用到 i5 以上，跑起来就没问题了。一般 4 核也够用。对于摄像头，找了好久，一般像素太差的摄像头做出来的效果也太差，高清点的摄像头由于效果比较好，究其原因就是像素变化范围查的仔细，变化幅度算得精确。其次，利用 usb 线链接摄像头比较方便。也可以发射 WiFi 进行组网，不过这个需要摄像头有发射 WiFi 的芯片。如果厂家没有的话需要自己添加 WiFi 模块到摄像头上。

## 4.3 算法设计预期

论文预期结果会得到一个二维的或者三维的热度图来表示人群数量，回归人群分布图，即输入图像，得到人群分布图。采用脉冲函数卷积高斯核的方式来定义密度图。利用卷积神经网络模型可以得到如下所示的结论：

训练流程图：首先采取图像，对于无遮挡的场景，选取  $3m \times 3m$  的图像块进行卷积神经网络模型化，得到人群密度图。对于有遮挡的场景，利用预选场景作为对比，预选场景图片应该是没人的时候，对不同年纪的人进行分析，得到预期的人群密度图，同样对图片进行卷积神经网络模型化，得到预期的人群密度图，如图 4.1 卷积神经网络模型图所示。

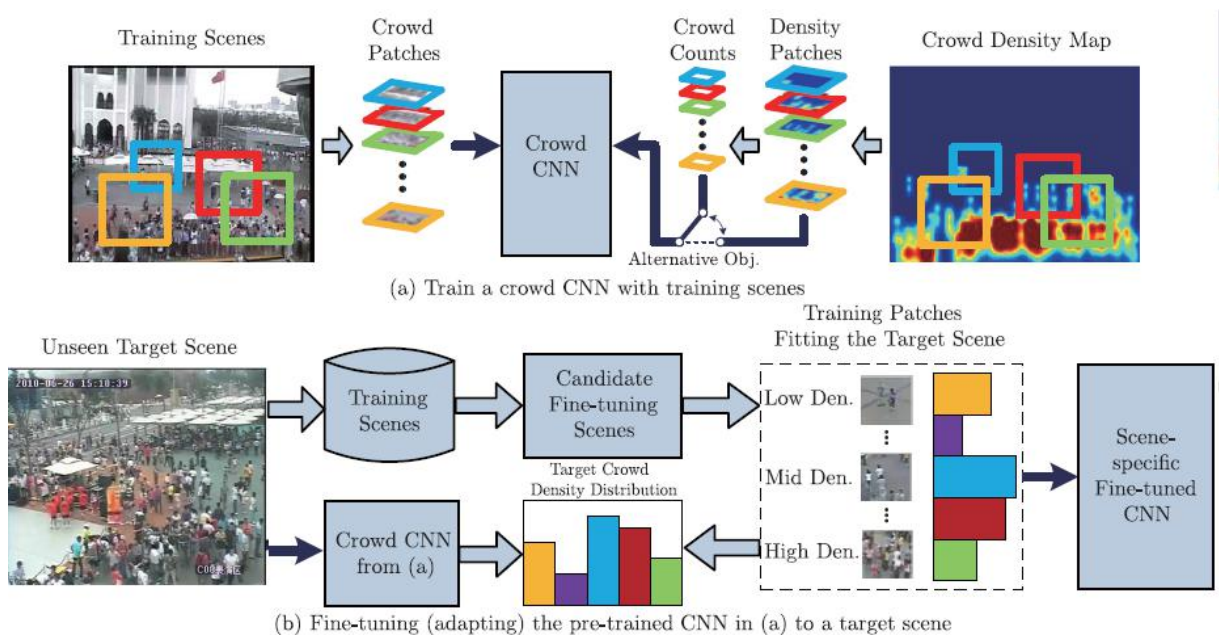


图 4.1 卷积神经网络模型图

预测网络结构, 如图 4.2 预测网络结构图所示

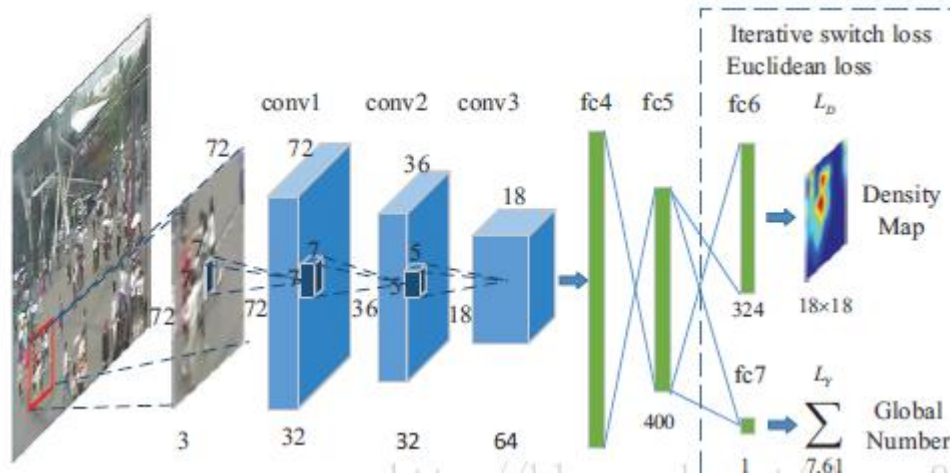


图 4.2 预测网络结构图

即把 CNN 网络结构分为三个卷积层和三个连接层，第一个卷积层有 32 个  $7 \times 7 \times 3$  滤波器，第二个卷积层有 32 个  $7 \times 7 \times 32$  滤波器，第三个卷积层有 64 个  $5 \times 5 \times 32$  滤波器，以此提高精度。

预测结果见表 4—1UCSD dataset 和表 4—2 UCF CC 50 dataset

表 4-1 UCSD dataset

Method	MAE	MSE
Kernel Ridge Regression[1]	2.06	7.45
Ridge Regression[6]	2.25	7.82
Gaussian Process Regression[4]	2.24	7.97
Cumulative Attribute Regression[5]	2.07	6.86
Our Crowd CNN Model	1.60	3.31

表 4-2 UCF CC 50 dataset

Method	MAE	MSE
Rodriguez et al[22]	655.7	697.8
Lempitsky et al.[12]	493.4	487.1
Idrees et al.[8]	468.0	590.3
Our Crowd CNN Model	467.0	498.5

## 4.4 高斯背景建模结果

本次毕业设计的重点在于背景建模，运动中的背景随着场景的不断变化而变化。本次设计得到最初的背景如下图 4.3 背景模型图所示

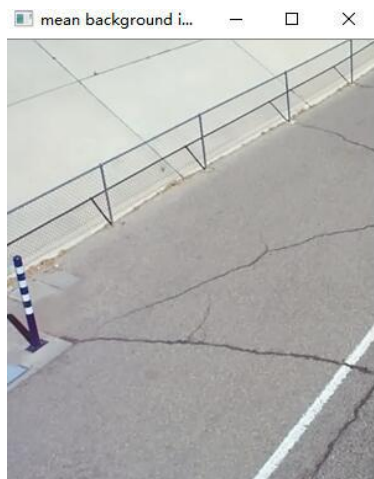


图 4.3 背景模型

随着人流量的增加，需要不断的统计人数，每次统计都是基于此次背景。

## 4.5 前景检测结果

背景建立后就开始分析人的数量以及人在背景中的变化。最开始可以直接选中在运动中的人，只要有物体出现在背景中都可以检测出来。如下图 4.4 单人检测图所示：



图 4.4 单人检测

对于多个人物同时出现在背景中一样可以检测出来，如下图 4.5 人群前景检测图所示：

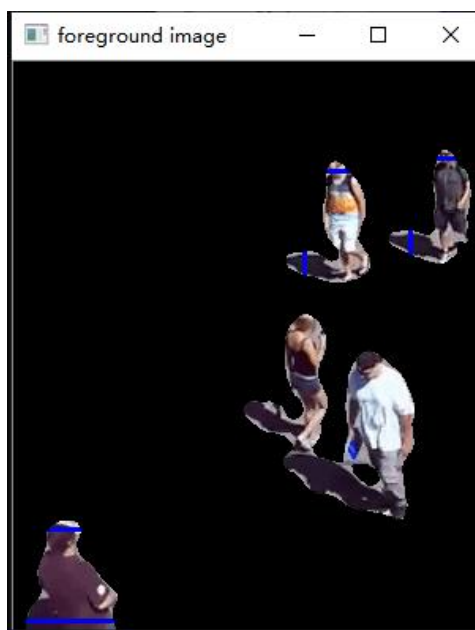


图 4.5 人群前景检测图

对于前景检测图的分析可以分割前景中的一个人来看，结果如下图 4.6 前景检测人物分割所示：



图 4.6 前景检测人物分割

对于前景像素的变化，较为规范的方式是做黑化处理，将背景全部黑化，只留下人和影子的图像，然后利用轮廓检测技术，人的部分全部抠出来，在利用帧差技术，将像素的变化值求出来，计算出人的数量。一个人的效果如图 4.7 单个人的前景处理所示：



图 4.7 单个人的前景处理

对于多个人的效果如图 4.8 多人前景处理所示：



图 4.8 多人前景处理

## 4.6 人与物的智能化判断

对于前景的变化，必须要区分是否是人，已到达此次设计的智能化。因此对于没有人出现在背景中时，可以判断是否有人，其结果如下图 4.9 人物判别所示。

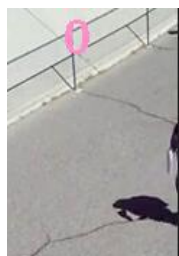


图 4.9 人物判别

很明显可以区分出来这个地方并没有一个完整的人出来，所以没有把影子计算在内，毕竟影子只能算是物。

## 4.7 算法的复杂性判断

高斯模型算法适用于对图像中人数的判断，其公式如下（4.1）所示：

$$f(x) = \frac{1}{\sqrt{2\pi}\sigma} \exp\left(-\frac{(x - \mu)^2}{2\sigma^2}\right) \quad (4.1)$$

随着  $x$  越接近中间值，得出来的效果越好，也就是说，效果得益于一个概率，因为概率越大，表明这是前景的可能性越大。主要的算法是否太过于复杂可以看进程内存和 **cpu** 所占比。内存包括虚拟内存和物理内存。用鼠标在电脑任务栏点右键，再点任务管理器,再选性能一项就可知道电脑内存使用率，**cpu** 使用率表示 **cpu** 的使用情况.进程使用率表示内存的使用情况。对于进程内存，一直占用 45 个百分点，如图 4.10 进程内存显示。



图 4.10 进程内存

对于 **cpu**，一直占用 36 个百分点。因此这次模拟所用的算法时间度比较合适，内存占用也不高，比较适合于人流量检测。



图 4.11 **cpu** 占用百分比





## 第五章 总结

随着人类的需求日益变大和社会发展的大幅度进步，视频处理技术也在快速发展，在现实生活中人流量检测占有着重要的地位。本系统的设计，全面系统地了解了 opencv 的知识和数据库的应用。对于其他软件，如 matlab，视频不能实时处理，所以站在巨人的肩膀上是省时省力的。C++语言编程，程序编写后，可以看出仿真效果可以更准确地计算人流量的数量。opencv 库不断更新着。考虑到现代文件和图像功能的巨大需求以及不同版本的计算机系统，传统的设计方法有明显的缺陷，软件闪烁可能随时发生。因此，这次设计理念是在每次编辑后自动备份数据，而且对于不该有的指针已及时释放掉了，为了去保证不该有指针指向错误。虽然有些框架和算法耗费了大量的资源，但是它的安全性得到了保证。现在的代码太重复，浪费资源严重，本设计尽量减少代码的重复，封装性好。因为这样可以保证程序功能的特殊性，但这种代码为了实现用户的方便，将两者结合起来，让用户选择灵活。因此，该程序设计在功能和适用性上比其他程序都要优越。这个毕业设计得到了从理论到实践的考验。

本文最初就选定好了设计的步骤，一步步按照计划和效果去实现。计划中也会尝试去改留下的问题：如何去解决人物重叠导致计数不成功。本来这个问题在摄像头领域是解决不了的，摄像头根本看不到人背后的东西，透视的话涉及到红外线或者热量图，这个不是视频能够处理的了的。后来提出的解决办法是像素重叠，计算每个人的像素点由多少个组成，实际情况中，前面的人不可能把一个人完全遮住，一般是可以漏半个身子的，这个解决方案可以比较灵活的处理人像重叠的问题。

从数据采集、软件开发和视频图像处理方面的问题，是需要一步步去实现的。在整个过程中，图像的轮廓检测以及前景检测是最为重要的两个处理图像技术。也是后续处理图像的基础。通过前景检测识别出人后，本文再次利用人头检测算法处理行人，达到人物区分的效果。对于区分后的物体再次进行跟踪，每次得到的前景都会与背景进行比较，每次比较都会重新标记行人。而为了更快的检测出前景，本文每个一定的帧数就会进行模型的更新，每次更新都会对前一次模型进行替换，为后续检测带来方便。

人流量检测研究存在很多不足，选择最需要从深度学习领域去解决人数过多的问题，而且很多公司只做出了撞针统计，而不是人流量统计。因此，行人统计还有一段很长的路要走。



## 致谢

本人就读期间孜孜不倦的学习，虽然没有获得过国家级奖状，但省级奖状，校级，院级奖状还是有不少的。对于奖状的获得感言很多，也离不开一些老师和同学的帮助。我还曾经利用课余时间在校外考了一些证书。总共过得过的奖状为全国物联网设计大赛华中地区二等奖，中级网络工程师职业认证，云计算工程师认证，全国大学生数学建模获奖证书，优秀班干部奖，院级演讲比赛三等奖。

特别感谢在此期间给我帮助过的老师和同学。回想四年，时间如流水匆匆，转眼间就将大学毕业，春梦秋云，似乎还记得还停留在大一。四年的大学生活即将画上一个句号，但对于我的人生只是一个逗号，我将面临一个新的旅程的开始。本研究 and 论文是在导师徐国庆的悉心照料和耐心指导下完成的。我的导师徐国庆老师治学严谨，视野开阔。您为我创造了良好的学术氛围，使我的论文更加严谨。

同时，我也要感谢和我一起完成论文的小伙伴们。没有你们的支持和投入的帮助，我无法解决这些困难和疑问，最终使论文顺利完成。从选题之初到论文答辩的顺利进行，有无数可敬的老师、朋友给了我很大的帮助，在此请接受我诚挚的感谢！

最开始写了一些代码，但这些代码效果太差，而且会有很多 bug，别人怎么做我就怎么做，不知谁是对的。因为不知道的东西太多，所以对于别人的代码不知如何去分析。不可能别人和你一样的例子和你一样的代码，乃至和你一样的 bug，这是不可能的。所以你必须去甄别，去知道怎么取舍，这是个很困难的过程。有可能一下午你才解决一个 bug，但不知道为什么这样解决。有可能这个例子你能看懂，别人一些功能函数那么用时对的，但你用的就是不对，有可能别人知道槽函数可以那么定义，但你引用参数就是会报错。有可能别人的类可以那么写，但你不知道版本已经改了，你写上去就是不对。有可能你为了让自己的格式正确，你不断的转换字符，却发现字符总是出现乱码。有可能你写一个树，想读取文本中的文件，但最初打不开文本，别人却可以打开，你费劲一天时间找到打开文本的问题后，你发现识别不了文字，你识别的字是空的等等。历经了几个月的奋战，终于一个 bug 一个 bug 的解决了，一个问题一个问题的找到原因所在了，此时此刻，真的感觉到了荣誉感。

再次要由衷地感谢我的导师徐国庆老师，本科期间得到了很多徐老师的帮助与指导，他严谨的教学态度和平易近人的处事方式深深地影响着我，使我不断成长和进步。在课题开题之前，徐老师给了很多意见和建议，让我明确了自己的研究方向。从开题、研究工作的开展、论文进度、初稿、修改到定稿等一系列流程，徐老师都给了我很多中肯的意见，开拓了我的思路并在我止步不前的时候帮助我一起攻坚克难，最终在老师的指导和帮助下完成了本次课题的研究。攻读本科期间能够遇到这样一位好老师，我受益匪浅、深感荣幸。

我要再次向关心和帮助我完成论文的同学和朋友们表示衷心的感谢。离开学校的日子一天天临近，毕业论文的完成也即将结束。从一开始到论文的顺利完成，离不开老师、同学和朋友们的热情帮助。请接受我诚挚的谢意。

我也要感谢本文引用的学者们的专著。没有这些学者研究成果的启发和帮助，我无法完成这篇论文的最终写作。

我要感谢我的父母。在论文即将完成之际，我的心情无法平静，从一开始就有多少可敬的老师、同学、朋友给了我无言的帮助，在此请接受我诚挚的谢意！感谢母校武汉工程大学，在四年里，我学到了很多在学习和思考。

人生的道路，一定要努力，一定要坚强，捡起来容易，放弃困难。在这里我要感谢我的导师，工作非常细致，文献的收集，框架的设计，程序的调试，和论文的最终稿，从格式到内容，从标点符号头衔，都一点点嘱咐。没有老师的悉心指导，不断督促我前进，我的论文就不会顺利完成。

由于我的学术水平有限，我的论文难免有瑕疵。请评委老师批评指正。

## 参考文献

- [1]周晓冰, 张永领.大型社会活动拥挤踩踏事故机理分析及应对策略研究[J].灾害学, 2015, 30(04):156-162.
- [2] R.Cucchiara, C.Grana, M.Piccardi, A.Prati, Detecting moving objects, ghosts, and shadows in video streams.[J]IEEE Transactions on Pattern Analysis and Machine Intelligence 25 (10) (2003) 1337-1342.
- [3]吴博, 李景文.基于 R FID 和 GRNN 的景区人流量预测 GIS 的研究与设计[J].测绘与空间地理信息, 2015(10):73-75.
- [4]唐娟娟, 龙赢,翟炜街道空间品质的测度、变化评价与影响因素识别—基于大规模多时相街景图片的分析[J]新建筑, 2016, (5): 110-115.
- [5]乐勇.基于监控视频动态信息分析的人流密度检测算法的研究[D].南京邮电大学, 2015.
- [6]尔惟, 贾梦圆, 陈天.城市中心区开放空间网络体系建构研究[J].城市, 2015,28(01):63-67.
- [7]胡成, 郭婧婷, 李强, 陈晋.开放性公共场所人群聚集风险和关键点评估方法[J].中国安全科学学报, 2015,25(12):164-169.
- [8]Henriques JF, Caseiro R, Martins P, et al.High-speed tracking with kernelized correlation filters.IEEE Transactions on Pattern Analysis and Machine Intelligence,2015, 37(3): 583-596. [doi: 10.1109/TPAML2014.2345390].
- [9]Wu HL, Li WH. Robust online multi-object tracking based on KCF trackers and reassignment. Proceedings of the 2016 TF:FF Crlobal Conference on Signal and Processing (GlobalSIP).Washington, DC,Information JSA.2016 124-128.
- [10]Liu W,Anguelov D, Erhan D, et al.SSD:Single shot MultiBox detector. Proceedings of the 14th European Conference on Computer Vision-ECCV Amsterdam,The Netherlands. 2016. 21-37.
- [11]Goodfellow Ian, Pouget-Abadie J, Mirza M, et al. Generative adversarial nets[C] Advances in Neural Information Processing Systems. 2014: 2672-2680.
- [12]Denton E L, Chintala S, Fergus R. Deep Generative Image Models using a Laplacian Pyramid of Adversarial Networks[C] Advances in neural information processing systems. 2015: 1486-1494.
- [13]Radford A,Metz L,Chintala S.Unsupervised representation learning with deep convolutional generative adversarial networks[J].arXiv preprint arXiv:1511.06434, 2015.
- [14]石峰,楼文高.基于灰狼群智能最优化的神经网络 PM<sub>2.5</sub> 浓度预测[J].计算机应用.2017,10.
- [15]杨华,苏航,郑世宝.大规模群体密度估计算法[J].电视技术.2010,5-30
- [16]许少华,何新贵,李盼池;自组织过程神经网络及其应用研究[J].计算机研究与发展;2003,11-20
- [17]项云飞,康志宏,郝伟俊.基于线性回归与神经网络的储层参数预测复合方法[J].科学技术与工程.2017,31-38
- [18]邓春华.群体目标识别与分析技术研究[D].华中科技大学;2016 年
- [19]刘志刚,许少华,李盼池.基于 ELM 和连续过程神经网络的抽油机工况诊断[J].计算机工程与科学,2017,10-14

[20]Sean O'Callaghan<sup>1,2</sup>, David P De Souza<sup>1,2,a</sup> Python toolkit for processing of gas chromatography-mass spectrometry (GC-MS) data.[J]Application and comparative study of selected tools. BMC Bioinformatics 2012 13:115-119

## 在校期间取得的成果

1. 2018 年 5 月-2018 年 10 月，全国大学生物联网设计大赛华中地区二等奖（队长）
2. 2017 年 9 月，获得华为中级网络工程师证书
3. 2019 年 4 月，获得云计算 Na 证书
4. 2017 年 9 月，获得全国大学生数学建模奖（队长）