

密级： 保密期限：

北京邮电大学

硕士学位论文



题目： 基于参与式感知的空气质量检测系统的
研究与实现

学 号： 2012111487

姓 名： 刘肖阳

专 业： 通信与信息系统

导 师： 王文东

学 院： 网络技术研究院

2014 年 12 月 29 日

独创性（或创新性）声明

本人声明所呈交的论文是本人在导师指导下进行的研究工作及取得的研究成果。尽我所知，除了文中特别加以标注和致谢中所罗列的内容以外，论文中不包含其他人已经发表或撰写过的研究成果，也不包含为获得北京邮电大学或其他教育机构的学位或证书而使用过的材料。与我一同工作的同志对本研究所做的任何贡献均已在论文中作了明确的说明并表示了谢意。

申请学位论文与资料若有不实之处，本人承担一切相关责任。

本人签名：_____ 日期：_____

关于论文使用授权的说明

学位论文作者完全了解北京邮电大学有关保留和使用学位论文的规定，即：研究生在校攻读学位期间论文工作的知识产权单位属北京邮电大学。学校有权保留并向国家有关部门或机构送交论文的复印件和磁盘，允许学位论文被查阅和借阅；学校可以公布学位论文的全部或部分内容，可以允许采用影印、缩印或其它复制手段保存、汇编学位论文。

本学位论文不属于保密范围，适用本授权书。

本人签名：_____ 日期：_____

导师签名：_____ 日期：_____

基于参与式感知的空气质量检测系统的研究与实现

摘 要

如今，全球大气污染日趋严重，雾霾严重地影响着人们的日常生活与生产。而由于目前城市雾霾监测点数量少，密度低，且成本较高。我们对雾霾尚不能形成低成本细粒度的检测。但随着科技的发展，尤其是通信技术的发展与相关硬件水平的提高，及内置大量先进传感器的智能终端的日益普及。人们利用配备较高像素的摄像头，便可以获取到大量的图像信息。但目前学术界尚没有提出一种利用智能手机且不借助外部传感器分析 PM2.5 浓度的方法。相关研究大部分都集中在基于专业传感器的空气质量检测。此外，随着结合参与式感知 (Participatory Sensing) 理论的发展，一些新的应用场景不断被提出。参与式感知系统具有覆盖面广、数据丰富、不需要额外硬件投资等优点，能够向基于移动协作感知网络的应用提供全面、便捷、深度的数据。

因此，本课题提出了结合移动参与式感知的基于图像分析的大气质量检测算法，并实现一个集数据收集，处理，分析的空气质量检测系统，从而可以有效地实现空气质量检测低成本化，细粒度化。本课题主要提出了两种有效的方法使用参与者所收集到的雾霾图像进行图像特征提取与分析，这些图像特征将被用来建立 PM2.5 估计模型，从而进行 PM2.5 浓度估计。为了验证算法有效性，本课题收集了长达 6 个月的 4 个在不同地点使用不同型号手机拍摄的图像数据集，并在其上进行了大量实验分析，以验证算法的可行性及系统性能。

论文首先对空气质量检测及参与式感知的相关理论和技术背景以及系统开发中所使用的开发技术进行介绍。之后详细描述了基于图像分析的 PM2.5 估计方法，并且分别阐述了两种不同的 PM2.5 估计方法的实现原理与异同比较。随后，论文继续介绍了 PM2.5 估计系统的总体设计，架构及模块设计。接下来分别描述了系统客户端与服务器端的详细设计与内部各个子模块的设计与实现；最后，通过在数据集上进行大量的实验对 PM2.5 估计算法进行了实验分析，并且对最终系统进行了系统测试及效果展示，从而表明整个系统的运行结果达到了预期的设计目标。

关键词 空气质量 智能手机 PM2.5 估计 图像处理 参与式感知

RESEARCH AND IMPLEMENTATION OF AIR QUALITY MONITORING SYSTEM BASED ON PARTICIPATORY SENSING

ABSTRACT

Nowadays, air pollution has become one of the most pressing environmental issues in many countries, and it now seriously affects people's daily life. However, existing monitoring systems with PM2.5 stationary sensors are expensive, and have small number and low density. We can not make a low-cost and fine-grained air quality monitoring. With the development of technologies, especially communication technology and smartphones equipped with various sensors. Ordinary people can collect photos using their smartphones, and thus we may use the images captured by smartphones to infer PM2.5 concentration without any peripheral sensors. Unfortunately, there is no such method proposed in literatures. Most of the researches focus on utilizing professional sensors to monitor air quality. Besides, with the development of participatory sensing, many new application scenarios are proposed. Participatory sensing is wide-coverage, and can collect richer data with lower hardware cost. So it can provide comprehensive and convenient data for mobile collaboration network based applications.

Therefore, the paper proposed a novel approach to infer air quality using photos from smartphones in participatory sensing, and implemented an air quality monitoring system with data collecting, processing and analyzing to realize a low-cost and fine-grained air quality monitoring system. Two methods are proposed to utilize haze photos collected by participants to extract different haze-related image features and build the corresponding PM2.5 estimation model, which can be further used for PM2.5 estimating. In order to verify the validity of the approach, we conducted extensive experiments over six months

with four datasets to demonstrate the accuracy and energy efficiency of the proposed solution using different models of smartphones in different places.

First, the paper introduces relative theories of air quality monitoring and participatory sensing and techniques used in system implementation. Then the paper describes the image based PM2.5 estimation approach, and presents the design and implementation of two different methods with comparison. The paper then describes the overall design, architecture and module design of the proposed system, and introduces the detailed design and interior module implementation of the client side and server side. Finally, through extensive experiments on the data sets collected, system testing and demos, the paper shows that the whole system has reached the expected goal.

KEY WORDS: air quality; smartphone; PM2.5 estimation; image processing; participatory sensing

目录

第一章	绪论	1
1.1.	课题背景	1
1.2.	课题主要研究内容	2
1.3.	主要工作内容	2
1.4.	论文结构	3
第二章	相关技术与理论	4
2.1.	参与式感知	4
2.2.	空气质量分析技术	5
2.2.1.	基于外部传感器的空气质量分析	5
2.2.2.	基于图像的空气质量分析	6
2.3.	雾霾图像特征分析技术	6
2.3.1.	基于视觉的图像特征	6
2.3.2.	基于物理模型的图像特征	7
2.4.	系统开发技术	9
2.4.1.	图像处理与分析	9
2.4.2.	Android 移动客户端开发	10
2.4.3.	Web 开发技术	11
2.5.	本章小结	12
第三章	基于图像分析的 PM2.5 估计算法	13
3.1.	异构相机图像的归一化处理	13
3.1.1.	相机成像模型	13
3.1.2.	相机辐射标定	13
3.2.	PM2.5 图像特征提取	14
3.2.1.	雾霾模型分析	14
3.2.2.	基于视觉的图像特征提取	16
3.2.3.	基于物理的图像特征提取	17
3.3.	PM2.5 估计模型	18
3.3.1.	基于学习的 PM2.5 估计模型	18
3.3.2.	基于对象的 PM2.5 估计模型	20
3.4.	本章小结	22
第四章	PM2.5 分析系统设计与实现	23
4.1.	概要设计	23
4.1.1.	总体设计	23
4.1.2.	模块设计	24
4.2.	客户端设计	25
4.2.1.	应用层设计与实现	26
4.2.2.	数据层设计与实现	28

4.2.3. 引擎层设计与实现	30
4.3. 服务器端设计	34
4.3.1. 相机归一化模块	34
4.3.2. 气象数据抓取模块	35
4.4. 开发与运行环境	40
4.5. 本章小结	41
第五章 实验分析及系统测试	42
5.1. 实验设计与分析	42
5.1.1. 实验设计	42
5.1.2. 实验结果分析	44
5.2. 系统测试与效果展示	49
5.2.1. 功能测试	50
5.2.2. 性能测试	53
5.3. 本章小结	54
第六章 总结和展望	55
6.1. 全文总结	55
6.2. 未来工作展望	55
6.3. 研究生期间工作	56
参考文献	58
致 谢	60
作者攻读学位期间发表的学术论文目录	61

第一章 绪论

1.1. 课题背景

长久以来, 空气污染治理问题一直是众多发展中国家亟需解决的重要问题。随着科技的发展与工业化水平的提升, 大量的大气污染物给自然环境和人类健康带来了严重的影响。尤其是近年来, 以 PM2.5 为首要的大气污染物给中国多地造成了持续的雾霾天气。据相关数据显示^[1], 北京在 2014 年上半年的平均 PM2.5 浓度为 $100 \mu\text{g}/\text{m}^3$, 已经远远超过正常的 PM2.5 浓度界限 $35 \mu\text{g}/\text{m}^3$ 。其中, 在严重污染的时候, PM2.5 的浓度已经远远超过了空气质量检测设备的浓度监测范围, 达到了“爆表”的 $755 \mu\text{g}/\text{m}^3$ 的浓度。由此可见空气污染在中国大部分地区已经达到了极为严重的程度。由 PM2.5 所导致的雾霾环境不仅对人们的身体带来了极大的危害, 例如各种呼吸系统及心血管系统疾病, 同时对公共交通安全也产生了不可忽视的影响。因此, 为了有效的治理城市大气污染, 对 PM2.5 的实时监测是一个十分重要的环节。目前, 大多数国家的 PM2.5 浓度检测手段主要依靠大型的检测设备, 其虽然测量结果较为精确。但是此类设备成本较昂贵, 需要经常性的人工维护及保养。此外, 该类设备在城市中的分布密度较低, 无法实现细粒度的实时监测。以北京为例, 在北京市 16411 平方公里的面积中仅仅分布 35 个监测站, 平均每个监测站要覆盖 468 平方公里, 相当于一个海淀区的面积总和。因此, 如何低成本细粒度地实时监测空气质量成为了空气质量检测的中需要解决的重要课题。

另一方面, 随着智能手机的性能大幅提升及其方便携带等特点, 使得依靠内置多种传感器的智能手机的感知, 存储及计算能力进行大范围数据采集, 存储, 分析计算成为了可能。参与式感知^{[2] [3]}是近年来所提出的利用普通民众的智能设备去收集并分享其周围的环境信息的新方法。例如, 利用智能手机的麦克风, 我们可以去收集环境的噪音信息。利用 GPS 传感器, 我们可以收集用户的位置信息, 进而分析人群热度。利用智能手机内置的方向传感器及加速度传感器, 我们可以得到用户的状态信息, 如行走或者跑步。利用摄像头我们可以收集到视频或图像信息。此外, 通过无线或者有线连接, 如蓝牙或者 USB 接口, 我们可以使用外加的传感器设备, 如空气质量检测仪等。相比传统的传感器网络, 参与式感知利用众人的智能设备实现数据的采集, 处理及传输, 其部署成本极低。此外, 智能设备的携带者天然具有分布广泛, 及移动特性。通过外部的人为激励策略, 吸引参与者完成数据收集任务, 使得低成本, 细粒度的数据采集及分析成为可能。

本课题在现有的参与式感知的基础上, 设计并实现了一个基于图像分析的 PM2.5

检测系统。

1.2. 课题主要研究内容

本课题设计并实现一个基于图像分析的 PM2.5 检测系统，主要研究内容包括：异构手机图像的归一化处理、空气质量与图像特征的关系研究及基于图像的空气质量检测系统。

要利用智能手机所拍摄的图像进行空气质量分析，首先需要利用手机所拍摄的相片进行图像处理与分析。但是不同型号的手机摄像头硬件参数的不同，拍摄时焦距，曝光度，白平衡处理及图像分辨率的不同，都会对最终图像处理，分析产生一定的影响，最终使得无法较准确进行 PM2.5 浓度估计。因此，需要首先对手机所拍摄的图像进行归一化处理，使其可以作为 PM2.5 图像分析算法的输入图像。

接下来需要研究的重点内容是空气质量与图像特征的关系研究。要想从图像分析中去量化 PM2.5 的浓度，首先要研究 PM2.5 浓度与图像特征之间的关系。针对该问题的研究在学术界较少，大部分研究主要集中在分析大气能见度或其它一些大气光学参数与图像特征的关系。本课题主要从图像的视觉特征及物理特征着手。视觉特征主要描述了图像的色彩，锐度，对比度等特征。物理特征主要从一些现有的天空模型，大气模型等进行分析。本课题将利用不同地点，不同手机所拍摄的图像，使用以上图像特征来分析其与 PM2.5 浓度的相关程度。更进一步地，分析出哪些图像特征相比其它特征与 PM2.5 浓度具有更高的相关特性，且拥有较高的计算效率，。

最后，本课题依据利用研究得到异构手机图像归一化模型，PM2.5 图像特征提取算法，实现了一个基于图像的空气质量检测系统。该系统首先根据参与者的手机型号分析得到其相机的成像模型。随后利用参与者在同一地点拍摄的若干天的图像进行图像预处理分析。然后利用通过图像处理及分析算法得到雾霾图像特征，最终使用得到的图像特征，及参考 PM2.5 浓度值，从而得到该地点的基于图像的 PM2.5 估计模型。进而使得该参与者通过在同一地点拍摄相片而分析其相应的 PM2.5 浓度。该系统主要分为两个部分，智能手机客户端主要负责图像及相关数据采集，处理与 PM2.5 分析。服务器端主要负责 PM2.5 数据的存储及气象数据的抓取与存储。

1.3. 主要工作内容

本课题研究中，论文作者主要负责的工作内容如下：

1. 负责雾霾图像分析算法的调研和可行性分析
2. 负责 PM2.5 图像数据收集及相关实验
3. 负责结合参与式感知的空气质量检测系统的需求分析

4. 负责基于图像的空气质量检测系统客户端的架构设计和详细设计
5. 负责服务器端相机归一化模型及气象数据的采集及存储工作
6. 负责客户端图像处理分析核心算法的实现与优化
7. 负责基于图像的空气质量检测系统的部署和测试

1.4. 论文结构

本论文的结构安排如下：

- | | |
|-----|---|
| 第一章 | 绪论。介绍了本论文的课题背景、主要研究内容及工作内容等。 |
| 第二章 | 相关技术与理论。主要介绍了参与式感知技术，雾霾图像特征提取及相关空气质量分析方法。此外，还介绍了系统开发中所用到的相关技术。 |
| 第三章 | 基于图像分析的 PM2.5 估计算法。首先介绍了通过单一图像实现相机辐射标定的方法。随后，介绍了本文使用具体的图像特征提取算法及相关优化方法。最终，基于该特征提取算法，设计并实现了 PM2.5 的分析建模过程。 |
| 第四章 | PM2.5 分析系统的设计与实现。首先介绍了该系统的总体需求分析及系统设计。随后依次详述了系统客户端和服务端的设计与实现。客户端主要包括了图像及数据预处理模块，雾霾图像特征提取模块，PM2.5 分析建模模块，及 PM2.5 估计模块。服务端主要包括了相机归一化模块与气象数据抓取模块。并最终介绍了系统实际的开发及运行环境。 |
| 第五章 | 实验分析及系统测试。首先，使用本课题收集到的数据集进行了大量的实验，并且对实验结果进行了详细的分析。其次，介绍了 PM2.5 分析系统的测试环境，然后针对各个流程及相应子模块进行系统测试及性能分析。最终，在客户端上演示并检验运行效果。 |
| 第六章 | 总结与展望。对课题的工作进行了总结，指出了 PM2.5 分析系统的不足之处，并给出了进一步的完善的方向，并对未来工作进行了展望。 |

第二章 相关技术与理论

2.1. 参与式感知

参与式感知是一种将普通民众的移动设备，如智能手机来组建参与式传感器网络，利用参与者的移动设备收集，分析并分享其周边的数据。通过这样一种方式，我们可以将数据的收集，计算与传输都集中在智能手机上，并使其作为参与式感知网络的一个节点，进行大范围细粒度的环境感知。相比传统的传感器网络，由于所有的感知设备都在参与者的控制之下，所以需要有效并持续的激励策略以保持用户较高的参与度，来完成数据采集任务。随着对参与式感知越来越多的尝试与挖掘，在该领域产生了一些新颖的上层应用。总体来说，参与式感知应用可以分为基于人与基于环境两类^[4]。基于人的应用主要目的用于分析人群的活动及理解人的行为。而基于环境的应用主要用来收集参与者周边的环境参数，例如光照强度，噪声强度，空气污染指数等。而本课题即属于基于环境的参与式感知应用。在该场景下，参与者主要利用智能手机的内置传感器或外部传感器测量环境参数。我们可以利用手机中的麦克风去收集参与者周围的环境噪音等级，从而判断某个地区的噪音污染等级。例如 EarPhone^[5]利用用户智能手机收集环境噪音，从而生成地域噪音分布图，进而分析噪音与人类活动的关系。我们可以利用 GPS 传感器收集用户的方位信息，移动轨迹等。例如 GreenGPS^[6]利用参与者的 GPS 信息，分析两点之间不同路径的油耗信息，从而得到最节省汽油的路径，达到环境保护的目的。我们也可使用光照传感器来收集参与者所处环境的光强信息。例如，SurroundSense^[7]利用光照及其他信息来判断用户所处的环境。此外，通过外接传感器，例如一氧化碳，二氧化硫，二氧化氮等有害气体传感器，来感知用户周边空气污染程度。如 David Hasenfratz 等人^[8]提出的利用可与智能手机相连的手持空气质量传感器收集并共享环境信息，其效果如图 2-1 所示。总而言之，参与式感知给我们提供了一个新的途径，即通过普通民众的智能设备实现大范围细粒度的环境信息的收集，分析与共享。

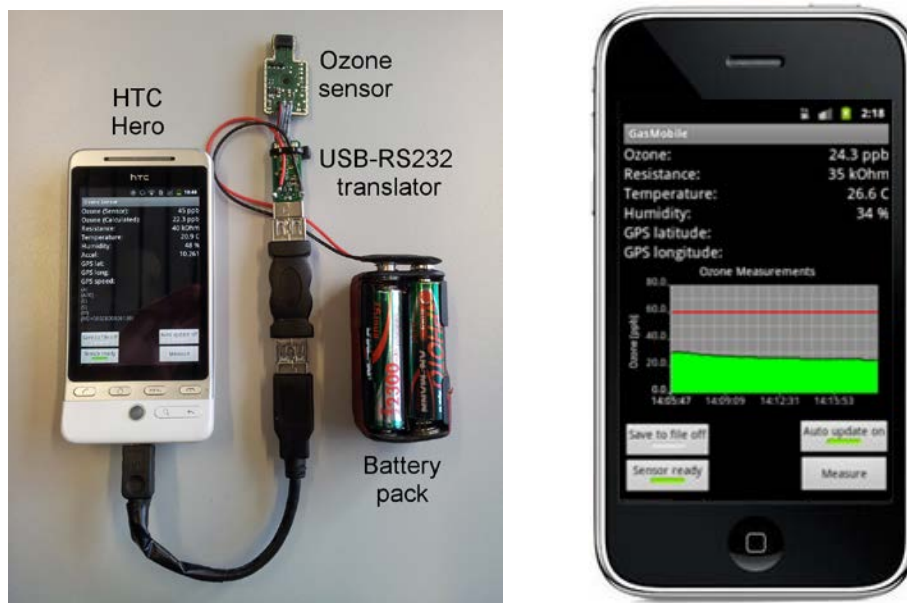


图 2-1 参与式感知下利用智能手机收集周围环境数据

2.2. 空气质量分析技术

利用智能手机实现空气质量分析，主要分为两类方法，其一为基于外部传感器的空气质量分析方法。其二为基于图像的空气质量分析方法。前者在学术界有较为广泛的研究。而后者可供参考的理论及系统实现相对较少。此外，大部分论文通过使用大气能见度^[9]，大气消光系数^[10]，大气浊度^[11]等来作为空气质量的度量。以上若干参量皆与 PM2.5 浓度有着密不可分的关系^{[12][13]}。

2.2.1. 基于外部传感器的空气质量分析

随着传感器设备逐渐的小型化，低成本化。我们可以携带传感器设备感知环境数据。通过自身的无线连接模块或与智能手机相连，便可以实现数据的传输与分享。目前，大部分的研究都集中在利用小型空气质量传感器通过蓝牙或者 USB 与智能手机连接，通过用户随身携带来收集不同区域的空气质量参数，如图 2-2 所示。最终通过移动网络上传至后台服务器。CommenSense^[14]使用手持式的专业传感器设备通过蓝牙和 GPRS 定时传输空气质量数据。虽然其测量结果较为精确，但是其体积及净重过大，不易随身携带。Srinivas Devarakonda 等人^[15]提出使用移动车载空气质量传感器及便携式传感器组成实时移动空气质量传感器网络。其功能繁多，包括有害气体传感器，温度，湿度，光强，大气压强等传感器。但是单一设备成本高昂，不适合普通民众参与数据收集。InAir^[16]使用小型有害气体传感设备采集室内的有害颗粒物浓度，并可以通过智能手机实时查看室内空气环境，及历史数据。但是其设备体积较大，且无法测量室外的空气质量。综上所述，通过外部传感器获取空气质量数据成本较为昂贵，用户参与度不高，并不是一种

有效的大范围收集空气质量数据的方法。

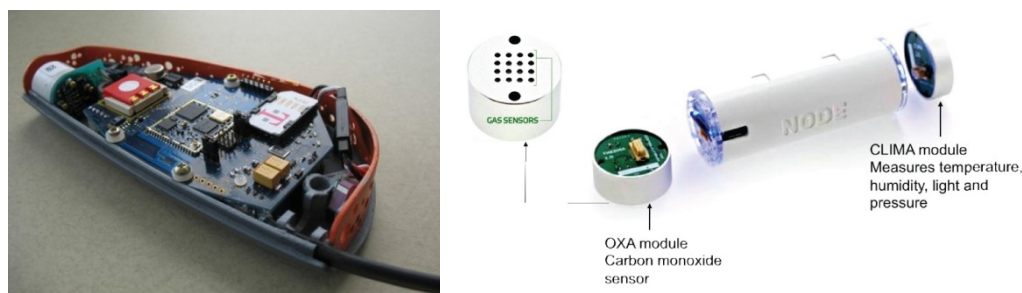


图 2-2 手持式与便携式空气质量检测设备

2.2.2. 基于图像的空气质量管理

基于图像的空气质量管理主要通过提取不同的图像特征从而分析其与空气质量的关系。目前，学术上大部分的研究都集中在利用固定摄像机，如交通摄像头，气象观测专用摄像头等，在某一固定场景长时间收集一段图像数据并分析其与大气能见度，大气光学参数之间的关系。其方法间的不同主要体现在所采用的理论模型，所选取的图像特征及建模方法，具体方法可见第三章。但是，在参与式感知场景下，利用固定摄像机拍摄图像成本上较为昂贵，且无法有效利用智能手机的移动特性与计算能力。

2.3. 雾霾图像特征分析技术

雾霾图像特征分析技术主要目的是通过对图像的分析处理，获取其有效的图像特征，进而计算出所拍摄场景的能见度信息，大气质量信息等。与雾霾相关的图像特征可主要分为两类：基于视觉的图像特征及基于物理模型的图像特征。

2.3.1. 基于视觉的图像特征

在计算机视觉中，图像的视觉特征主要描述的是图像的色彩，形状，纹理等底层特征。而由空气污染所导致的雾霾现象将会对图像的某些视觉特征产生较大的影响。总体上，我们可以将图像视觉特征分为两个部分，空间域特征与频率域特征。

2.3.1.1. 图像空间域特征

分析雾霾图像空间域特征主要从图像的灰度，对比度，梯度，色彩或直方图等统计特征着手。基于灰度的分析主要是利用雾霾对图像整体色彩深度的改变。其算法快速而简单，但是其可靠性较低，受太阳光照等外界因素的影响较大。因此，很少有单纯利用图像灰度分析雾霾图像的算法。图像对比度特征主要是利用雾霾对图像不同区域的颜色

亮度差异的影响。例如 Xie 等^[17]利用固定摄像装置拍摄同一场景，将地平线附近的景物与天空作对比，计算两者的色彩对比度从而分析其与大气消光系数的关系。但其结果并不精确，而且需要人为的预先设定好地平线的位置。图像梯度特征主要利用雾霾对图像边缘锐度减弱的效应，如 Graves 等^[18]利用固定摄像头拍摄城市全景，将图像做简单的分割处理后，分别计算每个图像块的梯度特征。经过长达数月的观测，其发现部分图像块的梯度特征与大气消光系数存在较大的相关度。图像色彩特征主要是利用雾霾对色彩的色调，饱和度等参数的影响。Kim 等^[19]通过观察天空的 HSI^[20]的色彩特征发现天空区域的色彩饱和度与大气消光系数存在较大的相关性，并呈现指数关系特征。此外，还有少量论文利用图像统计特性，如直方图灰度均值，直方图灰度标准差，直方图扭曲度，直方图的熵来分析雾霾对图像的影响。

2.3.1.2. 图像频率域特征

除了图像空间域特征之外，图像的频率域特征也以一定程度的反应雾霾大小的程度。Liaw 等^[21]使用 DFT 变换进行高通滤波，发现图像的高频部分能量与大气消光系数间有较强的相关性。在图像处理中，图像的高频分量代表着图像强度变化最剧烈的地方，如物体的轮廓，边角等。而因空气污染导致的雾霾则会减轻这种图像强度的变化，从而减小高频能量。但是该方法始终无法解决光照对结果的干扰。在其之后的研究中^[22]，通过使用同态滤波及 DWT 小波变换，来减少非均匀光照对算法的影响，从而进一步提升了算法估计准确度。但其仅在单一地点做了长时间的实验与测试，其对相机拍摄的场景依赖性较强，且准确度不高。因此，依赖图像的频率域特征，会极易受到外界因素干扰，算法稳定性较差，无法得到准确的分析结果。

2.3.2. 基于物理模型的图像特征

与图像视觉特征不同，基于物理模型的图像特征并不直接利用图像处理提取的特征与真实空气质量数据做回归分析来统计出相关性最强的视觉特征，而是利用已有的物理成像理论及模型来直接度量空气质量。其主要分为 Perez 天空亮度模型及 Koschmieder 大气成像模型。

2.3.2.1. 天空亮度模型

天空亮度模型是由 Perez 等^[23]发现的天空亮度与太阳角度，任意指定天空角度及大气浊度的关系模型。Poduri 等^[24]提出了利用智能手机拍摄天空图像，进而分析大气浊度，从而估计空气质量。如图 2-3 所示，其通过用户使用手机拍摄天空，并圈画一块矩形天空区域，通过相机的方向传感器，加速度传感器，GPS 传感器，拍摄时间，得到天空角度与太阳光照角度。最终利用 Perez 天空亮度模型得到大气浊度。该方法无需任何回归

分析，利用用户的手机即可粗略的分析出空气质量。但是其需要用户的操作过多，且需要大量传感器信息，并对传感器的误差较为敏感。此外，如果天空受到遮盖，如建筑物，山，云等，将无法进行空气质量分析。

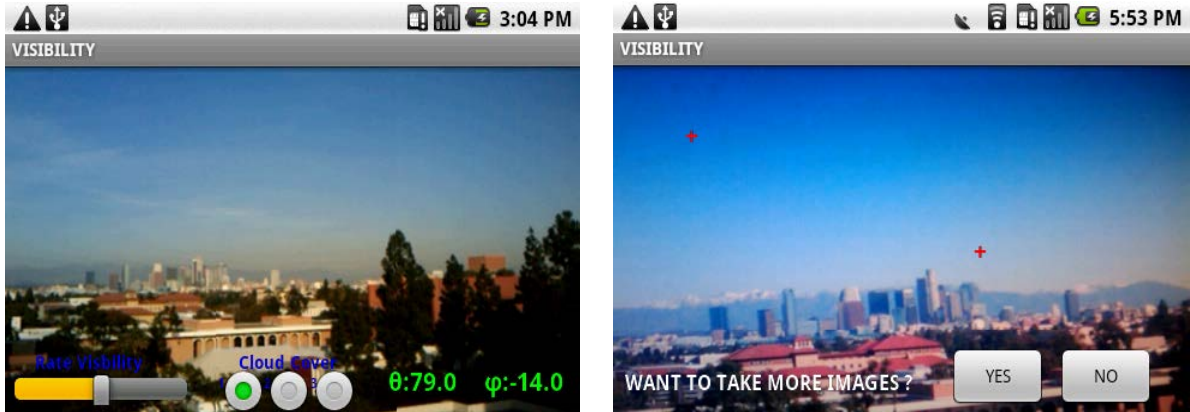


图 2-3 基于智能手机利用天空亮度模型估计大气能见度的方法

2.3.2.2. 大气成像模型

通过利用大气成像模型^[25]，我们可以得到任意物体观测到的辐射强度，实际辐射强度与大气辐射强度及大气传播系数的关系。其中大气传播系数是一个与雾霾程度密切相关的参量。如果已知物体的距离，我们便可以计算得到大气消光系数，进而分析出空气质量。该模型在计算机视觉中的图像去雾领域广为应用。其解决的问题是如何恢复一张有雾的图像为去掉雾霾效果的图像。其中的主要问题即为如何利用大气成像模型，估计出大气传播系数。Fattal 等^[26]提出的基于物体表面阴影与大气传播系数的局部统计部相关性方法及 Tan 等^[27]提出的基于极大化局部对比度的方法虽然可以得到较好的去污效果，但是它们都无法很好地估计出大气传播系数。Johannes 等^[28]提出利用已知的 3D 模型图，及场景深度信息，计算大气传播系数的方法虽然较为准确，但是其需要过多的场景信息，因此难以在实际中应用。Narasimhan 等^[29]提出利用同一场景不同天气的多幅图像，基于场景结构的对比度恢复方法，可以较为有效的恢复大气传播系数，但是其需要同一场景不同天气的多天图像，条件较为苛刻。而近年来，He 等^[30]通过大量的实验与观察，发现了暗原色先验统计，并由此分析出了有雾图像暗通道与大气传播系数的近似关系。因此，可以通过该理论较好分析出大气质量相关参数。

2.4. 系统开发技术

2.4.1. 图像处理与分析

2.4.1.1. Matlab

Matlab 为矩阵实验室，是一种用于实现算法开发，数据分析，数值计算及数据可视化的高级语言及交互环境，并且专门针对现代的处理器和存储结构进行了一定的优化。Matlab 是一种交互系统，其基本的数据元素为矩阵。这可以允许许多技术计算问题的明确表达并求解，特别是涉及矩阵表示的问题，如图像处理等。除此之外，其还可以调用其他的主流语言程序，如 C/C++，Java，Python 等。尽管 Matlab 典型应用为数值计算，但其还提供适用于不同领域的附加工具箱，例如控制系统设计与分析、图像处理、信号处理与通讯、金融建模和分析等。其中 Matlab 图像处理工具箱(Image Processing Toolbox) 提供了大量较为常用的图像处理函数，这些函数按功能可以分为图像分割、特征提取，图像空间变换及图像配准，图像滤波、增强及去模糊，图像变换如傅里叶变换、离散余弦变换等，图像形态学处理，还有一些互动模块化工具，如感兴趣区域选择及直方图测量等。因此，利用 Matlab 以上所提供的大量图像处理函数，我们只需考虑图像分析算法的逻辑实现，而不必受到在底层代码实现的困扰。

2.4.1.2. OpenCV

OpenCV 是一个开源的基于 C/C++实现的计算机视觉库，其程序可以在多种操作系统中运行，如 Windows，Linux，Mac 等操作系统。随着移动设备的逐渐普及，最新版本的 OpenCV 对 Android, IOS 等移动设备操作系统也有了较好的支持。除此之外，OpenCV 还提供了 Matlab, Python 及 Ruby 等其它语言接口。OpenCV 的目标是构建一个简单易用的计算机视觉框架，以更加方便，快捷地开发出复杂的计算机视觉相关应用程序。OpenCV 主体总共分为 5 块，如图 2-4 所示。它们分别为包含基本图像处理函数及高级计算机视觉算法的 CV 模块，包含基于统计的分类和聚类算法的 ML 模块，包含图像及视频输入输出函数的 HighGUI 模块，包含 OpenCV 基础数据结构及相关函数的 CXCore 模块，以及包含一些已淘汰或实验性质的图像算法函数的 CvAux 模块。因此，通过使用 OpenCV 函数库，我们可以轻松的开发出高效的图像分析算法，并在多个平台实现代码移植。

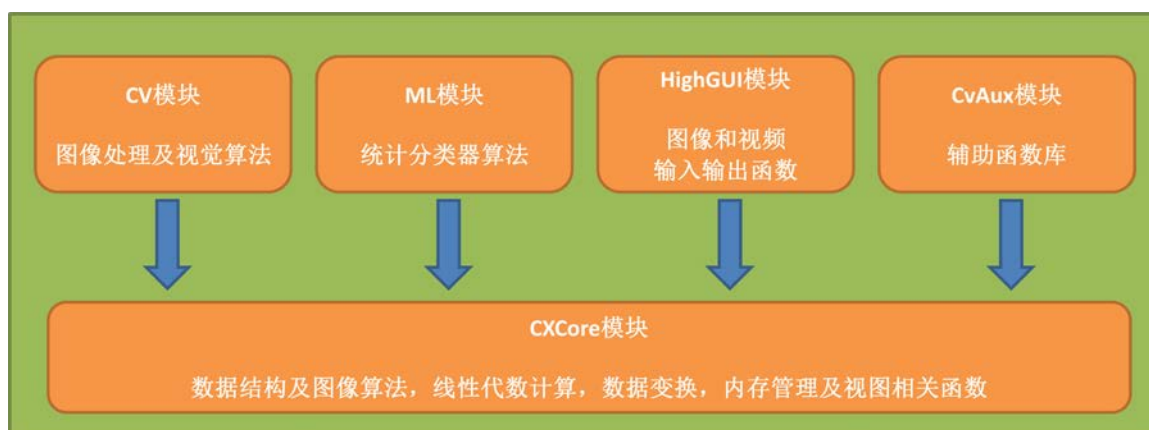


图 2-4 OpenCV 基本模块结构

2.4.2. Android 移动客户端开发

Android 是当前最流行的基于 Linux 操作系统的开源的移动设备操作系统，其主要用于智能手机与平板电脑中，我们可以使用官方提供的基于 Java 的 Android SDK 进行 Android 应用程序开发。Android 系统主要由 Linux 内核层，系统中间件，应用程序框架层及应用程序层组成，如图 2-5 所示。Android 通过如图所示的分层设计使得各层间相互分离，每一层都有自己明确的功能，保证了层级间的低耦合性及系统可扩展性。Android 应用主要包含电话，短信，联系人，浏览器等上层核心应用程序。应用程序框架层主要负责为上层应用程序提供统一的 API 调用接口，例如系统 UI 框架，拨号管理，资源管理，位置管理等框架程序。函数库层主要包含各种 C/C++ 库，如系统使用的为嵌入式系统改良的标准 C 系统库，负责不同音视频格式编解码的媒体影音库，负责提供底层 2D 图形渲染的引擎库，及负责轻量级关系数据库的 SQLite 库等。与函数库处在同一层级的为 Android 运行时环境，包含 Android 核心库集和 Dalvik 虚拟机。Android 核心库集主要为 Java 核心库提供大部分功能。而 Dalvik 是一种适用于较低 CPU 计算速度及少量内存空间的虚拟机，因此它非常适合在移动终端中使用。与传统 JVM 相比，首先 DVM 主要运行其专有的 dex 文件，dex 文件是一种可以将所有 class 文件整合起来，减少其冗余信息，进而提高程序运行性能的文件格式。其次，DVM 是基于寄存器实现的，而非像 JVM 基于栈实现的，其往往在移动终端上拥有更高的性能表现。Linux 内核层主要负责为 Android 系统提供内存管理，进程管理，网络协议栈及驱动软件等底层核心服务。Android 应用程序以 APK (Android Package) 作为其文件格式，其为包含 Dalvik 字节码及资源文件等数据的压缩包。

Android 的 SDK 是基于 Java 的，这意味着所有的第三方应用都使用 Java 作为开发语言。但 Google 提供的 NDK (Native Development Tool) 为我们提供了一种可以使用 C/C++ 开发函数库并与 Java 代码打包的方法。因此我们可以 Java 平台所提供的 JNI (Java Native Interface) 标准使得 Java 代码与其它本地已编译的代码交互，从而是开发者方便

的使用大量已有的 C/C++ 开源库，如 OpenCV 等。同时也可以提高程序性能，及可移植性。

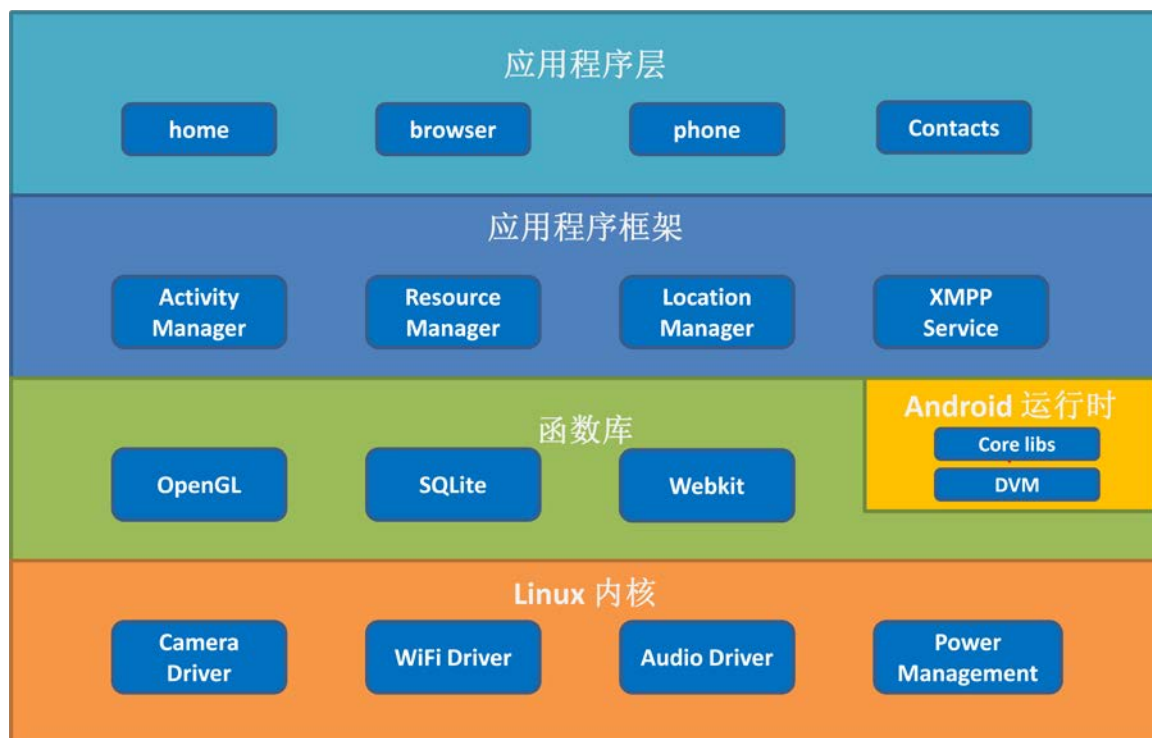


图 2-5 Android 系统体系结构

2.4.3. Web 开发技术

2.4.3.1. XPath

Xpath (XML Path Language) 是一种基于 XML 树状结构，提供在数据树中搜索节点能力的路径语言。Xpath 是 W3C 的一个标准协议，其使用路径表达式在 XML 文档中导航，在 XPath 中，有七种类型的节点，它们分别为元素、属性、文本、命名空间、处理指令、注释以及文档节点。XPath 沿着节点的路径来选取符合条件的节点，其路径表达式主要由一个节点到另一个节点的步骤顺序组成，其中每个步骤包含 3 个部分，轴描述，节点测试及节点描述。因此，我们可以借助 XPath 帮我们在 HTML 页面中快速定位符合条件的 DOM 元素。

2.4.3.2. JSON

JSON (JavaScript Object Notation) 是一种轻量级的数据交换格式。与 XML (eXtensible Markup Language) 相比，JSON 的解析速度更快、可读性更高，且占用空间更少。除此之外，大部分的程序语言都有对 JSON 的支持库，如 Java 中的 Jackson JSON，JSON-lib 等，Android SDK 中也内置了对 JSON 的支持。因此，以上这些特性使得 JSON 成为了理想的数据通信格式。

2.5. 本章小结

本章主要对与 PM2.5 图像分析及实现相关的理论方法与实现技术进行了综述。分别介绍了参与式感知的原理与应用场景，空气质量分析的相关技术与当前的不足，雾霾图像的研究方法和相关计算机视觉理论。此外，还介绍了在系统实现中主要利用的图像处理及统计分析工具，及相关的客户端和服务端开发工具。

第三章 基于图像分析的 PM2.5 估计算法

3.1. 异构相机图像的归一化处理

本课题使用手机拍摄的图像进行 PM2.5 分析，即我们要通过对雾霾图像的处理，分析其中的物理含义。由于手机拍摄的图片是经过相机成像系统对拍摄的真实场景进行的一系列变换的结果。因此，首先我们要将手机所拍摄的图像进行一个逆变换过程，将其恢复为实际场景。为了解决这个问题，我们首先要分析相机成像模型，了解其本质的变换过程。其次，我们需要利用基于单幅图像的恢复算法，将手机所拍摄的图像尽量恢复为原始拍摄的场景。

3.1.1. 相机成像模型

在我们使用手机拍摄照片时，摄像头的光学透镜会将场景的光辐射聚焦到相机的成像平面上，并通过电子或化学光学感应设备将实际的场景辐射转化为图像亮度。相机成像模型主要用来分析场景辐射和图像亮度的映射关系。而这个映射关系往往是十分复杂的，其中包含了大量的复杂的，难以计算得到的因子，例如感光器件的灵敏度系数，成像设备产生的电子噪声参数等等。因此，我们可以将这个关系统一分解为两个过程^[31]，如图 3-1 所示。第一个过程为一个线性变换过程，第二个过程为一个非线性变换过程。这个非线性变换过程也被称为相机相应函数（Camera Response Function）。

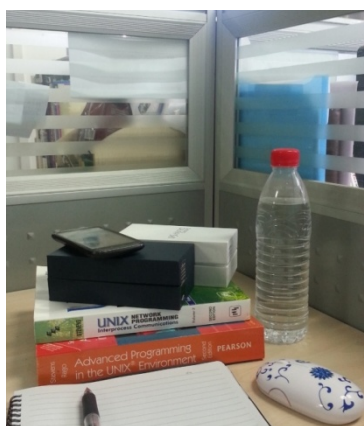


图 3-1 相机成像过程

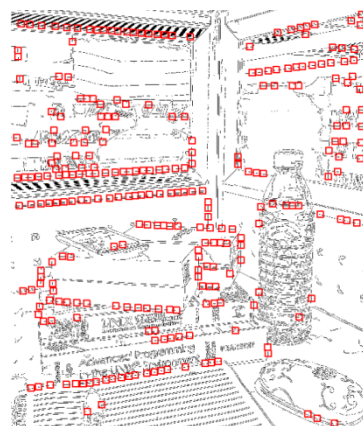
3.1.2. 相机辐射标定

相机辐射标定（Camera Radiometric Calibration）的目的是根据相机成像模型，通过对所拍摄图片的分析，将图像亮度恢复为实际场景辐射。本课题使用了 S.Lin 等人^[32]所提出的利用单幅图像进行辐射标定的方法。与之前 M. D. Grossberg 等人^[31]提出的利用不同曝光度的多幅图像估计方法相比，该算法只需利用一张图像便可以进行相机辐射标定，且无需预先设置相机参数，如曝光度等。其基本思想是分析相机中传感器的非线性作用对图像中边界像素的影响，从而分析其相机响应函数。首先，在图像中检测到足够数量的边界区块，并利用极大后验估计（MAP），其中似然函数与区块边界两侧 RGB 空

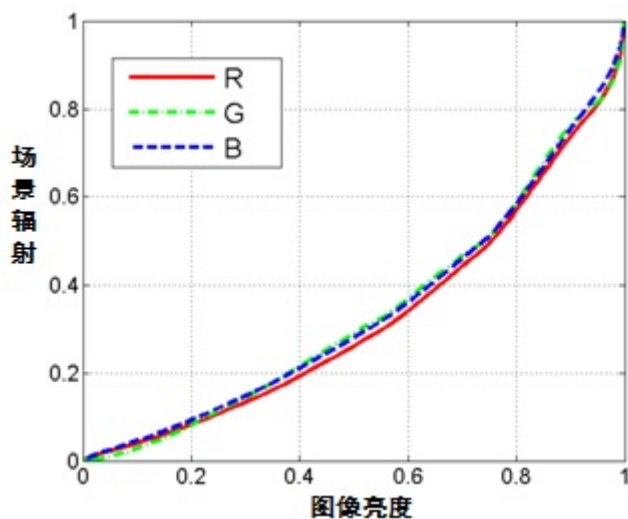
间距离呈负指数关系，最终完成相机辐射标定，得到恢复后的场景辐射。如图 3-2 所示，其中图 3-2(a)表示为利用三星 S3 手机所拍摄的图像，图 3-2(b)表示为算法所检测出来的有效边界区块，图 3-2(c)表示为最终的到相机辐射标定的逆变换函数。



(a) 三星 S3 手机所拍摄的图像



(b) 检测出的有效边界



(c) 相机逆变换响应函数

图 3-2 相机辐射标定过程

3.2. PM2.5 图像特征提取

利用辐射标定后的图像后，首先我们需要分析雾霾对图像会产生的影响，并从中采用不同的图像特征提取方法，通过分析比较，得到与 PM2.5 浓度相关性相对较高的图像特征。

3.2.1. 雾霾模型分析

通过对雾霾天图像的观察，我们可以发现较高的 PM2.5 浓度环境会使图像的质量造

成较大的下降，使得图像变得模糊，发灰。因此，我们可以利用图像质量下降的程度来建立雾霾模型，从而估计 PM2.5 浓度。

理论上，图像与雾霾模型的物理参数间的关系可由一个大气成像模型^[26]表示，其含义可由图 3-3 及式(3-1)给出。

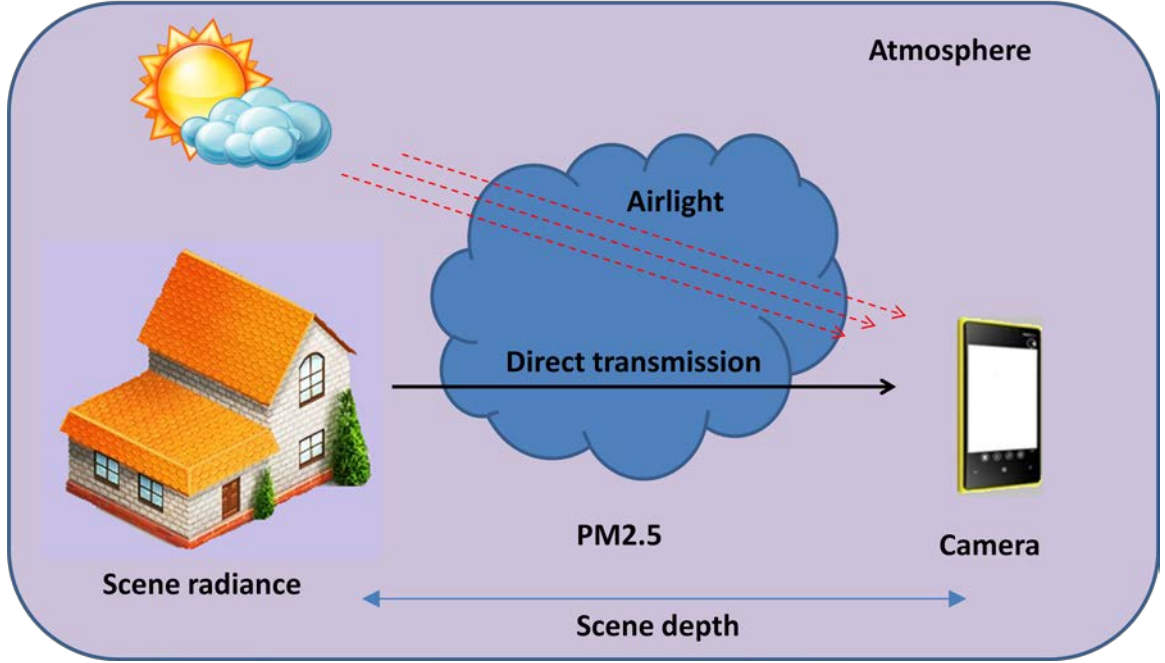


图 3-3 大气成像模型

$$I(x) = J(x)t(x) + A(1 - t(x)) \quad (3-1)$$

其中， x 为二维的图像平面坐标， $J(x)$ 为场景辐射强度， A 为天空亮度(airlight)。 $I(x)$ 为观测到的场景辐射强度，其可以通过对图像亮度经过相机辐射标定的逆变换过程得到。观察式(3-1)主要可发现其由两部分组成， $J(x)t(x)$ 的意义为物体本身辐射经大气衰减后进入相机的辐射强度，其可由图 3-3 中的黑色实线表示。 $A(1 - t(x))$ 的意义为太阳光经大气反射或折射后进入相机的辐射强度，其可由图 3-3 中的红色虚线表示。其中 $t(x)$ 是传播系数 (transmission)，是一个十分关键的大气物理模型参数，它的含义为光透过大气传播的比例，取值为 0 到 1 之间。

$$t(x) = e^{-\beta d(x)} \quad (3-2)$$

其中， β 为消光系数 (light extinction)， $d(x)$ 为场景深度，即拍摄的目标物与相机的直线距离。因为 PM2.5 颗粒物是影响消光系数的主要因子^{[12][13]}，所以我们可以基于消光系数，并利用 H. Ozkaynak 等^[33]提出的相关理论模型，实现 PM2.5 浓度预测。

$$\beta \approx pM_f \quad (3-3)$$

$$t(x) = e^{-pd(x)M_f} \quad (3-4)$$

在该模型中, M_f 为 PM2.5 浓度。 p 为常数项, 且在城市环境中, 一般设定为 3.75^[33]。综上, 我们之后将利用式(3-1)与式(3-4)进行 PM2.5 浓度估计。

3.2.2. 基于视觉的图像特征提取

基于视觉的特征主要分为两种: 空间域特征及频率域特征。空间域特征主要为图像灰度, 对比度, 梯度, 色彩等特征。频率域特征主要为图像经过傅里叶变换或离散余弦变换后的能量值。在本课题中, 我们选用的视觉图像特征为图像梯度特征及图像色彩特征。

3.2.2.1. 图像梯度特征

通过对雾霾图像的观察, 我们可以发现雾霾会降低图像中对比度的下降, 损失了大量的图像锐度信息。因此我们可以利用图像的梯度幅值来定义雾霾图像特征^[18]。对于相邻的像素, 我们可以认为它们拥有相同的场景深度, 天空亮度, 及 PM2.5 浓度。因此, 我们可以对式(3-1)两边同时求梯度。梯度特征可定义为

$$\begin{aligned} F_{sc} &= |\nabla_x I(x)| = |\nabla_x (J(x)t(x) + A(1-t(x)))| \\ &= t(x) |\nabla_x (J(x))| \\ &= e^{-pd(x)M_f} |\nabla_x (J(x))| \end{aligned} \quad (3-5)$$

通过对式(3-5)两边同时取对数, 我们可以得到

$$M_f = -\frac{1}{pd(x)} \ln(F_{sc}) + \frac{1}{pd(x)} \ln(|\nabla_x J(x)|) \quad (3-6)$$

通过对式(3-6)观察, 我们可以发现 PM2.5 浓度与 $\ln(F_{sc})$ 是呈线性关系的。通过使用常用的空间滤波器如 Prewitt 滤波器, Sobel 滤波器, Roberts 滤波器, 我们便可以快速的实现图像梯度求值。在本课题中, 我们选用 Sobel 滤波器对图像进行特征提取。

$$F_{sc} = f_{sobel}(I(x)) \quad (3-7)$$

在实验中, 我们可以发现地面上的物体, 如楼房, 公路, 树木等的梯度特征随着 PM2.5 浓度越大, 而梯度值越低。

3.2.2.2. 图像色彩特征

图像的梯度特征主要利用了雾霾对地面物体锐度降低的特点。事实上, 天空的色彩变化也可以被用来估计 PM2.5 浓度。天空的 HSI 色彩差异在一定条件下是与能见度, 大气消光系数等相关^[16], 并可用式(3-8)表达。

$$\beta = ae^{b\Delta D} \quad (3-8)$$

$$F_{hsi} = \Delta D = \sqrt{(\Delta H)^2 + (\Delta S)^2 + (\Delta I)^2} \quad (3-9)$$

其中, a 与 b 为该模型的两个未知参数。 ΔD 为 HSI 色彩差异, 我将其作为天空色彩特征 F_{hsi} 。如果我们使用一张无雾霾的图像作为参考图像, 则 HSI 色彩差异可由式(3-9)表示, ΔH , ΔS 及 ΔI 分别为目标图像与参考图像间天空色调, 饱和度及亮度的差值。最终, 利用式(3-3)和式(3-8), 我们可以得到

$$M_f = \frac{a}{p} e^{bF_{hsi}} \quad (3-10)$$

3.2.3. 基于物理的图像特征提取

与基于图像视觉特征的方法不同, 基于物理的图像特征的方法通过分析大气成像模型与 PM2.5 浓度关系, 来寻找可表征 PM2.5 浓度变化的物理模型参数。通过式(3-4)我们发现传播系数 $t(x)$ 与 PM2.5 浓度 M_f 呈现指数关系。并且 p 为已知的系数, $d(x)$ 可以认为在同一小块的图像中拥有相同的值。因此, 我们可以利用 $t(x)$ 估计 PM2.5 浓度。在计算机视觉的图像去雾领域中, 求解 $t(x)$ 也是一步重要的工作, 其影响着对雾霾图像复原效果。He 等人^[30]提出的利用暗原色先验估计算法方法可以有效地恢复图像的 $t(x)$ 图。本课题使用了该理论估计得到的 $t(x)$ 作为图像特征, 进行 PM2.5 浓度估计。其基于, 在同一图像区块中至少存在一个像素的某一色彩通道的强度接近于零, 这一发现实现对 $t(x)$ 的求解。首先, 需要定义图像的暗原色图像

$$J_{dark}(x) = \min_{y \in \Omega(x)} \left\{ \min_{c \in \{r, g, b\}} J^c(y) \right\} \quad (3-11)$$

其中, J_{dark} 坐标为 x 的像素的暗原色值。 $J^c(y)$ 为原始图像在 y 坐标下的 c 通道像素值。 $\Omega(x)$ 为一个以 x 为中心的图像区块。从公式(3-11)中, 我们可以发现针对任意给定像素的暗原色值为在相邻区块中的三个通道中的最低强度值。而暗原色先验便可以表达为

$$J_{dark} \rightarrow 0 \quad (3-12)$$

进一步地, 我们将式(3-11)带入到式(3-1)中, 我们即可以得到 $t(x)$ 的估计值

$$\tilde{t}(x) = 1 - \min_{y \in \Omega(x)} \left\{ \min_c \frac{I^c(y)}{A^c} \right\} \quad (3-13)$$

其中 $\tilde{t}(x)$ 为传播系数估计值, A^c 估计得到的天空亮度, 其等于暗原色图像中最亮的像

素所对应的原始图像中的亮度值。通过对式(3-4)的观察，我们可以发现传播系数 $t(x)$ 与 PM2.5 浓度呈现指数关系。因此，我们可以选用 $\tilde{t}(x)$ 作为图像特征，用 F_{dc} 来表示，从而得到

$$M_f = -\frac{1}{pd(x)} \ln(F_{dc}) \quad (3-14)$$

3.3. PM2.5 估计模型

通过使用上一节所介绍的方法计算得到的三种图像特征，我们将进一步使用其估计 PM2.5 浓度。本课题提出了两种可行的估计方法，基于学习的 PM2.5 估计方法（LB 方法）与基于对象的 PM2.5 估计方法（OB 方法）。我们对两者进行了大量的实验以验证其可行性，并在第五章中详细的说明了实验的结果与分析。LB 方法使用用户收集的在同一地点不同天拍摄的多张图像及最近 PM2.5 监控站点测量的数值，通过训练建立估计模型。而 OB 需要用户使用单张图像，并指出某一可以获取到距离的建筑物，无需训练过程。两者的不同体现在：首先，LB 方法需要一段较长时间的图像收集及模型训练过程，而 OB 方法可以实现实时估计。其次，LB 方法需要参考 PM2.5 浓度值作为模型训练的参考值，而 OB 方法无需外部传感器参考值，但是需要得到目标物的距离信息。最后，LB 方法无需任何用户交互过程，仅利用用户拍摄的多张照片。而 OB 方法需要用户手动圈画图像中的建筑物，并提供其名称以用来获取距离信息。

3.3.1. 基于学习的 PM2.5 估计模型

基于学习的模型首先需要主动地激励参与者在同一个地点收集 m 天数的 PM2.5 图像，并使用附近空气检测站点的 PM2.5 参考值，训练生成 PM2.5 浓度估计模型。通过在客户端使用图像特征提取算法与监督学习算法，针对某一场景，我们可以生成相应的 PM2.5 估计模型。这样，对于给定的同一场景的新图像，我们可以利用生成的估计模型进行 PM2.5 估计，其过程如式(3-15)给出

$$M_f = Model_{LB}(I, I_{ts}, P_{ts}) \quad (3-15)$$

其中， I 为测试图像， $I_{ts} = \{I_{ts}^1, I_{ts}^2, I_{ts}^3, \dots, I_{ts}^m\}$ 为 m 天的训练图像， $P_{ts} = \{P_{ts}^1, P_{ts}^2, P_{ts}^3, \dots, P_{ts}^m\}$ 为 m 张训练图像所对应的参考 PM2.5 浓度值。 M_f 为利用模型估计得到的 PM2.5 浓度值。LB 方法的框架如图 3-4 所示。其主要包含 3 个步骤，分别为图像预处理过程，图像特征提取过程，PM2.5 估计过程。

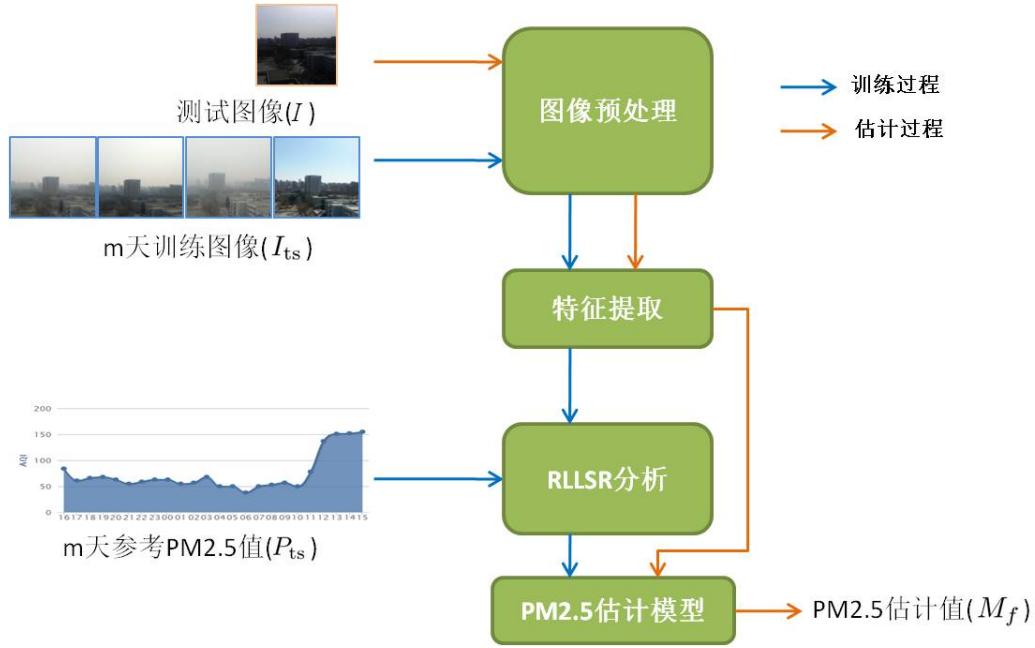


图 3-4 基于学习的 PM2.5 估计算法框架

3.3.1.1. 图像预处理过程

图像预处理过程主要用来在特征提取前完成两个任务。首先，用户收集到的图像并不都是完全有效的图像。例如，一些在特殊天气情况，如雨雪，沙尘的环境中，大气中的其它颗粒物会对大气成像模型产生随机的干扰^[34]，从而影响 PM2.5 的估计过程。因此，我们需要过滤掉不符合有效天气条件的图像。为了达到此目的，本课题将通过抓取拍摄地所在地区的气象数据作为依据，判断其是否符合有效数据的气象条件。其次，如 3.1 节所述，我们还需要利用先前得到的拍摄所用手机型号对应的相机辐射响应模型对图像进行辐射标定，将图像恢复为原始场景辐射。

3.3.1.2. 图像特征提取过程

本课题分别使用图像梯度特征 F_{sp} ，图像天空色彩特征 F_{hsi} ，图像暗原色特征 F_{dc} 去估计 PM2.5 浓度。并且在第六章中进行了大量实验与分析，从而发现 F_{dc} 特征可以有效地实现较为精确的 PM2.5 估计。因此，在最终实现的系统中，我们在特征提取步骤中只使用 F_{dc} 特征。

3.3.1.3. PM2.5 估计过程

在 PM2.5 建模过程中，我们将 m 天的图像及相应 PM2.5 参考值作为输入，使用式 (3-14)，利用稳健线性回归算法(RLLSR)分析，进而得到式中的参数。例如，根据式(3-14)，

针对任意 F_{dc} 特征值，我们可以计算出其相应 $PM_{2.5}$ 浓度值 M_f 。

但是，通过大量实验，我们发现并不是所有的图像区域都能够有效的分析出 $PM_{2.5}$ 信息。例如，针对 F_{sc} 和 F_{dc} 特征，距离过远或者过近的物体图像区域都不与 $PM_{2.5}$ 浓度呈现较大的相关性（见图 5-4）。针对 F_{hsi} 特征，地面的物体都会给 $PM_{2.5}$ 估计造成较大的误差。因此本课题使用了一种滑动窗口策略来提取 $PM_{2.5}$ 相关特征。该滑动窗将沿着图像的纵横方向逐步移动，并计算窗口内特征平均值，作为该窗口的特征值。进而分析出与 $PM_{2.5}$ 浓度相关性最高的窗口作为最优窗口进行 $PM_{2.5}$ 估计。

3.3.2. 基于对象的 $PM_{2.5}$ 估计模型

本课题同时也提出了另一种在周围没有 $PM_{2.5}$ 参考值可提供的场景下，无需较耗时的训练过程的 $PM_{2.5}$ 实时估计方法，基于对象的 $PM_{2.5}$ 估计模型。

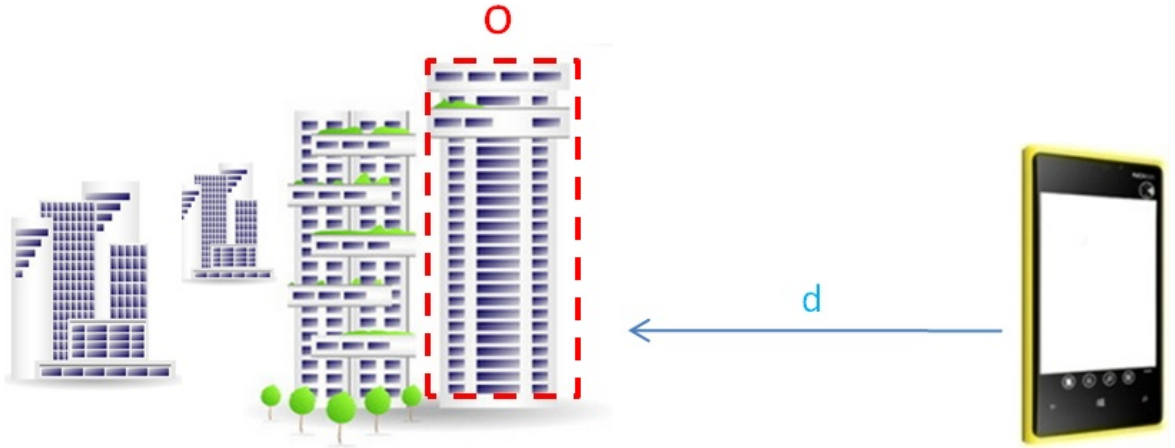


图 3-5 基于对象的 $PM_{2.5}$ 估计方法示意图

如图 3-5 所示，用户首先需要输入一张图像，并且手动圈画出某一目标物所在的矩形区域，如图 3-5 中的 o 区域。选取的目标物应为已知名称的可以从背景中区分开的建筑物，距离 d 可通过该建筑物名称使用相关互联网服务，如 Google Maps 等获取该建筑物的 GPS 信息，由此利用拍摄时智能手机获取的 GPS 信息计算得到距离。其过程可由式(3-15)给出。

$$M_f = Model_{OB}(I, o, d, f_{lp}) \quad (3-16)$$

其中， I 为测试图像， o 为目标建筑物， d 为建筑物与用户的距离， f_{lp} 为大气消光系数与 $PM_{2.5}$ 浓度的关系，其可从式(3-3)中获取。 M_f 为 $PM_{2.5}$ 浓度估计值。图 3-6 描述了基于对象的 $PM_{2.5}$ 估计算法框架，其可以分为图像预处理，特征提取及 $PM_{2.5}$ 模型估计 3 个过程。

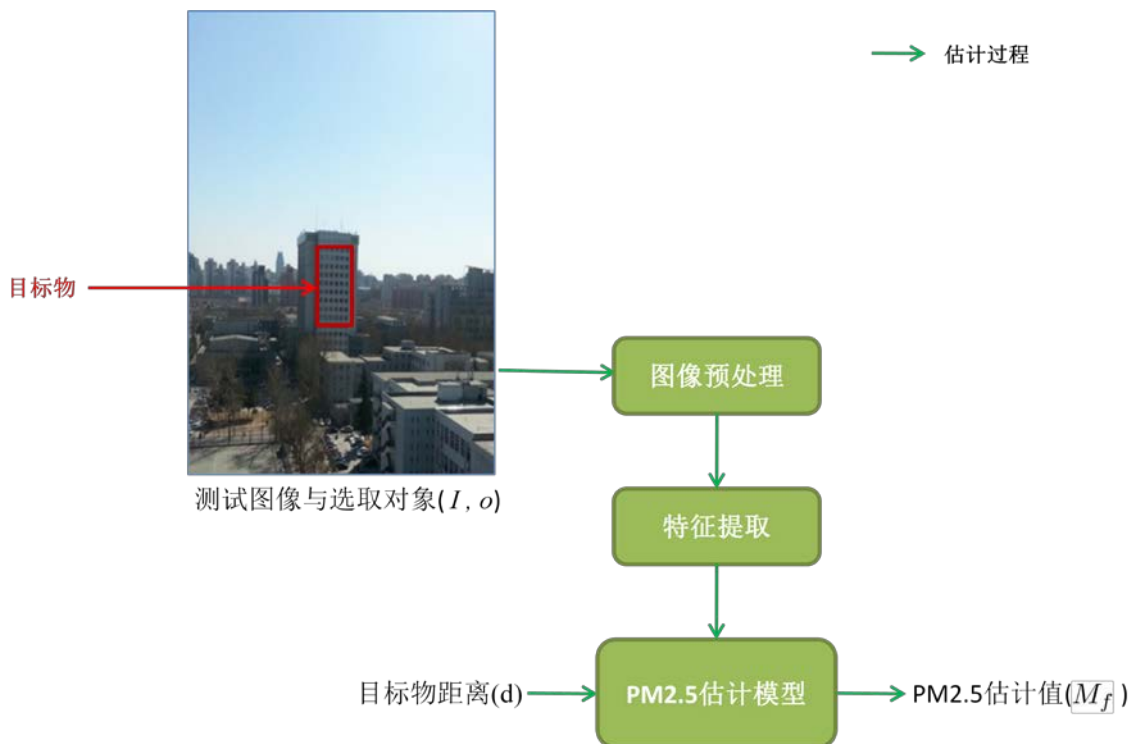


图 3-6 基于对象的 PM2.5 估计算法框架

3.3.2.1. 图像预处理过程

基于对象方法对图像预处理过程基本与基于学习方法的图像预处理过程相似。都需要通过抓取拍摄地所在地区的气象数据作为依据，判断其是否符合有数据的气象条件。其次，我们还需要利用先前得到的拍摄所用手机型号对应的相机辐射响应模型对图像进行辐射标定，将图像恢复为原始场景辐射。但是基于对象的方法需要用户在图像中使用矩形圈画出目标建筑物，并且需要连接互联网获取相关距离信息。

3.3.2.2. 图像特征提取过程

该步骤与基于学习的方法类似，都使用图像暗原色特征 F_{dc} 去估计 PM2.5 浓度。但在基于对象的方法中，我们使用用户所圈画矩形所在图像区块的特征平均值作为最终的图像特征。

3.3.2.3. PM2.5 估计过程

在基于对象的方法中，我们利用式(3-14)计算 PM2.5 浓度估计值，因为距离 $d(x)$ 已知。此外， p 将根据城市环境中的大气理论模型^[33]设定为 3.75。由此，利用第二步得到图像特征 F_{dc} ，我们便可以直接计算出 PM2.5 浓度估计值。

3.4. 本章小结

本章主要介绍了本课题所提出的基于图像分析的 PM2.5 估计方法。首先，分析了异构手机所拍摄的图像的归一化过程。随后，依次给出了几种可以被利用分析 PM2.5 的图像特征，并详细分析了其计算方法。最后，根据不同的应用环境，本课题提出了两种 PM2.5 估计方法，分别为基于学习的 PM2.5 估计方法和基于对象的 PM2.5 估计方法，并详细介绍的其系统框架及工作流程。

第四章 PM2.5 分析系统设计与实现

4.1. 概要设计

4.1.1. 总体设计

PM2.5 分析系统主要利用分布在城市中不同位置的参与者利用其智能手机收集到足够的相片，并使用之建立 PM2.5 估计模型，从而实现大范围，细粒度的 PM2.5 浓度估计。图 4-1 展示了该系统的网络架构图，本系统由一定数量的智能手机及一个云端服务器组成。气象站所抓取的真实 PM2.5 浓度数据被系统作为模型训练过程中的参考值从而提高估计准确度。参与者将被激励每隔一段时间使用智能手机拍摄一张相片，用来估计区域 PM2.5 浓度值。随后，智能手机可以通过 WiFi 或移动 3G 将 PM2.5 估计值上传至服务器端，经过一定数据融合算法处理后，即可以创建城市的 PM2.5 浓度分布图。

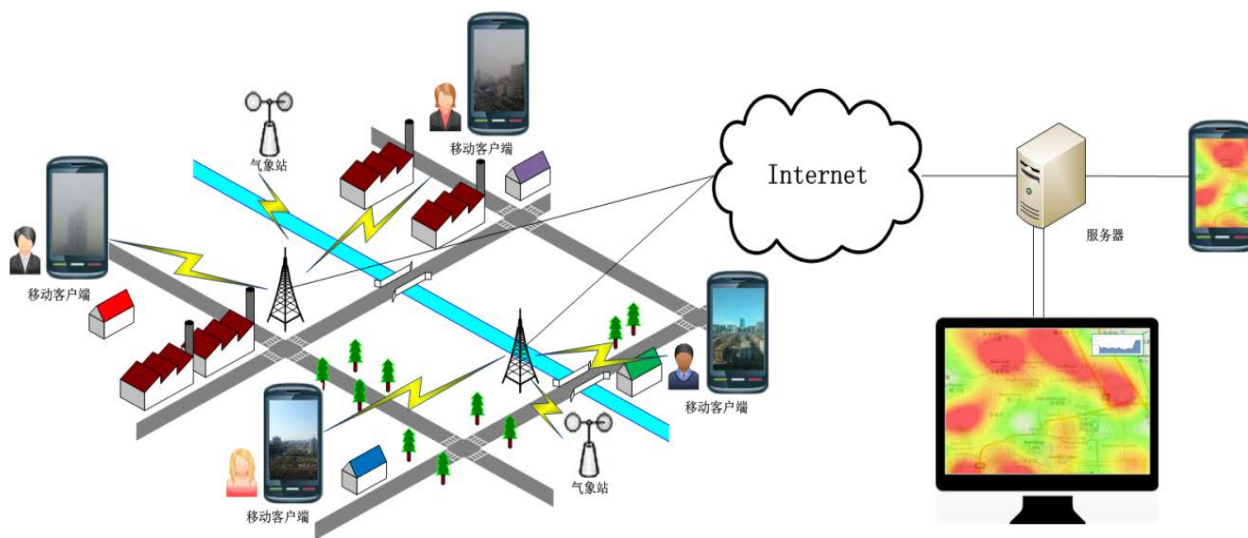


图 4-1 系统架构图

本课题所选用的系统设计为分布式设计，即所有的 PM2.5 数据收集，建模，计算等过程都在移动客户端进行，而无需上传至服务器端进行统一处理。相比之下，集中式设计，即用户首先需要上传所有相片，并在后台进行计算与分析。虽然可以更有效地利用后台服务器强大的存储与计算能力，但是其无法避免了因为数据无线传输所造成的带宽及能量消耗。而分布式设计却能够很好的减少 WiFi 或 3G 的数据传输消耗及数据上传导致的电量消耗。因此，本课题采用了如上所述的分布式设计。

PM2.5 分析系统需要实现的主要功能为图像收集，雾霾特征提取，估计模型建立及 PM2.5 估计等。图 4-2 展示了详细的系统结构与数据流图。其中，系统的基本处理过程为数据收集过程，特征提取过程，PM2.5 建模过程，PM2.5 估计与上传过程。

数据收集过程主要用来通过智能手机收集原始的图像信息，利用罗盘传感器收集拍摄方向信息，利用 GPS 传感器收集位置信息，利用加速度传感器收集手机拍摄的姿态信息等。

特征提取过程主要使用 F_{dc} 特征进行雾霾特征提取，我们使用轻量级的图像处理算法在智能手机上快速地实现图像特征提取过程，从而避免在服务器端进行图像处理所需的数据传输所带来的开销。

PM2.5 分析建模过程使用用户手机得到的若干图像数据及相应参考 PM2.5 浓度信息进行建模。根据不同的应用场景，本课题提出了两个 PM2.5 分析方法，基于学习的方法（LB）与基于对象的方法（OB）。LB 方法同时使用图像信息与参考 PM2.5 信息，需要达到一定图像训练集大小才可以建立较为准确的 PM2.5 分析模型。而 OB 方法可以在附近没有参考 PM2.5 值可使用的情况下进行较为粗略的 PM2.5 实时估计。但是其需要用户手动圈画一个目标物出来，并从互联网相应服务中得到该目标物与用户的距离信息。因此，我们主要基于 LB 方法实现了 PM2.5 的估计系统。

PM2.5 估计与上传过程主要使用新收集到图像与 PM2.5 分析模型，在智能手机中进行 PM2.5 浓度估计，并将相应的估计值上传至服务器中。服务器随后利用收集到所有 PM2.5 估计值进行数据融合处理，并最终得到细粒度的城市 PM2.5 浓度分布图等相关数据可视化信息。

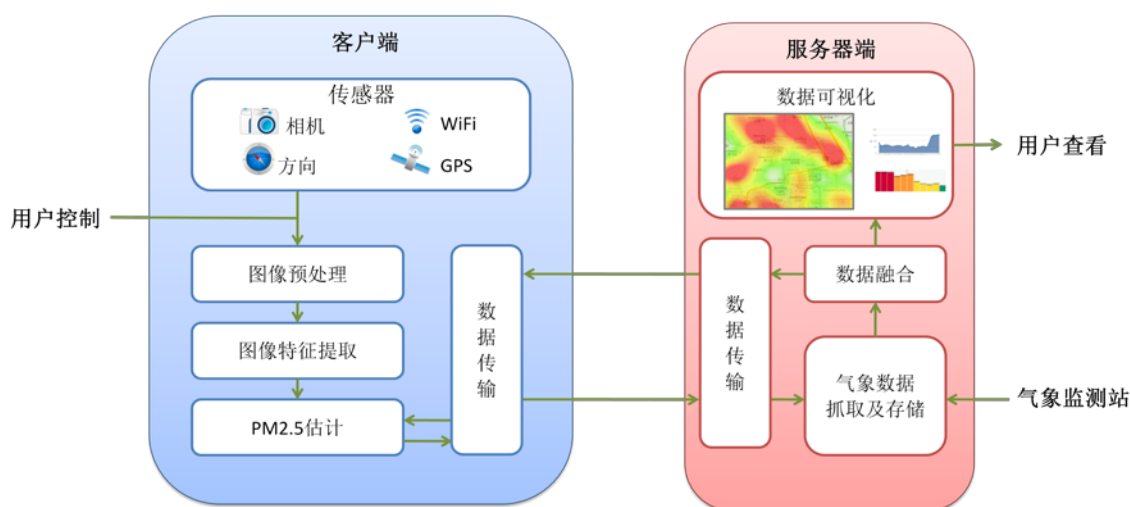


图 4-2 系统结构及数据流图

4.1.2. 模块设计

PM2.5 分析系统主要包含两个流程，PM2.5 建模过程和 PM2.5 估计过程。两个过程又总共由 5 个逻辑模块组成，分别为相机模型标定模块，数据预处理模块，图像特征提取模块，预测模型训练与 PM2.5 估计过程。

PM2.5 建模过程如图 4-3 所示。首先，该过程利用用户拍摄的一张图像分析相机的

辐射相应模型，得到相机辐射变换的相关参数。然后将对用户所拍摄的多张雾霾照片进行有效性分析与归一化处理。随后，针对每一张预处理图像进行特征提取，依据图像特征及 PM2.5 参考值，利用滑动窗口技术，对预测模型进行参数估计，得到准确的预测模型。



图 4-3 PM2.5 建模过程

PM2.5 估计过程如图 4-4 所示。首先，对用户拍摄的一张雾霾图像，进行有效性检测与图像归一化处理。然后，针对该预处理图像进行特征提取。随后依据图像特征及之前得到的 PM2.5 估计模型，计算 PM2.5 估计值。



图 4-4 PM2.5 估计过程

4.2. 客户端设计

PM2.5 分析系统的大部分功能都主要在客户端实现，本课题采用分层架构将各个模块主要分为三层。图 4-5 展示了 PM2.5 分析系统客户端的三层设计架构，从上至下依次为应用层，数据层及引擎层。其中，应用层主要负责封装底层的算法实现并向不同的客户端程序提供 PM2.5 分析接口，如数据预处理接口，图像特征提取接口等。在本课题中，PM2.5 估计系统的客户端部分在 Android 系统中实现。因此，应用层主要使用 Java 语言，并利用 Android SDK API 完成，其与客户端的操作系统相关。数据层与引擎层使用 C/C++ 语言与 OpenCV 函数库的支持完成，其可以做到与客户端的操作系统无关，因此具有较强的可移植性，可以在大部分主流的移动操作系统上运行。其中，数据层主要负责对 PM2.5 图像分析，建模，估计过程中用到数据进行抽象，封装为不同的数据结构，例如训练数据，预测数据，估计模型，相机辐射逆变换模型等。引擎层主要负责实现最基础的图像特征提取过程与稳健线性回归过程。

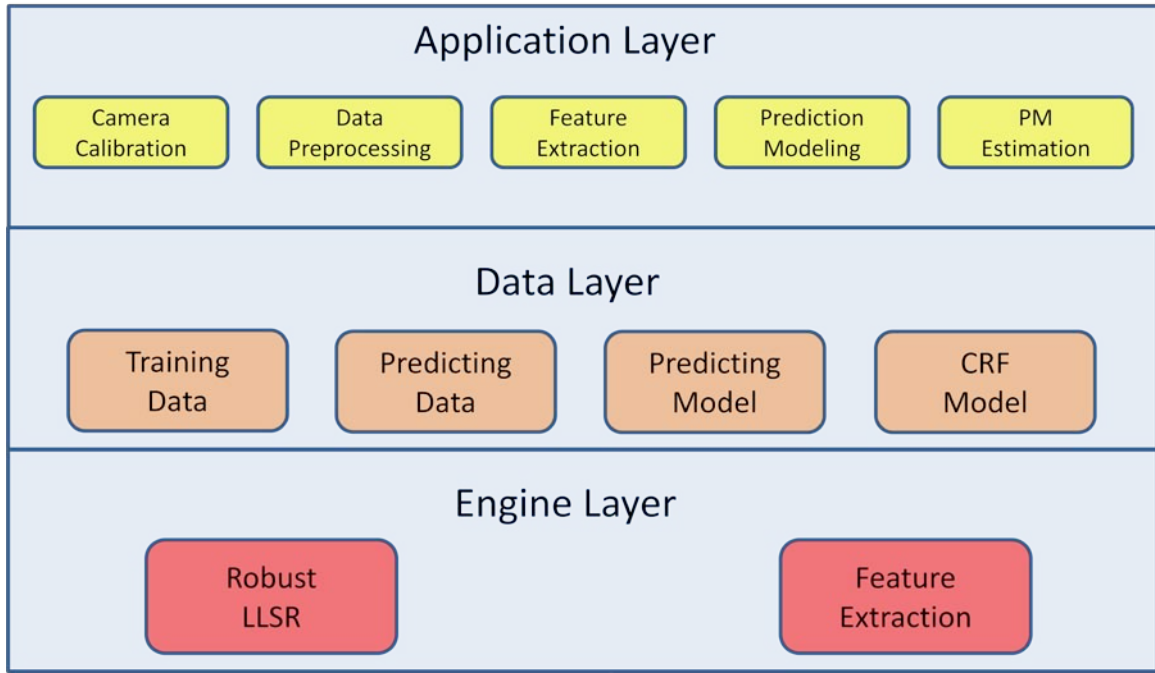


图 4-5 客户端模块结构

4.2.1. 应用层设计与实现

应用层主要负责将底层实现的 PM_{2.5} 图像分析，建模等过程进行封装，向上层的移动客户端应用程序提供相应的接口。其中主要包含图像辐射标定模块，数据预处理模块，图像特征提取模块，PM_{2.5} 建模模块，PM_{2.5} 估计模块。

4.2.1.1. 图像辐射标定模块

图像辐射标定模块主要负责相机模型标定过程，图像辐射逆变换过程。其详细的流程图如图 4-6 所示。首先，我们利用智能手机的 IMEI 来判别手机的型号。IMEI 码由 15 位数字组成，前 6 位为型号核准号码（TAC 码），接着 2 位为装配号（FAC 码），之后 6 位数为生产序列号，最后 1 位检验码，通常为 0。因此，我们利用 IMEI 码的前 8 位的 TAC 码与 FAC 码作为手机型号的唯一标识，简称为相机辐射标示码（CRI 码）。经过大量的实验测试，我们可以认为同一型号的手机所装备的摄像头具有同样的辐射响应函数。其次，我们利用 CRI 码在本地中获取相机辐射逆变换模型。若本地不存在，则从服务器端获取。若服务器端依然不存在该型号的模型，则将自动上传一张照片利用单幅图像标定算法分析出相机辐射逆变换模型，并下载至手机端。最终利用该模型实现辐射逆变换过程。

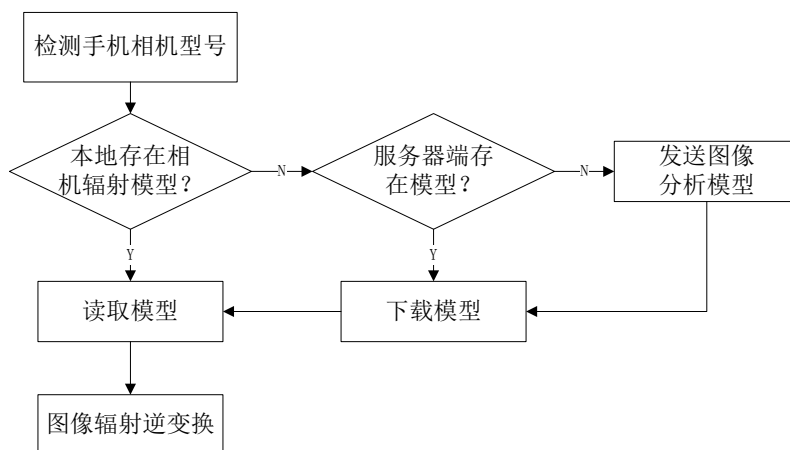


图 4-6 图像辐射标定流程图

4.2.1.2. 数据预处理模块

数据预处理过程主要负责利用从服务器端获取气象数据，对训练集图像和估计图像进行有效性测试。首先，输入图像必须符合一定条件才可以被用来进行 PM2.5 估计。条件主要分为两类：图像参数条件及气象参数条件。图像参数条件主要规定输入图像的统一图像格式，分辨率参数，图像曝光及白平衡参数等。气象条件主要根据图像拍摄时的时间戳信息在服务器中请求相应的天气信息，包括天气状态，湿度，风速等。本课题的 PM2.5 分析算法只利用非特殊天气下的照片进行图像分析。例如，在实验中我们发现在雨雪天所拍摄的照片将严重影响 PM2.5 分析模型的准确度，因此，我们将不对特殊天气条件下所拍摄的图像进行 PM2.5 分析。随后，我们将统计图像训练集中的有效照片数，在系统中，超过 15 张以上的训练集才可被用来进行建模。

4.2.1.3. 图像特征提取模块

图像特征模块主要负责封装底层引擎库所实现的特征提取过程。其主要目的是在用户拍摄完一张图像后，立刻在后台进行图像特征提取过程，并将其存储在临时目录中。因此，在建模过程中，我们可以直接利用特征图像进行 PM2.5 估计建模，节省了大量计算时间，同时也一定程度上提高了用户体验。

4.2.1.4. PM2.5 建模模块

PM2.5 建模模块主要负责利用图像训练集与参考 PM2.5 数据进行回归分析，并生成拍摄场景的 PM2.5 估计模型，其过程如图 4-7 所示。首先，系统为了能够使尽量缩短改模块的计算时间，将首先使用已经生成好的特征图像，如果不存在，则使用特征提取模块生成。随后，根据图像时间戳读取相应的 PM2.5 信息。然后，系统将使用默认的配置文件初始化模型信息，如建模中需要的滑动窗口大小，比例，移动步长等参数设置。之后利用底层引擎库实现的 RLLSR 回归算法计算出估计模型及相关参数，如最优窗口，

相关系数及平均绝对误差等信息，并以 yml 格式将模型文件存储在本地，使用以图像所拍摄场景生成的标记信息命名。

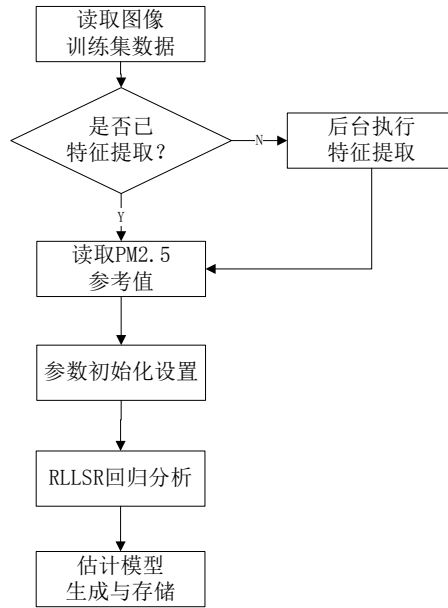


图 4-7 PM2.5 建模模块流程图

4.2.1.5. PM2.5 估计模块

PM2.5 估计模块负责使用已经生成的 PM2.5 估计模型对新拍摄的图像进行 PM2.5 浓度估计。首先，根据新拍摄的图像的 GPS，拍摄方向等生成的标记信息检索本地的 PM2.5 估计模型文件。如果存在该文件，就先读取经过特征提取的图像，及模型最优窗口信息及 PM2.5 估计方程，进行 PM2.5 浓度计算，最终得到估计值。

4.2.2. 数据层设计与实现

数据层主要封装了两大类信息，图像数据集及模型数据。图 4-8 与图 4-9 分别表示了图像数据集的数据结构，与模型数据的数据结构。

其中，模型数据结构主要为估计模型与相机辐射模型，我们使用 BaseModel 去抽象模型所必须的数据，如 id 及存储路径等。CamResModel 为相机辐射模型的数据结构，其主要包含一个 $3 \times 256 \times 256$ 的逆变换矩阵，针对 8 位深度的 RGB 色彩空间图像。PredictionModel 为估计模型的数据结构，其主要包含一个训练数据集 mTrainingData, mBlockDiv, mStepDiv, mBlockScale 为滑动窗口的参数，mStepDiv 为滑动窗口的长度相对于整张图像宽度的比例。mBlockDiv 为滑动窗口移动步长相对于图像宽度的比例，mBlockScale 为滑动窗口的长宽比例。例如，针对一张 1152×2048 的图像，如果 mBlockDiv 为 4，mStepDiv 为 8，mBlockScale 为 1，则滑动窗口为长度为图像宽度 1/4 的正方形，且移动步长为图像宽度的 1/8。mMaxRsBlockRow 与 mMaxRsBlockCol 为最

佳窗口的横纵坐标, mRs 与 mMae 分别为模型的相关系数与平均绝对误差, mCoefArray 为最终要利用的 PM2.5 估计方程中的相关参数。

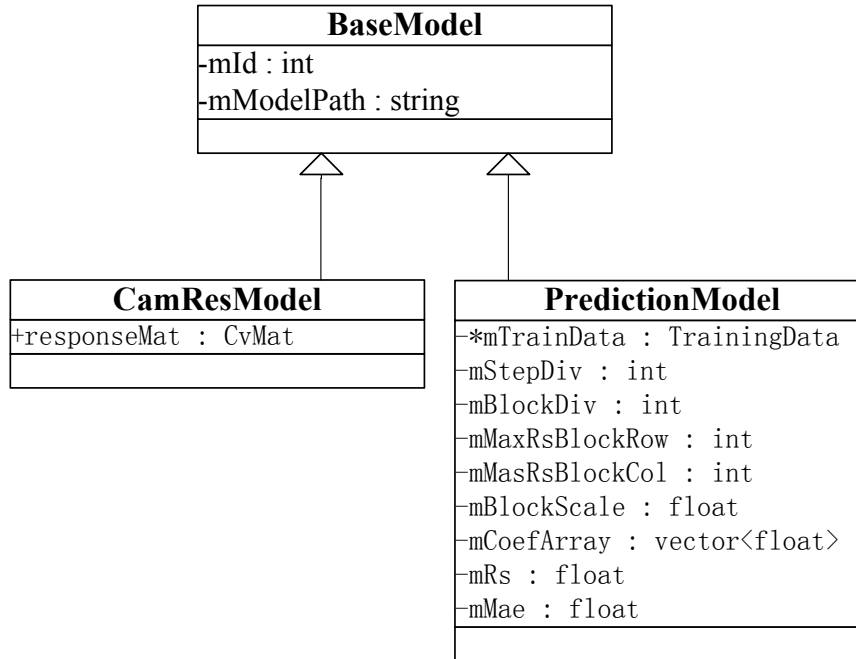


图 4-8 模型类数据结构设计

图像数据结构主要为训练集数据与估计数据。我们使用 **BaseData** 去抽象图像数据类的基本数据结构单元。mId 用来作为数据的唯一标识, mImgWidth 与 mImgHeight 标识图像的宽度与高度, mImgChannel 与 mImgDepth 为图像的通道数与位深度, mBlockRows 与 mBlockCols 为图像可以容纳的横向与纵向窗口数, mSlidingRowStep 与 mSlidingColStep 为滑动窗口的横向与纵向移动步长。mImgType 为图像数据的格式信息, crfModel 为该图像数据所应用的相机辐射标定模型。PredictingData 为估计数据, 其中 mPredFilePath 为该图像的存储路径, mPredPM 为最终估计得到的 PM2.5 浓度, mGTPM 为 PM2.5 的真实值。TrainingData 为训练数据集, mSrcFilePathArray 为所有原始图像的数据路径, mTmpFilePathArray 为经过所有特征图像的路径, mRefPM 为训练时参考的 PM 浓度值序列。

综上所述, 数据层主要实现了 PM2.5 估计算法中的所有的数据实体, 并利用底层引擎库中实现的算法逻辑, 完成 PM2.5 估计算法中的具体的特征提取, 模型建立, PM2.5 估计等模块, 并向上层应用层提供所有相应的接口。

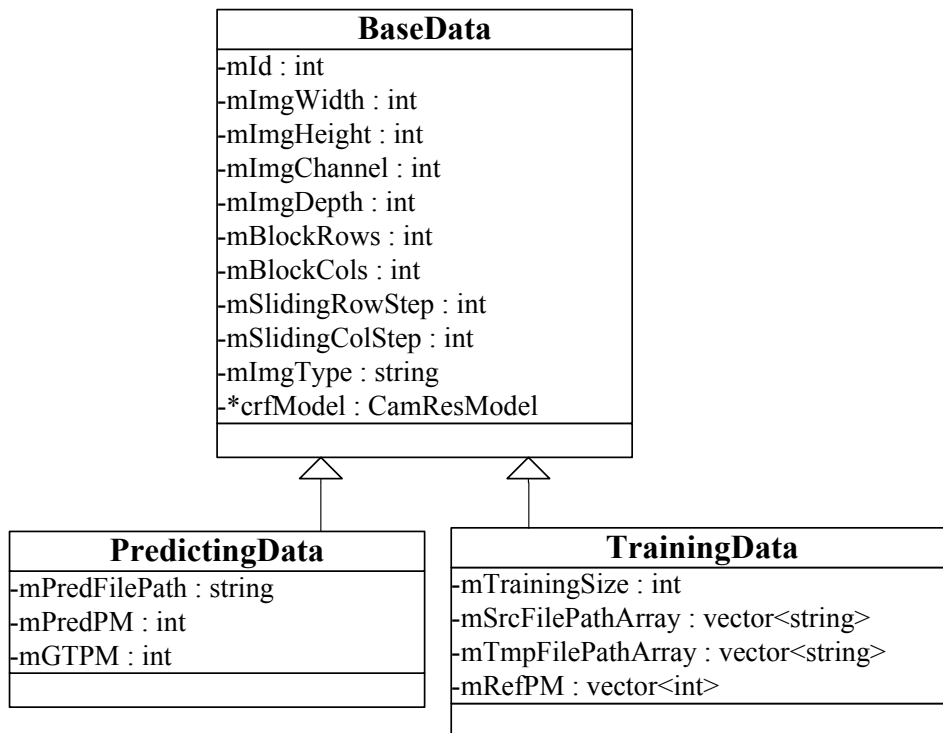


图 4-9 数据类数据结构设计

4.2.3. 引擎层设计与实现

如图 4-5 所示，客户端的引擎层主要为回归分析模块与特征提取模块。其中，特征提取模块主要负责底层的图像处理算法。回归分析模块主要负责稳健回归算法。

4.2.3.1. 特征提取模块

特征提取模块实现了图像 F_{dc} 特征的提取算法。该算法对输入的一幅 RGB 空间图像分别进行快速暗通道计算，天空亮度计算与传播系数计算等，最终输出一幅单通道 F_{dc} 特征图像，其算法的主要过程如图 4-10 所示。

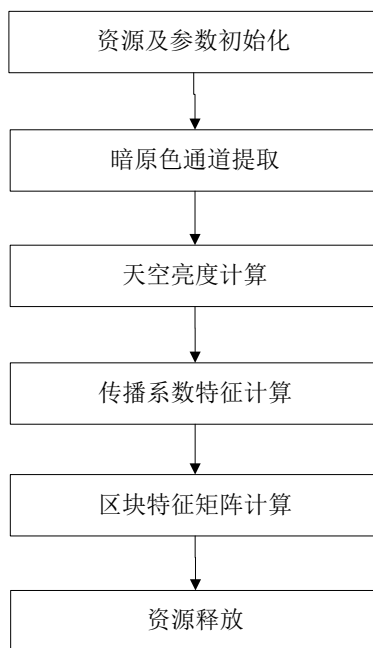


图 4-10 特征提取模块流程图

其主要步骤如下：

第一步，进行资源及参数初始化，由于图像处理程序经常会占用较大的资源空间，且智能手机中的内存资源相对较少，所以我们需要格外小心地控制向移动终端操作系统所申请的内存资源，如暗原色图像，传播系数图像等，在使用完毕后，将会立即得到释放，避免占用客户端过多的内存空间。此外，还有诸如图像区块参数的配置数据及为了计算天空亮度所使用的优先队列等数据结构需要提前被初始化。

第二步，暗原色通道提取过程将通过原始图像利用式(3-11)计算图像的暗通道。其主要过程为计算每个图像像素的邻域像素所有通道的最小值。假设一张图像大小为 $h \times w$ ，邻域区块大小为 $p \times p$ ，则计算总的时间复杂度将为 $O(hwp^2)$ 。这样复杂的计算将在智能手机端十分耗时。因此，我们利用改进的 HGW 算法^[35]将时间复杂度降低为 $O(hw)$ ，即每个像素点只需要遍历常数不超过 6 次即可，而非 p^2 次。该算法使用四倍的图像内存空间来换取时间性能的优化。其分别利用 4 块内存空间计算正逆最小行矩阵与正逆最小列矩阵。其计算过程首先如图 4-11 所示，先求正逆行最小值矩阵。图像的每一行首先将如图中的纵向虚线所示被邻域长度 p 分为若干部分，随后，分别如图中菱形及圆形所示，依次计算逆向及正向的行最小值，并分别保存在 rowMin 矩阵及 rowMinReverse 矩阵。这样，我们即可以快速的求出每一行任意一点长度为 p 的数列最小值。例如，假设 p 的大小为 5，如果我们要计算以该行第 4 列为中心，长度为 5 的行最小值，如图 4-11 所示，其结果将为 rowMin[2] 与 rowMinReverse[6] 中的最小值。其中，我们可以利用正向最小值计算的结果来优化逆向最小值计算过程，如果逆向遍历时最小值结果为正向遍

历的最小值，则之后只需要将剩下的填充为该最小值，而无需多余的比较过程。接下来，我们利用得到的 rowMin 矩阵及 rowMinReverse 矩阵，我们利用同样的方法即得到 colMin 矩阵及 colMinReverse 矩阵，则最终的暗原色通道图像便可由这两个矩阵得到。同时，在此过程中，我们还将使用一个大小为图像像素数 0.05% 大小的优先队列存储最亮的暗原色像素的坐标，为之后更快速地计算天空亮度。

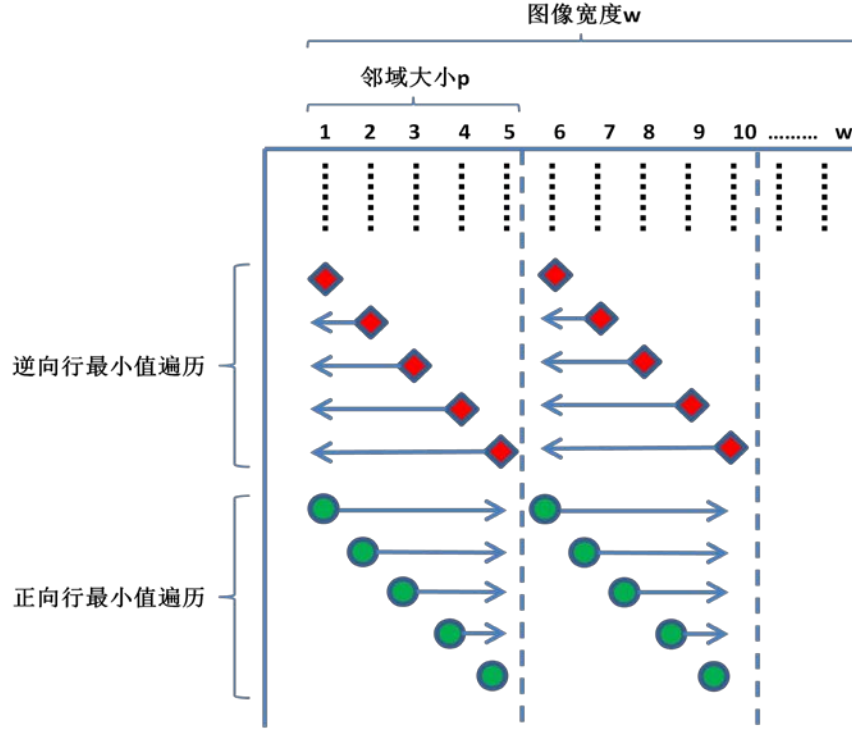


图 4-11 暗原色通道提取算法示意图

第三步，根据在计算暗原色通道步骤中更新得到的优先队列中的原图像亮度最大的像素对应的亮度作为天空亮度。

第四步，通过前几步骤得到的暗原色通道及天空亮度，使用式(3-13)计算得到传播系数。其过程与步骤二相似，都需要计算 4 个正向与逆向的行列矩阵，并利用第三步得到的天空亮度计算传播系数矩阵。

第五步，根据滑动窗口的设置，分别横向及纵向的移动滑动窗口，并将窗口内所有像素的传播系数平均数作为该窗口的特征值 F_{dc} 。从而得到最终的特征矩阵，矩阵的行数及列数分别对应着窗口横向及纵向所能移动的最大步数。例如，图像大小为 $h \times w$ 窗口大小为 $l \times l$ ，滑动窗口移动步长为 $\frac{1}{4}l$ ，则矩阵的大小为 $\left\lfloor \frac{4(h-l)}{l} + 1 \right\rfloor \times \left\lfloor \frac{4(w-l)}{l} + 1 \right\rfloor$ 。

最终，我们便可以利用这个矩阵进行稳健回归分析。

第六步，由于最后我们只利用特征矩阵，且特征图像将会存储在临时的硬盘空间中已供在未来不同参数设置的特征矩阵生成，所以我们需要将所有中间生成特征图像所占

用的内存资源释放，例如，暗原色图像，传播系数图像，天空亮度优先队列等资源，从而结束图像特征提取过程。

4.2.3.2. 回归分析模块

回归分析模块使用稳健线性最小二乘回归分析（RLLSR）方法，利用式(3-14)，对输入的特征值对数与 PM2.5 参考浓度值进行回归分析。最终以 $M_f = a \ln(F_{dc}) + b$ 的形式输出参数 a, b。其具体的实现算法有由多次的迭代过程组成，每一次迭代主要由四步组成，如图 4-12 所示。

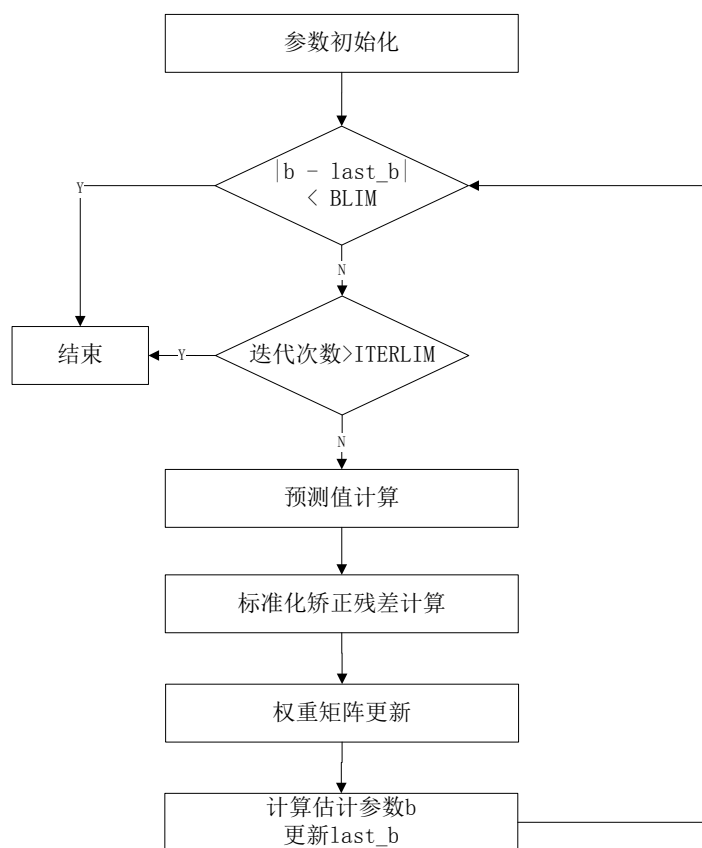


图 4-12 稳健回归算法流程图

首先，算法将初始化基本参数信息，如估计参数残差阈值 BLIM，最大迭代次数阈值 ITERLIM 等。其次，还需要初始化权重矩阵，Hat 矩阵，估计参数矩阵等。随后将进入迭代过程，该过程如图 4-12 所示，主要由预测值计算，标准化矫正残差计算，权重矩阵更新及估计参数更新等四步组成。其中，第一步预测值计算过程利用上一步骤中得到的估计参数 b 进行预测值计算。第二步中，为了减少异常值对估计参数的影响，通过标准误差估计和调节参数来调整残差，从而得到标准化矫正残差。并利用其在第三步中，重新计算权重矩阵，从而通过权重值来降低异常值对估计模型影响，达到降低数据集噪

音的目的。第四步，通过更新的权重矩阵重新计算估计系数 b ，并将上一次迭代过程得到的估计系数作为 $last_b$ 。最终，经过若干次上述迭代过程后，如果 b 的变化量小于 $BLIM$ ，则将 b 输出作为模型最终的估计参数。此外，算法还设置了最大迭代次数阈值 $ITERLIM$ 以增加对算法的可控性，使其在有限的时间内完成。

4.3. 服务器端设计

PM2.5 估计系统的服务器端的主要任务为为客户端的 PM2.5 估计提供气象数据以进行数据预处理和 PM2.5 建模。同时，其还为图像预处理过程中的相机辐射逆变换提供相机辐射模型。图 4-13 展示了服务器端的总体架构。其总体主要包含四个模块，数据抓取模块，相机模型分析模块，数据访问模块及网络访问模块。其中数据抓取模块主要负责抓取气象数据，如天气状态，湿度，风速等信息，及 PM2.5 浓度信息等。相机模型分析模块主要负责对客户端上传的照片进行辐射模型标定，并将分析得到的相机模型存储在数据库中，并为相同手机型号的客户端提供相机辐射模型。数据访问模块主要负责与后台不同种类的数据库进行数据读写。该模块具有较好的可扩展性，可以轻易的适配其它类型的数据库。网络访问模块主要负责对外部网络的访问请求控制。

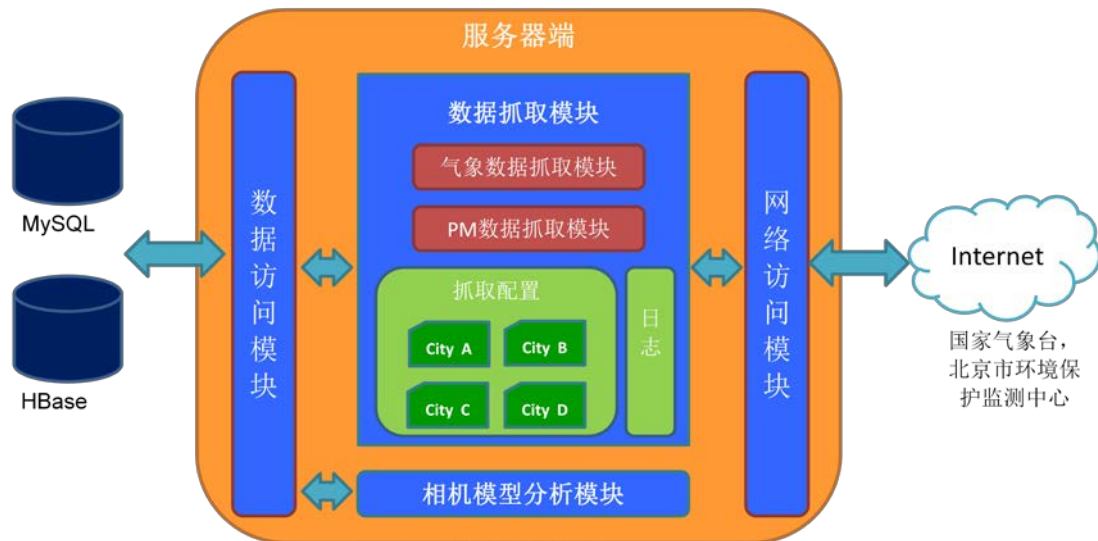


图 4-13 服务器端架构框图

4.3.1. 相机归一化模块

相机归一化模块是在图像预处理中所需要的重要过程，本课题利用 Lin.S 等人^[32]所提出的利用单幅图像进行辐射标定的方法进行相机模型分析，并在服务器端实现。经过测试，我们发现分析过程计算量较大，较为耗时。因此我们选择将其以网络服务的形式部署在服务器端，利用服务器强大的计算能力进行处理，而没有选择在客户端实现该模

块。

表 4-1 展示了相机模型数据表设计。其主要用于记录使用手机 IMEI 码的前 8 位作为手机型号标识所对应的相机模型。

表 4-1 相机模型数据表

字段	含义	类型	主键	非空
id	用来作为唯一标识的序列，数据库保证其递增且不重复的	int	√	√
phoneModel	手机型号 (8 位 TAC+FAC 码)	int		√
crfModel	相机模型存储路径	varchar(100)		√
datetime	模型存储时间	datetime		√

如果客户端所请求的手机模型所对应的相机模型不存在，则说明目前系统中尚无人使用过该型号手机。因此，系统将要求客户端上传一张图像使用单幅图像辐射标定算法计算得到相机标定模型，并在数据库中插入该手机型号的模型信息。因此该模型对外所提供的操作主要为根据手机型号，返回相机模型文件和根据上传测试图像，返回模型分析结果信息。

4.3.2. 气象数据抓取模块

气象数据抓取模块主要负责在全国个主要城市进行天气数据及 PM2.5 浓度数据的抓取与存储。抓取过程主要利用第三方 Webmagic^[36]开源框架在特定气象数据与 PM 数据公开网站的页面中进行抓取。其在数据库中分别对应的是 Monitor 表与 Weather 表，如表 4-2 与表 4-3 所示。其中，Monitor 表主要包含检测站站点 id，PM2.5，PM10 与 AQI 指数，记录时间。Weather 表主要包含城市 id，天气状态，温度，湿度，气压，风速，降雨量，记录时间等。

表 4-2 Monitor 数据表

字段	含义	类型	主键	非空
id	用来作为唯一标识的序列，数据库保证其递增且不重复的	int	√	√
siteId	监测站 id	int		√
fpm	PM2.5 浓度 (ug/m3)	smallint		√
cpm	PM10 浓度 (ug/m3)	smallint		√
aqi	AQI 指数	smallInt		√
datetime	记录时间	datetime		√

表 4-3 Weather 数据表

字段	含义	类型	主键	非空
id	用来作为唯一标识的序列，数据库保证其递增且不重复的	int	√	√
cityId	城市 id	int		√
temperature	温度 (°C)	tinyint		√
humidity	相对湿度 (%)	tinyint		√
pressure	大气压强 (hPa)	smallint		√
windspeed	风速 (0.1*M/S)	smallint		√
precipitation	降雨量 (0.1*mm)	smallint		√
weather	天气状态 (1~6)	tinyint		√
datetime	记录时间	datetime		√

此外，气象数据抓取模块需要两个配置文件：城市抓取配置与日志配置。由于 PM2.5 浓度分布历史数据属于国家不公开的信息，目前尚不存在一个统一的全国性质的 PM2.5 浓度发布平台。因此我们需要对每个城市的空气质量部门分别进行 PM2.5 数据抓取，并且为每个监测站标记其相应的 GPS 信息。因此，我们需要为每个城市分别配置城市 PM 数据抓取配置。其格式如图 4-14 所示，其中每个城市的气象站信息都以 XML 格式保存，每个站点信息分别包含 id，名称，经纬度等。

```

<city name="Beijing">
  <site>
    <name>海淀区万柳</name>
    <id>110001</id>
    <gpsLat>39.9810723093</gpsLat>
    <gpsLog>116.2814060586</gpsLog>
  </site>
  <site>
    <name>西直门北</name>
    <id>110002</id>
    <gpsLat>39.9446251288</gpsLat>
    <gpsLog>116.3410692642</gpsLog>
  </site>

```

```

.....
<city>

```

图 4-14 气象数据抓取模块监测站配置格式

由于存在部分监测站存在因人员维修, 保养等导致的 PM2.5 浓度信息无法实时获取, 或第三方气象数据服务器更新, 或因网络中断等不可抗力导致的 PM2.5 信息断点。我们需要日志配置来记录一些抓取失败的 log 信息, 以用来发现因以上问题所导致的 PM2.5 数据丢失。其中, 日志信息主要包含时间, 城市, 站点 id, 抓取失败原因等信息。

客户端在数据预处理过程及估计需要服务器端提供抓取得到的气象及 PM 数据。因此, 需要移动客户端向服务器端发送 JSON 格式的请求以说明其请求的数据类型, 时间, 站点等信息。图 4-15 与图 4-16 分别展示了客户端对气象数据与 PM 数据的 JSON 请求及相应回复格式。

```

/* 客户端请求格式 */
{
/* 请求类型 */
    "request_type" : "pm_list",
/* 位置信息 */
    "request_gps" : {
/* GPS 维度 */
        "gps_lat" : "102.1231",
/* GPS 经度 */
        "gps_log" : "23.4557"},
/* 请求数量 */
    "request_num" : "2",
/* 请求时间列表 */
    "request_datetimes" : [
        {"datetime" : "20140617100823"},
        {"datetime" : "20140618100512"}]
}

/* 服务器端回复格式 */
{
/* 回复类型 */
    "response_type" : "pm_list",

```

```
/* 回复站点 ID */
    "response_site" : "1",
/* 回复数量 */
    "response_num" : "2",
/* 空气质量信息 */
    "response_pms" : [
        {
            /* 时间 */
            "datetime" : "20140617100823",
            /* PM2.5 浓度 (µg/m³) */
            "fpm" : "45",
            /* PM10 浓度 (µg/m³) */
            "cpm" : "78",
            /* 空气质量指数 */
            "aqi" : "110"},
        {
            /* 时间 */
            "datetime" : "20140618100512",
            /* PM2.5 浓度 (µg/m³) */
            "fpm" : "12",
            /* PM10 浓度 (µg/m³) */
            "cpm" : "23",
            /* 空气质量指数 */
            "aqi" : "32"}]
    }
```

图 4-15 PM 数据请求交互示例

```
/* 客户端请求格式 */
{
/* 请求类型 */
    "request_type" : "weather_list",
/* 位置信息 */
    "request_gps" : {
/* GPS 维度 */
        "gps_lat" : "102.12",
/* GPS 经度 */
        "gps_log" : "23.45"},
/* 请求数量 */
    "request_num" : "2",
/* 请求时间列表 */
    "request_datetimes" : [
        {"datetime": "20140617100823"},
        {"datetime": "20140618100512"}]
}

/* 服务器端回复格式 */
{
/* 回复类型 */
    "response_type" : "weather_list",
/* 回复城市 */
    "response_city" : "Guangzhou",
/* 回复数量 */
    "response_num" : "2",
/* 回复天气信息 */
    "response_weahters" : [
        {
/* 时间 */
            "datetime" : "20140617100823",
/* 温度 */
            "temperature" : "24",
/* 湿度*/

```

```
        "humidity" : "34",
        /* 气压 */
        "pressure" : "1004",
        /* 风速 */
        "windspeed" : "34",
        /* 降水量 */
        "precipitation" : "0",
        /* 天气类型 */
        "weather" : "2"},
    {
        /* 时间 */
        "datetime" : "20140617100823",
        /* 温度 */
        "temperature" : "22",
        /* 湿度 */
        "humidity" : "33",
        /* 气压 */
        "pressure" : "1005",
        /* 风速 */
        "windspeed" : "31",
        /* 降水量 */
        "precipitation" : "0",
        /* 天气类型 */
        "weather" : "1"}]
}
```

图 4-16 气象数据请求交互示例

4.4. 开发与运行环境

移动客户端的开发环境如下：

- 后台编程语言：Java（JDK 1.7）
- 集成开发环境：Eclipse Indigo Release
- Android SDK 版本：4.1.2
- OpenCV 版本：2.4.3
- Android 本地编译工具：Android NDK r8c

- 版本控制: git
- 测试机: Samsung S3, HTC G14

服务器的开发环境如下:

- 后台编程语言: Java (JDK 1.7)
- 集成开发环境: Eclipse Indigo Release
- 数据库: MySQL
- 爬虫框架: Weblogic 0.5.2
- Web 端调试工具: Chrome 开发工具
- 版本控制: svn

4.5. 本章小结

本章重点讲述了基于图像的 PM2.5 估计系统的概要设计、详细设计与实现以及开发运行环境。

首先给出了该系统的总体设计架构,并阐述了其所包括的两个主要流程及主要步骤: PM2.5 建模模块与 PM2.5 估计模块。然后分别介绍了客户端与服务器端的各个功能模块的设计和实现方法,包括客户端的应用层,数据层及引擎层的设计及详细模块设计,还有服务器端的气象数据抓取模块,相机模型分析模块的设计及整体数据库设计。最终,介绍了项目客户端和服务器的开发及运行环境。

第五章 实验分析及系统测试

本章详细描述 PM2.5 估计方法的实验设计与分析，系统实现的测试工作，首先介绍实验的总体设计与利用两种估计方法的结果分析。随后，介绍了系统的测试环境，然后分别针对客户端及服务器端进行单元测试，接着对整个系统进行了集成测试，最终展示出客户端的实际演示效果。

5.1. 实验设计与分析

为了验证本课题所提出的基于图像的 PM2.5 估计方法。我们首先利用多种型号的智能手机在不同的地点收集了大量的雾霾图像，利用本文所提出的两种 PM2.5 分析方法，进行了大量的实验。最终，我们展示出了两种方法的实验结果与相应分析。

5.1.1. 实验设计

5.1.1.1. 实验数据集

我们使用收集到的 4 组数据集，及分布在北京不同地点的 PM2.5 检测站得到的 PM2.5 读数^{[37][38]}作为真实值。如图 5-1 所示，目前北京市主要分布着 22 个 PM2.5 监测站点，并且每隔一个小时实时公开各站点 PM2.5 浓度值数据。

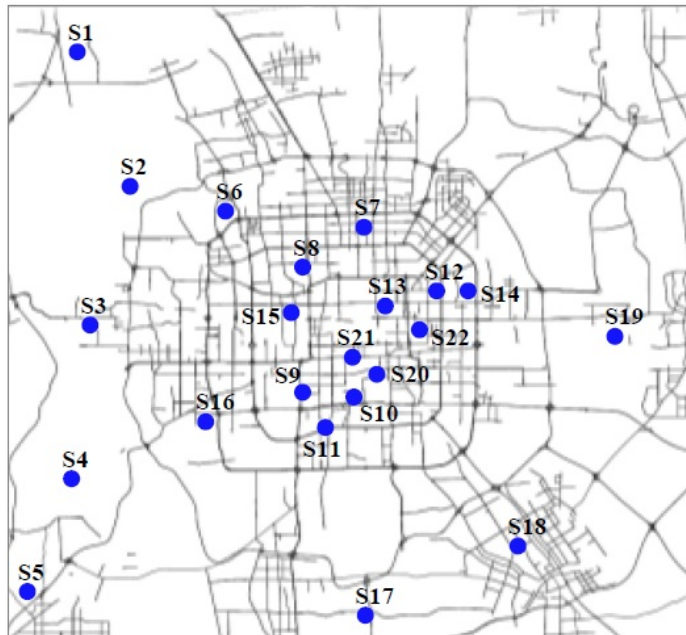


图 5-1 北京市所有 PM2.5 监控站点

所有的图像数据集包含在北京不同地点连续 6 个月的时间收集得到图像，如表 5-1 所示。

表 5-1 PM2.5 估计实验数据集

数据集	数据集 1	数据集 2	数据集 3	数据集 4
地点	场景 1: 校园教三楼	场景 2: BTV 大厦	场景 3: 校园科研楼	场景 4: 校园教三楼
方向	西方	西方	南方	南方
手机型号	三星 S3 HTC G14	数码相机	三星 S3 HTC G14	三星 S3 HTC G14
拍摄时间	14:00	8:00	14:00	14:00
时间跨度	2014.02.27 至 2014.05.13	2014.10.01 至 2014.01.30	2014.02.27 至 2014.05.13	2014.02.27 至 2014.05.13
图像大小	1152 × 2048	560 × 600	1152 × 2048	1152 × 2048
数据集大小	100	100	80	80

其中，数据集 1,3,4 为本课题在北京邮电大学校园内不同的建筑物收集到的图像，与该地点最接近的 PM2.5 监测站为 S8 站点（见图 5-1）。我们使用主要使用了两种手机（Samsung S3, HTC G14）连续 3 个月每天同一个时间段进行图像收集。为了能够在其它地点测试 PM2.5 估计算法，我们同时使用了由邹毅提供的连续 4 个月在北京电视台拍摄的图像^[39]，作为数据集 2。与数据集 2 收集地点最近的监测站为 S22（见图 5-1）。

PM2.5 真实值则是使用北京市环境保护检测中心的实时数据。在基于学习的方法中，将使用其作为 PM2.5 参考值参与估计模型的建立过程，并在测试时作为准确度评估的依据。而在基于对象的方法中，则利用其与 PM2.5 估计值进行比较与结果分析。

5.1.1.2. 实验评价标准

在本课题中，我们使用相关系数（ R^2 ）及平均绝对误差（MAE）来分析 PM2.5 估计方法的准确度。PM2.5 估计值和 PM2.5 真实值的相关系数可由式(5-1)给出。

$$R^2 = 1 - \frac{\sum_{i=1}^n (p'_i - p_i)^2}{\sum_{i=1}^n (p_i - \bar{p})^2} \quad (5-1)$$

其中， p'_i 为第 i 个 PM2.5 估计值， p_i 为第 i 个 PM2.5 真实值。 n 为数据集的大小， \bar{p} 为真实 PM2.5 的平均值。 R^2 的有效范围为 0 至 1。越高的 R^2 表示越高的 PM2.5 估计准确

度。

此外，我们还使用平均绝对误差来描述 PM2.5 估计准确度。其可由式(5-2)给出。

$$MAE = \frac{1}{n} \sum_{i=1}^n |p'_i - p_i| \quad (5-2)$$

其中，平均绝对误差值应大于零，且越接近于零的 MAE 值代表着越高的 PM2.5 估计准确度。

5.1.2. 实验结果分析

5.1.2.1. 基于学习的 PM2.5 估计方法结果分析

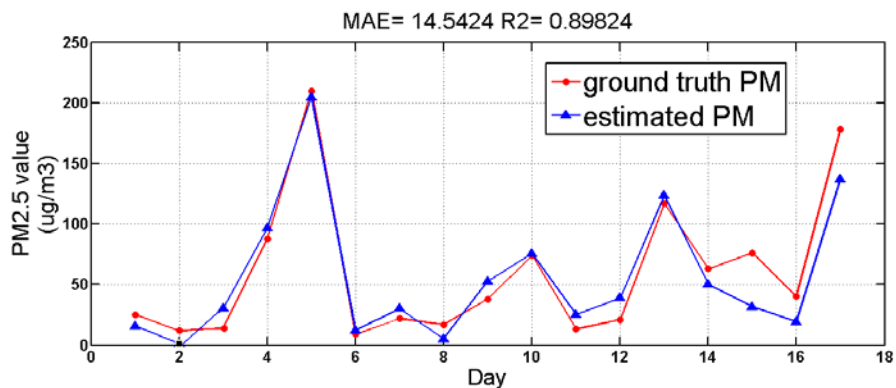
我们使用 R^2 及 MAE 等参数度量了利用 F_{sc} , F_{hsi} 及 F_{dc} 三种图像特征进行 PM2.5 浓度估计的实验结果。表 5-2 总结出了基于学习的 PM2.5 估计方法的实验结果。通过观察，我们可以发现 F_{dc} 特征相比其它的特征拥有最高的 R^2 及最低的 MAE，其在所有的测试地点及测试手机上都拥有最高的性能。因此在系统中，我们选用 F_{dc} 特征作为 PM2.5 估计的图像特征。相比之下， F_{hsi} 特征在三种特征中拥有最差的性能。我们发现 F_{hsi} 特征受到积云的影响较为严重，例如在一些多云或阴天的情况下，积云将遮蔽一半以上的天空，使得相机无法采集到天空实际的色彩特征，导致 HSI 色彩上产生较大偏差。此外，通过与场景 1 中 F_{sc} 与 F_{dc} 的 R^2 较小的差距对比场景 2 中两者较大的差距可以发现， F_{sc} 特征对图像分辨率较为敏感，对低分辨率的图像进行 PM2.5 估计的准确度较差。

表 5-2 LB 方法实验结果总结

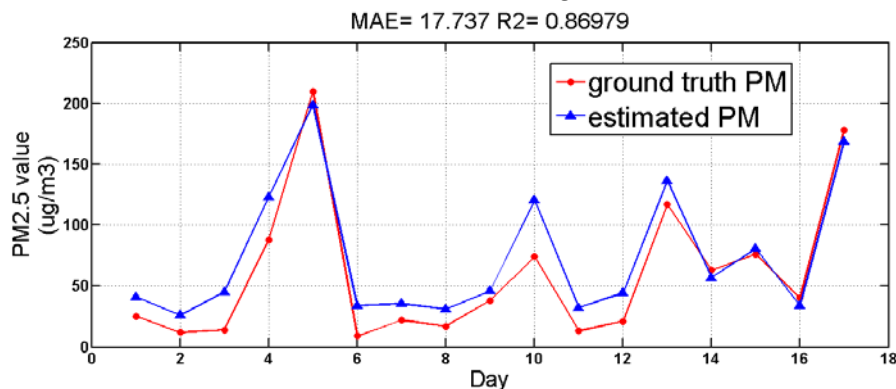
图像特征	场景 1				场景 2	
	HTC G14 手机		Samsung S3 手机		数码相机	
	MAE	R^2	MAE	R^2	MAE	R^2
F_{sc}	27.18	0.8463	27.06	0.8014	31.06	0.6662
F_{dc}	17.73	0.8689	14.54	0.8982	22.23	0.8361
F_{hsi}	33.52	0.7155	30.56	0.6996	32.23	0.6801

图 5-2 比较了在场景 1 中使用不同型号的智能手机及在场景 2 中使用普通数码相机，利用 F_{dc} 特征估计 PM2.5 的实验结果。从中，我们可以发现场景 1 中的 PM2.5 估计值与场景 2 中的估计值相比更为贴近于 PM2.5 真实值。这可能是由于场景 2 中的图像分辨率相比场景 1 中较低所引起的。我们同时还可以发现使用 Samsung S3 所拍摄的图像相比 HTC G14 手机所拍摄的图像估计的 PM2.5 值要更接近于真实值。此外，我们发现场景 1 中使用 HTC G14 手机估计得到的 PM2.5 估计值总是略高于实际值，而场景 2 中使用数

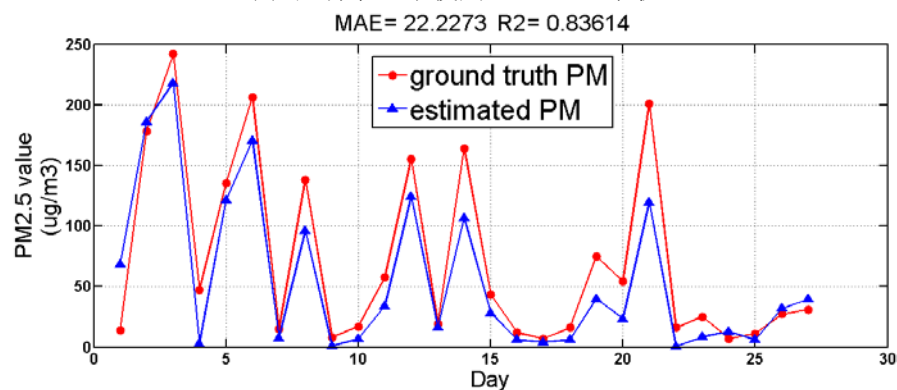
码相机估计得到的 PM2.5 估计值总是率低于实际值,这可能由相机辐射标定所产生的误差所引起的。同时我们还可以发现,在场景 2 使用数码相机所拍摄的图像相比场景 1 所拍摄的图像相比,场景 2 中的估计准确度要更低。我们认为这可能是因为场景 2 中所使用的图像,由于之前没有相机辐射标定模型,而未经过辐射逆变换所导致的。



(a) 在场景 1 中使用 Samsung S3 手机



(b) 在场景 1 中使用 HTC G14 手机



(c) 在场景 2 中使用普通数码相机

图 5-2 在 LB 方法中使用 F_{dc} 特征的 PM2.5 估计结果

接下来,我们调研了用户收集到的图像数据集大小对 PM2.5 估计准确度的影响。直观上来说,越大的数据集大小将会使我们的估计模型拥有更高的准确度。但是,这同时也意味这用户需要更长的时间去收集到更多的图像。因此,我们需要了解如何在收集较小的图像的同时也可以达到较准确的估计结果。图 5-3 展示了 PM2.5 估计结果在包含由

从 3 到 20 个图像的数据集的估计模型。从图中我们可以分析得到，随着数据集大小 m 的增大， R^2 逐渐增大，MAE 逐渐减小，波动也更小。 R^2 及 MAE 参数在数据集大小为 15 左右时逐渐趋于平稳。从实验结果中，我们可以发现半个月的图像训练集即可以建立准确度较高的 PM2.5 估计模型。

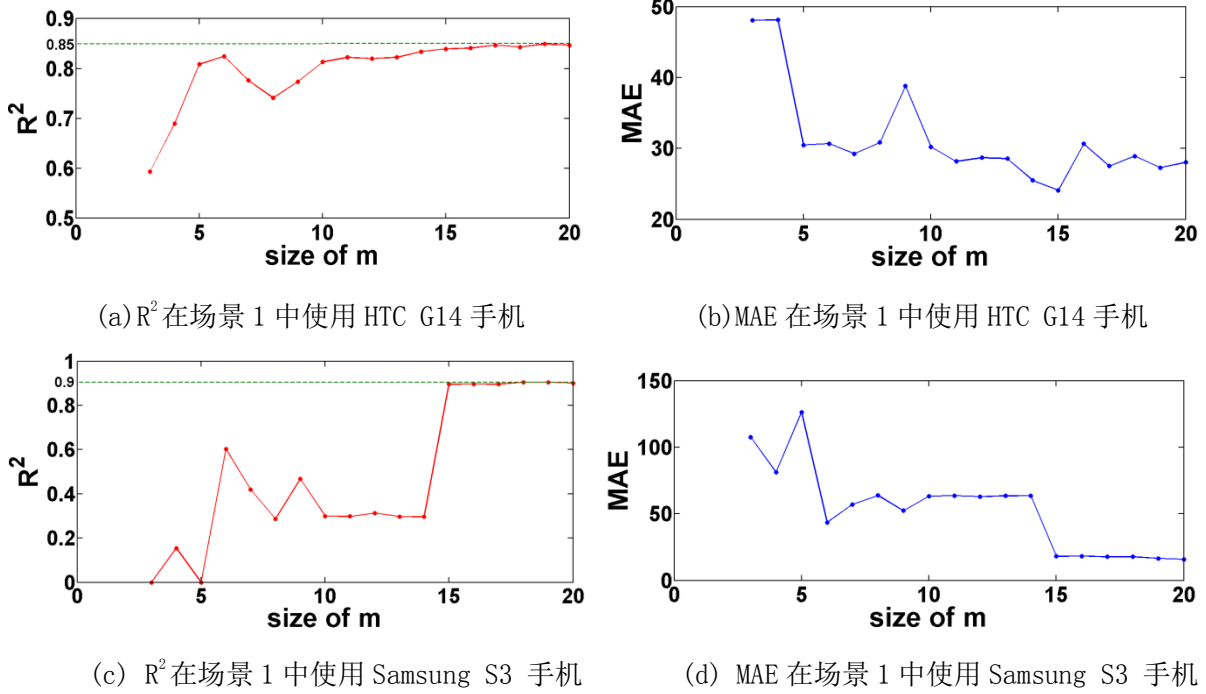
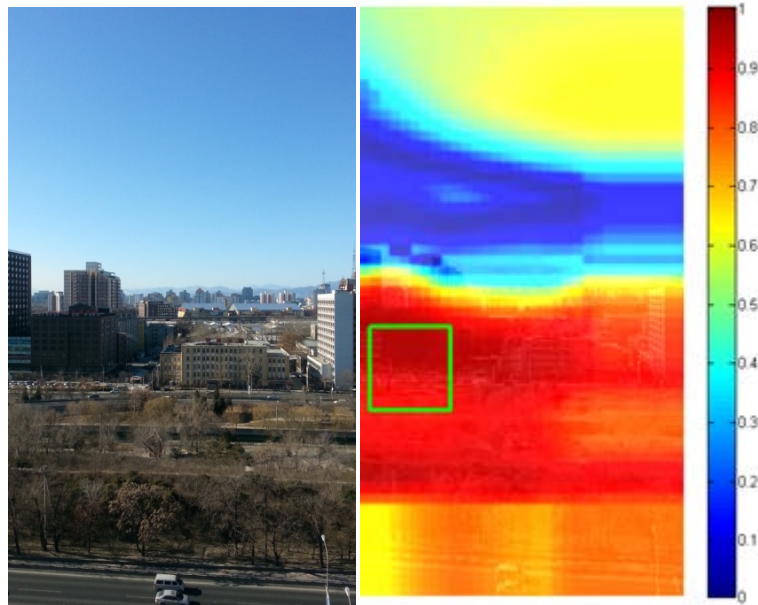
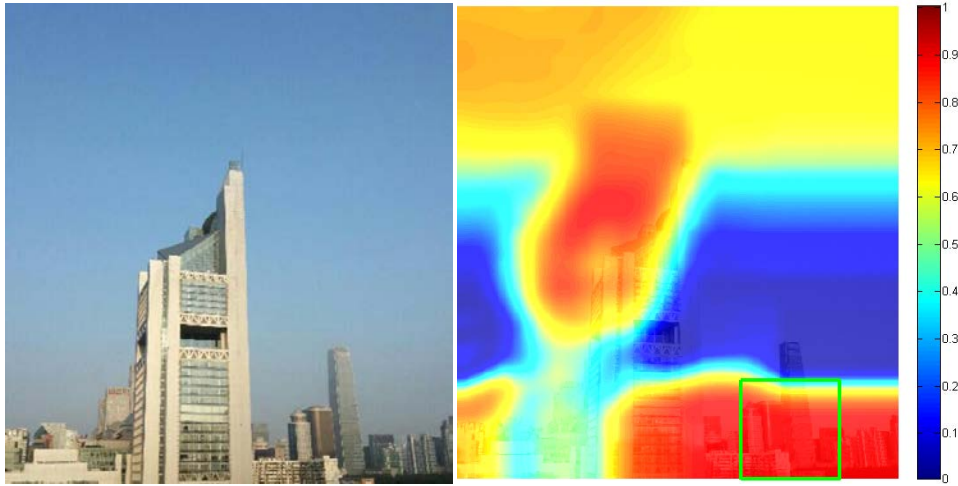


图 5-3 LB 方法的性能随数据集大小的变化

图 5-4 给出了使用 F_{dc} 特征分析的部分场景 PM2.5 相关度分布图。通过观察，我们可以发现具有一定距离的地面物体，如建筑物，树木等，与 PM2.5 浓度呈现较高的相关度。这一结论也恰好与暗原色先验较为吻合。图像中暗通道的低强度值往往是由图像中的具有较多阴影的物体产生，如城市中的汽车，楼房，自然场景中的树木，岩石等^[30]。



(a) 在场景 1 中使用 Samsung S3 手机



(b) 在场景 1 中使用 Samsung S3 手机

图 5-4 LB 方法中使用 F_{dc} 特征分析的部分场景 PM2.5 浓度相关度分布图

5.1.2.2. 基于对象的 PM2.5 估计方法结果分析

我们使用同样的 R^2 及 MAE 标准在基于对象的方法中测试 PM2.5 估计准确度。图 5-5 分别展示了两张在场景 3 和场景 4 中圈画出目标物的图像。表 5-3 与表 5-4 展示了在 OB 方法中使用不同手机型号在不同场景，不同距离下的 PM2.5 估计准确度。在每个场景中，我们分别在图像中圈画出两个距离不同的目标物，并以此计算得到 PM2.5 浓度估计值。从结果中，我们可以发现 OB 方法的平均 R^2 可以达到 0.6639，低于 LB 方法的平均 R^2 ，0.8677。因此，OB 方法相比 LB 方法只可以实现较粗略的 PM2.5 估计。但是，OB 方法为无需训练过程，可以较为快捷的实现 PM2.5 实时建模估计。



(a) 场景 3 与目标物



(a) 场景 4 与目标物

图 5-5 使用 OB 方法在场景 3 与场景 4 拍摄的图像

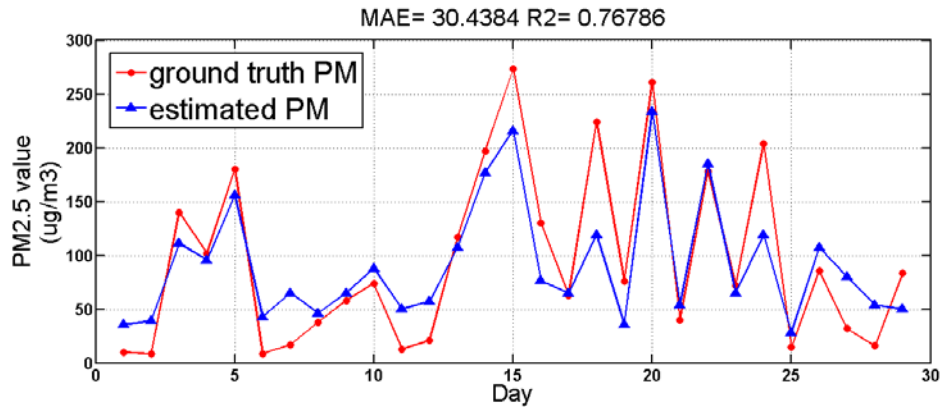
表 5-3 OB 方法在场景 1,2 中的实验结果总结

场景	场景 1				场景 2	
目标物	目标物 1		目标物 2		目标物 1	目标物 2
距离	200m		380m		400m	1300m
手机型号	G14	S3	G14	S3	数码相机	数码相机
MAE	28.44	28.04	22.12	19.33	26.78	31.08
R^2	0.4934	0.5291	0.7478	0.7834	0.7814	0.6278

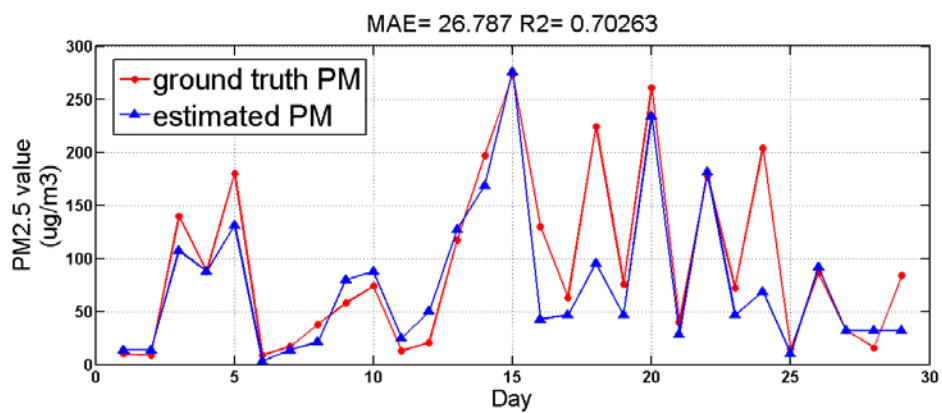
表 5-4 OB 方法在场景 3,4 中的实验结果总结

场景	场景 3				场景 4			
目标物	目标物 1		目标物 2		目标物 1		目标物 2	
距离	250m		750m		300m		650m	
手机型号	G14	S3	G14	S3	G14	S3	G14	S3
MAE	38.30	38.92	40.10	36.29	26.79	30.44	28.62	32.52
R^2	0.6697	0.6637	0.4420	0.5957	0.7026	0.7678	0.7243	0.7582

通过观察表 5-3 与表 5-4，我们还可以发现虽然 OB 方法的平均 R^2 低于 0.7，但是在距离为 300 米至 650 米的区间中平均 R^2 可高于 0.7。这说明通过使用距离适中的建筑物作为目标物可以提高估计准确度。图 5-6 展示了在场景 4 中利用连续 29 天分别使用 Samsung S3 手机与 HTC G14 手机收集的图像进行 PM_{2.5} 估计的 PM_{2.5} 估计值与真实值对比。此外，通过对各个场景中不同手机实验的结果，可以发现使用 HTC G14 手机相比使用 Samsung S3 手机的估计准确度较低，此处可能是由 HTC G14 手机所使用的辐射逆变换函数所引入的误差导致。最后，通过与 LB 方法的结果比较，我们可以发现 LB 方法可以提供比 OB 方法更准确的 PM_{2.5} 估计。这是因为 OB 方法的误差主要来源于式 (3-3) 中对参数 p 的确定。因此，在后续研究工作中需要考虑不同地域中 p 值的确定方法，从而提升基于对象方法的估计性能。



(a) 在场景 4 中使用 Samsung S3 手机



(b) 在场景 4 中使用 HTC G14 手机

图 5-6 在 OB 方法中使用 F_{dc} 特征的 PM2.5 估计结果

5.2. 系统测试与效果展示

本课题的系统测试环境为基础设备，移动客户端与系统后台服务器。具体的测试环境如图 5-7 所示。

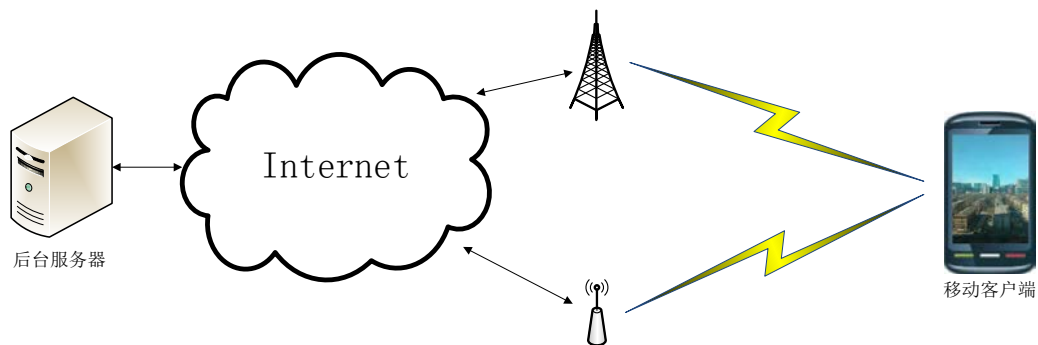


图 5-7 系统测试环境

其中，基础设施主要包括运营商提供的移动网络基站与 WIFI 接入点。后台服务器主要为 PM2.5 分析提供相关气象数据抓取，存储等服务。移动客户端为使用嵌入了 PM2.5 估计模块的客户端程序。本科题主要通过 PM2.5 建模与 PM2.5 估计这两个过程来进行客户端的测试与效果展示。

5.2.1. 功能测试

5.2.1.1. PM2.5 建模过程

● 功能性测试

表 5-5 PM2.5 建模过程功能性测试

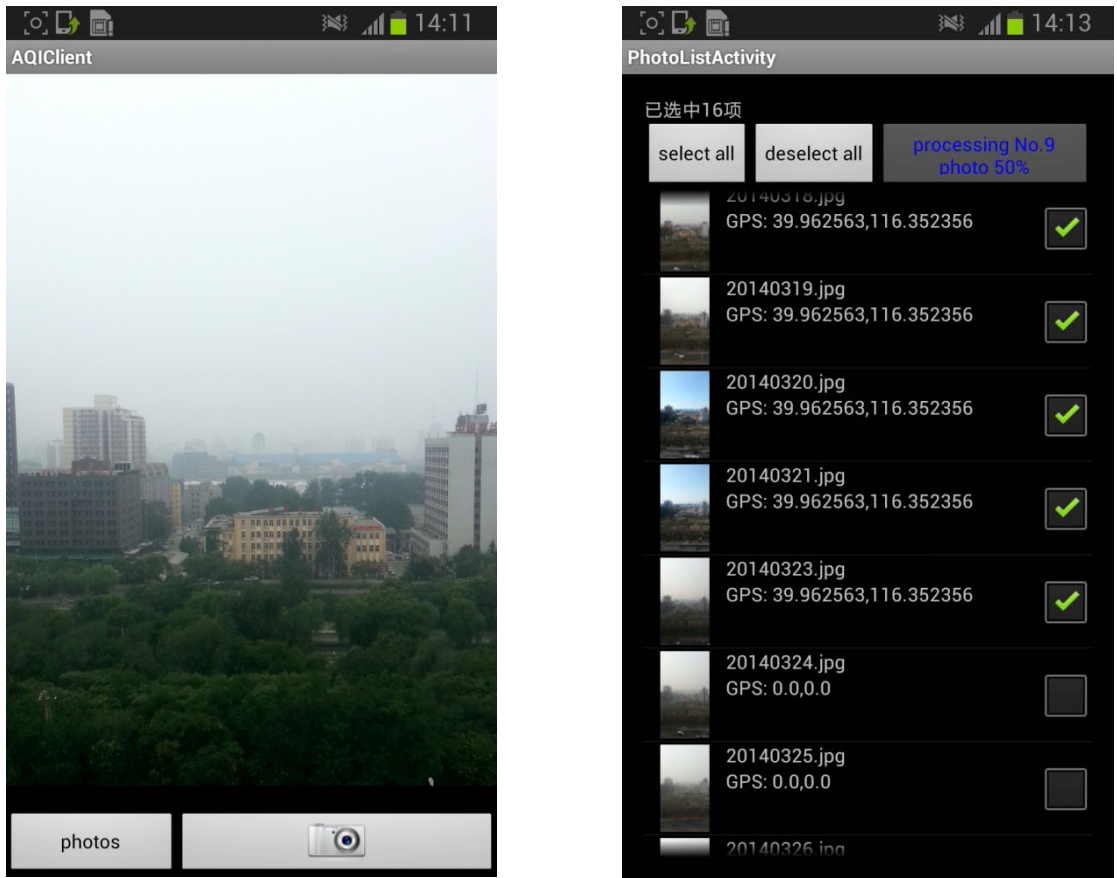
用例编号	测试用例	预测结果	测试结果
1-1-1	图像预处理	图像利用相机模型进行辐射逆变换，并且在临时文件夹生成特征图像。	符合
1-1-2	数据预处理	若训练集达到一定数目，向后台服务器请求相应气象数据，并进行训练集过滤，得到有效的训练数据。	符合
1-1-3	PM2.5 模型分析	使用经过过滤的有效训练集图像数据及相应参考 PM2.5 数据，进行建模过程。输出中间进度数据。	符合
1-1-4	PM2.5 模型生成	进行 PM2.5 模型分析后，生成有效的 YML 格式模型文件。	符合

● 容错性测试

表 5-6 PM2.5 建模过程容错性测试

用例编号	测试用例	预测结果	测试结果
1-2-1	相机模型不存在	从用户相册中抽取一张图像发送至服务器进行相机模型分析。成功分析后返回相机辐射模型。	符合
1-2-2	无法获取后台服务器数据	重新发送请求若干次，如无法获取，返回失败信息。	符合

为了测试以上功能，我们在移动客户端软件中分别测试了以上过程。图 5-8(a)展示了雾霾照片拍摄过程，用户只需要针对某一场景点击拍摄按钮即可。图 5-8(b)展示了 PM2.5 模型分析过程，用户只需要点选一定数量的照片加入到训练集中即可执行模型分析过程，图中右上边的字显示着建模的进度信息。



(a) 雾霾图像采集 (b) PM2.5 建模

图 5-8 PM2.5 建模过程效果展示

5.2.1.2. PM2.5 估计过程

● 功能性测试

表 5-7 PM2.5 估计过程功能性测试

用例编号	测试用例	预测结果	测试结果
2-1-1	图像预处理	图像利用相机模型进行辐射逆变换，并且在临时文件夹生成特征图像。	符合
2-1-2	PM2.5 估计	利用已有的 PM2.5 估计模型进行 PM2.5 浓度估计，并得到估计值。	符合

● 容错性测试

表 5-8 PM2.5 估计过程容错性测试

用例编号	测试用例	预测结果	测试结果
2-2-1	气象信息获取失败	试图重新发送请求若干次，如无法获取，返回失败信息。	符合
2-2-2	PM2.5 估计模型不存在	错误提示模型不存在详细信息	符合

为了测试以上功能，我们在移动客户端软件中分别测试了以上过程。图 5-9(a)展示了 PM2.5 估计分析过程，用户只利用之前建立好的 PM2.5 估计模型即可实现 PM2.5 浓度估计，图中蓝色的字为新拍的图像的最终 PM2.5 估计值。图 5-9(b)为利用本课题后台服务器端抓取的抓取的离拍摄地点最近的监测站点，西直门北监测站的实时 PM2.5 真实值作对比。从图中可以发现，利用本课题所提出的 PM2.5 估计算法得到的 PM2.5 浓度估计值为 $203\mu\text{g}/\text{m}^3$ ，而真实的 PM2.5 浓度值为 $205\mu\text{g}/\text{m}^3$ 。因此，我们可以认为，该 PM2.5 估计系统可以较为可靠地利用手机所拍摄的图像实现 PM2.5 浓度估计。



(a) PM2.5 估计



(b) PM2.5 真实值

图 5-9 PM2.5 估计过程效果展示

5.2.2. 性能测试

根据以上功能的测试,本课题还进行了基于图像的 PM2.5 估计系统的客户端具体性能测试,包括能量消耗与通信开销。我们将本课题所实现的分布式系统与第四章所提到的集中式架构进行了比较。在集中式架构中,所有的图像预处理过程,特征提取过程,PM2.5 估计过程将会在后台服务器进行。移动客户端只需要上传原始的图像数据。相比之下,本课题所实现的分布式系统结构将有效利用智能手机的计算能力实现图像预处理,特征提取过程与 PM2.5 估计过程。用户无需上传个人收集的图像,其可以有效的节省用户昂贵的通信开销,同时也保护了用户个人隐私。

我们分别在不同型号的手机上基于 Android 4.1.2 系统进行了性能测试,我们利用 PowerTutor^[40]测试软件分别在 Samsung S3 与 HTC G14 两种型号的手机中进行了 20 次 PM2.5 建模与估计测试,并记录了其平均的能量消耗与通信开销,如表 5-9 所示。

表 5-9 系统能量消耗与通信开销测试总结

系统架构		分布式		集中式	
方法		LB 方法	OB 方法	LB 方法	OB 方法
能量 消耗	CPU	24.1J	1.4J	4.7J	0.6J
	3G 通信	3.9J	4.3J	92.7J	7.2J
	总消耗 (总电量%)	28.0J (0.097%)	5.7J (0.019%)	97.4J (0.339%)	7.8J (0.027%)
通信开销		3.9KB	1.6KB	10262KB	693KB
耗时		5.1s	8.1s	32.8s	12.2

通过观察测试结果,我们可以发现分布式架构拥有更高的 CPU 计算能量消耗,而集中式架构在 3G 通信上消耗更多的能量。但是,分布式架构在总的能量消耗上要低于集中式架构,只占用了智能手机电量的极少部分(小于 0.097%)。此外,我们发现在两种架构中 LB 方法相比 OB 方法消耗更多的能量,其是由于 LB 方法需要在建模过程中利用更多的计算资源。LB 方法在分布式架构下耗时相比 OB 方法更少是由于 LB 方法中在每张照片拍摄时便计算器特征图像,在建模时便节省了大量时间,仅仅在需要执行回归分析算法。而 LB 方法在集中式架构下相比 OB 方法却消耗了更多的时间,我们分析其原因是由于 LB 方法需要上传所有的训练集数据所引起的网络传输时间损耗。在分布式架构中,通信消耗相比集中式将会更少,因为无需上传图像数据。在 LB 与 OB 方法中少于 4KB 与 2KB 的数据将会传送到服务器端。相比之下,集中式将最多上传接近 10MB 与 0.7MB 的数据量,如此大的数据量传输将会导致较多的传输能量消耗与无线通信费用。因此,分布式架构对于本系统是一种更优的选择。

5.3. 本章小结

本章通过大量的 PM2.5 图像估计算法实验及 PM2.5 系统的测试与效果演示展示了本课题所设计并实现的基于图像的 PM2.5 估计系统的有效性与可行性。

首先，本章先介绍了 PM2.5 估计算法的实验设计，分别设计对比了不同图像特征对 PM2.5 算法估计结果的影响，及使用基于学习的估计方法（LB 方法）与基于对象的估计方法（OB 方法）的性能对比。通过利用本科题收集到的使用不同手机型号在不同地点拍摄的 PM2.5 图像数据集所进行的大量实验，我们可以得知本系统所提出的 PM2.5 估计算法可以有效地实现大范围，细粒度的 PM2.5 浓度估计。

其次，本章节还阐述了系统测试结果与性能分析结果，并且对比了本系统所使用的分布式结构与集中式结构的性能。最终通过测试结果可以说明了基于图像的 PM2.5 估计系统的有效性与可行性。

第六章 总结和展望

6.1. 全文总结

本论文设计并实现了一个基于参与式感知的空气质量检测系统。整个系统利用参与者在不同地点利用不同手机收集到的雾霾图像进行 PM2.5 估计，最终得到城市大范围，细粒度的 PM2.5 浓度分布图。本课题的主要工作内容及贡献为：

1、本课题所实现的 PM2.5 估计方法是目前已知学术界第一个被提出的利用手机拍摄的图像进行 PM2.5 浓度估计的方法。此种方法提供了一种充分利用智能手机感知与计算能力在大范围区域中低成本，细粒度的 PM2.5 浓度检测方法。

2、本课题提出了两种基于图像的 PM2.5 估计算法：基于学习的估计方法(LB 方法)与基于对象的估计方法 (OB 方法)。LB 方法可以使用用户在同一地点不用时间所拍摄的一系列图像与相应的最近空气质量监测站提供的 PM2.5 参考值进行训练学习，并生成 PM2.5 估计模型，从而较为精准的估计出 PM2.5 浓度值。而 OB 方法直接使用用户所拍摄的照片及目标物信息进行实时的 PM2.5 浓度估计，其需要较多的用户交互过程，且估计结果相比 LB 方法较差。

3、本课题最终利用 LB 方法实现了 PM2.5 估计系统，并在移动客户端上进行了大量的实验与测试，其结果说明了本课题所提出的 PM2.5 估计方法的有效性与可行性。本科题收集了4个在北京不同地点使用不同型号的手机拍摄的雾霾图像及相应PM2.5参考值。并且在得到的数据集上进行大量的测试表明，本课题的 PM2.5 估计系统可以达到较高 PM2.5 估计准确度，可以有效实现较为精确的空气质量检测。

论文首先对空气质量检测及参与式感知的相关理论和技术背景以及系统开发中所使用的开发技术进行介绍；之后详细描述了基于图像分析的 PM2.5 估计方法，并且分别阐述了 LB 与 OB 方法的实现原理与异同比较。随后，论文继续介绍了 PM2.5 估计系统的总体设计，架构及模块设计。接下来分别描述了系统客户端与服务器端的详细设计与内部各个子模块的设计与实现；最后，通过在数据集上进行大量的实验对 PM2.5 估计算法进行了实验分析，并且对最终系统进行了系统测试及效果展示，从而表明整个系统的运行结果达到了预期的设计目标。

6.2. 未来工作展望

在未来的工作中，还需要从以下几个方面来改善本课题所提出的基于图像的 PM2.5 估计系统：

- 改进 PM2.5 分析算法，结合更多的图像特征，并尝试组合各个特征进行实验对比，从而进一步提升估计准确度。
- 考虑更多的外界因素对 PM2.5 估计算法的影响，如不同的区域气象环境条件，手机其它的硬件参数等。
- 在不同的地域激励参与者收集更多的图像数据及相应气象数据，以验证本系统在不同地域，不同场景的可行性。
- 进一步深入进行误差分析，对可能的解决方案进行实验，从而改进算法，进一步提升估计准确度。

6.3. 研究生期间工作

在研究生期间，论文作者参与的工作主要有：

一、参加 “智能图书馆” 项目

- 负责人脸识别算法调研
- 负责人脸识别原型子系统的客户端设计与实现
- 负责人脸识别原型子系统的测试与优化工作

二、参加 “BNRC 项目管理系统” 项目

- 负责 3 人项目团队的协调工作
- 参与前期的技术调研和需求分析
- 负责系统的总体架构的设计和详细设计
- 负责系统的 PHP 后台核心库的设计与实现
- 负责系统数据库的设计与实现
- 参与系统的前端页面设计与实现
- 负责系统的部署和测试
- 负责系统的版本控制工作

三、参加 “移动协作感知” 项目

- 参与移动协作感知系统客户端的需求分析与设计
- 负责 PM2.5 分析模块的前期学术调研与需求分析
- 负责 PM2.5 分析模块总体架构设计与实现
- 负责 PM2.5 分析算法的实验设计与分析
- 负责 PM2.5 分析算法的数据集收集与整理
- 参与移动协作感知系统的后台数据库设计
- 参与移动协作感知系统的部分业务模块设计与实现

四、参与诺基亚北京研发中心项目 “Video Wand”

- 负责短视频 3D 展示模块的改进与优化

- 参与短视频特效渲染模块的改进工作
- 五、从诺基亚内部申请国际专利 “Object-level Video Summarization Using Eye Information”
- 已通过国际专利申请，专利申请号为 PCT/CN2014/073120
 - 负责专利相关技术调研及专利文档撰写
- 六、以第一作者身份在 IEEE INFOCOM Workshops 2015 发表学术论文一篇，题目是《PM2.5 Monitoring using Images from Smartphones in Participatory Sensing》，已录用，待 EI 检索。

参考文献

- [1] 北京 2014 年上半年空气质量状况 <http://www.bjmemc.com.cn/g327/s962/t2037.aspx>, 2014.
- [2] Burke J A, Estrin D, Hansen M, et al. Participatory sensing[J]. Center for Embedded Network Sensing, 2006.
- [3] Goldman J, Shilton K, Burke J, et al. Participatory Sensing: A citizen-powered approach to illuminating the patterns that shape our world[J]. Foresight & Governance Project, White Paper, 2009: 1-15.
- [4] Kanhere S S. Participatory sensing: Crowdsourcing data from mobile smartphones in urban spaces[C]//Mobile Data Management (MDM), 2011 12th IEEE International Conference on. IEEE, 2011, 2: 3-6.
- [5] Rana R K, Chou C T, Kanhere S S, et al. Ear-phone: an end-to-end participatory urban noise mapping system[C]//Proceedings of the 9th ACM/IEEE International Conference on Information Processing in Sensor Networks. ACM, 2010: 105-116.
- [6] Ganti R K, Pham N, Ahmadi H, et al. GreenGPS: a participatory sensing fuel-efficient maps application[A]//Proceedings of the 8th international conference on Mobile systems, applications, and services. ACM, 2010: 151-164.
- [7] Azizyan M, Constandache I, Roy Choudhury R. SurroundSense: mobile phone localization via ambience fingerprinting[C]//Proceedings of the 15th annual international conference on Mobile computing and networking. ACM, 2009: 261-272.
- [8] Hasenfratz D, Saukh O, Sturzenegger S, et al. Participatory air pollution monitoring using smartphones[J]. Mobile Sensing, 2012.
- [9] 大气能见度 <http://en.wikipedia.org/wiki/Visibility>, 2014.
- [10] 消光系数 http://en.wikipedia.org/wiki/Extinction_coefficient, 2014.
- [11] 大气浊度 <http://en.wikipedia.org/wiki/Turbidity>, 2014.
- [12] 朱李华, 陶俊, 陈忠明, 等. 2010 年 1 月北京城区大气消光系数重建及其贡献因子[J]. 环境科学, 2012, 33(1): 13-19.
- [13] 王京丽, 刘旭林. 北京市大气细粒子质量浓度与能见度定量关系初探[J]. 气象学报, 2006, 64(2): 221-228.
- [14] Dutta P, Aoki P M, Kumar N, et al. Common sense: participatory urban sensing using a network of handheld air quality monitors[C]//Proceedings of the 7th ACM conference on embedded networked sensor systems. ACM, 2009: 349-350.
- [15] Devarakonda S, Sevusu P, Liu H, et al. Real-time air quality monitoring through mobile sensing in metropolitan areas[C]//Proceedings of the 2nd ACM SIGKDD International Workshop on Urban Computing. ACM, 2013: 15.
- [16] Kim S, Paulos E. inAir: measuring and visualizing indoor air quality[C]//Proceedings of the 11th international conference on Ubiquitous computing. ACM, 2009: 81-84.
- [17] Xie L, Chiu A, Newsam S. Estimating atmospheric visibility using general-purpose cameras[M]//Advances in Visual Computing. Springer Berlin Heidelberg, 2008: 356-367.
- [18] Graves N, Newsam S. Using visibility cameras to estimate atmospheric light

- extinction[C]//Applications of Computer Vision (WACV), 2011 IEEE Workshop on. IEEE, 2011: 577-584.
- [19] Kim K W, Kim Y J. Perceived visibility measurement using the HSI color difference method[J]. Journal of the Korean Physical Society, 2005, 46(5): 1243-1250.
- [20] HSI http://en.wikipedia.org/wiki/HSI_color_space, 2014.
- [21] Liaw J J, Lian S B, Huang Y F, et al. Using sharpness image with Haar function for urban atmospheric visibility measurement[J]. Aerosol and Air Quality Research, 2010, 10(4): 323-330.
- [22] Liaw J J, Lian S B, Huang Y F, et al. Atmospheric visibility monitoring using digital image analysis techniques[C]//Computer Analysis of Images and Patterns. Springer Berlin Heidelberg, 2009: 1204-1211.
- [23] Perez R, Seals R, Michalsky J. All-weather model for sky luminance distribution—preliminary configuration and validation[J]. Solar energy, 1993, 50(3): 235-245.
- [24] Poduri S, Nimkar A, Sukhatme G S. Visibility monitoring using mobile phones[J]. Annual Report: Center for Embedded Networked Sensing, 2010: 125-127.
- [25] Koschmieder H. Theorie der horizontalen Sichtweite: Kontrast und Sichtweite[M]. Keim & Nemnich, 1925.
- [26] Fattal R. Single image dehazing[C]//ACM Transactions on Graphics (TOG). ACM, 2008, 27(3): 72.
- [27] Tan R T. Visibility in bad weather from a single image[C]//Computer Vision and Pattern Recognition, 2008. CVPR 2008. IEEE Conference on. IEEE, 2008: 1-8.
- [28] Kopf J, Neubert B, Chen B, et al. Deep photo: Model-based photograph enhancement and viewing[C]//ACM Transactions on Graphics (TOG). ACM, 2008, 27(5): 116.
- [29] Narasimhan S G, Nayar S K. Contrast restoration of weather degraded images[J]. Pattern Analysis and Machine Intelligence, IEEE Transactions on, 2003, 25(6): 713-724.
- [30] He K, Sun J, Tang X. Single image haze removal using dark channel prior[J]. Pattern Analysis and Machine Intelligence, IEEE Transactions on, 2011, 33(12): 2341-2353.
- [31] Grossberg M D, Nayar S K. What is the space of camera response functions?[C]//Computer Vision and Pattern Recognition, 2003. Proceedings. 2003 IEEE Computer Society Conference on. IEEE, 2003, 2: II-602-9 vol. 2.
- [32] Lin S, Gu J, Yamazaki S, et al. Radiometric calibration from a single image[C]//Computer Vision and Pattern Recognition, 2004. CVPR 2004. Proceedings of the 2004 IEEE Computer Society Conference on. IEEE, 2004, 2: II-938-II-945 Vol. 2.
- [33] Ozkaynak H, Schatz A D, Thurston G D, et al. Relationships between aerosol extinction coefficients derived from airport visual range observations and alternative measures of airborne particle mass[J]. Journal of the Air Pollution Control Association, 1985, 35(11): 1176-1185.
- [34] Nayar S K, Narasimhan S G. Vision in bad weather[C]//Computer Vision, 1999. The Proceedings of the Seventh IEEE International Conference on. IEEE, 1999, 2: 820-827.
- [35] Van Herk M. A fast algorithm for local minimum and maximum filters on rectangular and octagonal kernels[J]. Pattern Recognition Letters, 1992, 13(7): 517-521.
- [36] Webmagic <http://webmagic.io/>, 2014.
- [37] 北京市环境保护监测中心, <http://zx.bjmemc.com.cn/>, 2014.
- [38] 美国使馆环境监测, <https://twitter.com/BeijingAir/>, 2014.
- [39] 邹毅, 雾霾图像数据集, <http://weibo.com/p/1005051000481815/>, 2014.
- [40] PowerTutor, <http://ziyang.eecs.umich.edu/projects/powertutor/>, 2014.

致 谢

转眼间，两年半的研究生生活即将结束。在此，我要向所有在本人研究生期间学习、生活以及工作上给予我帮助、支持及关心的人致以最诚挚的感谢。

首先，我要向我的导师王文东教授表达诚挚的感谢。王老师虽然日常工作繁忙，但是经常在我日常学习与生活中给予我指导和悉心教诲。在我遇到困难的时候，通过与王老师的交流与讨论，使我发现了存在的问题，并找到问题的解决方法，让我时刻保持认真的态度。在此研究生生活即将结束之际，我谨向尊敬的导师王文东教授表达我最真诚的谢意！

同时，我还要感谢龚向阳老师和阙喜戎老师，他们在日常生活中帮助我解决了很多困难，在科研上也给予了我很多的指导与支持，并对我的项目及论文工作提出了许多宝贵的意见与建议，在此，我要向龚老师和阙老师表示由衷的谢意。

感谢宋峥、田野等博士师兄，他们在我的学术研究，专利及论文上给予了很大的支持，感谢他们无私的帮助。感谢项目组的其它同学，特别是赖海腾，袁龙运，沈跃辉，冯云，王东升等同学。感谢他们在学习及生活中的相互理解与帮助，使得课题任务圆满完成。在他们身上我学到了很多，同时也结下了深厚的友谊。

此外，我还要感谢我的家人，好友和同学们在研究生期间对我最无私的理解，支持和鼓励。

最后，感谢各位评委老师在百忙之中抽出了宝贵的时间，评审我的学位论文，并给予指导。

作者攻读学位期间发表的学术论文目录

- [1] 刘肖阳. 一种基于视频的二维码组合方法. 已被中国科技论文在线刊载录用, 属于规定的核心学术期刊, 署名单位为北京邮电大学
- [2] X. Liu, Z. Song, E. C.-H. Ngai, J. Ma, and W. Wang, PM2.5 monitoring using images from smartphones in participatory sensing, IEEE Computer Communications Workshops (INFOCOM WKSHPS), 2015. (EI 收录, 第一作者, 录用待发表, 论文署名单位为北京邮电大学)