

## 摘要

随着经济的发展，人们因为从事经济活动，需要经常出入林区，对于火灾隐患的控制非常困难。尤其是初次出入林区的人防火意识薄弱，让防火变得难上加难。传统的人工巡逻或者视频监控方案只能够在火灾发生到一定程度甚至是林区大面积过火的时候才能发出警报。这类监控无法检测早期阶段的火灾，非常容易造成巨大的损失。本课题通过分析森林中的复杂环境，设计了一种红外森林防火监控系统。该系统可以检测早期火灾，并且能够排除浓雾等恶劣天气的影响，对森林火灾的扑救有很大的现实作用。

本文主要研究了红外森林防火监控系统中监控软件的设计和实现。首先，阐述了红外辐射、红外成像以及红外成像仪等基本知识，并结合红外成像技术与森林防火，分析了运用红外成像检测森林的依据。然后，通过算法对比并借鉴早期的林火识别算法，同时考虑红外采集设备的特性，在前人基础上设计了采用混合高斯模型的林火识别算法并对该算法做了验证。其次，通过分析系统的整体需求，并对比了传统的视频监控系统，提出了红外森林防火监控系统的方案，并阐述了该方案中软硬件模块以及软件的需求分析。之后，在考虑到软件的健壮性、可移植性以及可扩展性的条件下，使用面向对象建模思想，划分了软件的各个功能模块，再运用 UML 语言描述了各个功能模块。在界面模块的实现过程中，利用独立的界面库美化了软件界面。在鉴权模块中，运用网络编程发消息的方式实现了客户端到监控中心服务器的鉴权。在参数配置模块中，使用面向对象的继承特性，定义了内部类和外部类，提高了代码复用率。在林火识别模块中，通过结合红外采集设备的自身优势，实现了林火识别算法。在分析报告、数据库、视频播放、日志模块的设计和实现过程中都体现了面向对象的编程思想，提升了软件的可靠性、复用性，降低了开发周期。接着，搭建了测试场景，对各个模块和系统做了测试和分析，测试结果表明软件运行良好，烟火识别准确，达到了监控森林的要求。最后，对全文工作做了总结并且提出了改进方案。

**关键词：**森林防火监控系统，红外成像仪，混合高斯模型，面向对象



## Abstract

With the development of economy, people have to frequently visit forest, so the fire hazard is very difficult to control. Especially the person who was the first time to visit forest is weak in the fire prevention awareness, so the prevention from fire has become more important. Traditional manual patrols or video surveillance solution can only alarm when the forest fire become large. Such monitoring method can not detect the early stages of the fire, which will lead to huge losses. Through analyzing the complex environment, the topic designed a forest fire infrared surveillance systems is designed. The systems can detect the early fire, and has the ability to exclude rain, snow, fog and other inclement weather, which play an important role in the fighting of forest fires.

This paper studies on the infrared forest fire monitoring system design and implementation of software components. Firstly, described the infrared radiation, infrared imaging and infrared imaging devices and other basic knowledge, combined with infrared imaging technology and forest fire prevention, the use of infrared imaging detection analysis based on the forest. Secondly, by comparison algorithm and consider the fire of early recognition algorithms, taking into account the characteristics of infrared acquisition equipment, designed Gaussian mixture model of forest fire recognition algorithms and verified the algorithm. Thirdly, by analyzing the overall system requirements and compare with the traditional video surveillance system, put forward the forest fire infrared surveillance system programs, and explains each part of the program. Fourthly, after taking into account the robustness of the software and scalability of the conditions, making the overall structure of the software system design and software functional requirements analysis. Using object-oriented software method and modeling idea divide the functional modules of the software, then using the UML language described the various functional modules. In the implementation of the interface module, a separate interface library is used to beautify the software interface. In the authentication module, using network programming message to achieve the client to the monitoring center server authentication. In the parameter configuration module, using the inheritance feature of object-oriented, defining the internal classes and external classes to improve code reuse rate. Recognition module in the fire, collected

through a combination of infrared equipment to its own advantages, to achieve a forest fire recognition algorithms. In the analysis, databases, video player, log module design and implementation are reflected the object-oriented programming ideas in order to improve the software reliability, reusability and reduces development cycle. Then, set up a test scenario, tested and analysed the modules and systems, and show that the software can recognize fire and smoke, so it reaches the require of fire detection. Finally, an improved solution is put forward after summarizing the full text of the work.

**Key Words:** Forest Fire Monitoring System, Infrared Cameras, Gaussian Mixture Model, Object-Oriented



## 插图清单

## 第二章

图 2.1 红外成像模型.....	6
-------------------	---

## 第三章

图 3.1 不同阈值 $\Delta$ 下差分分析法效果.....	12
图 3.2 单高斯模型运行效果.....	13
图 3.3 林火识别算法流程.....	16
图 3.4 滤波中的边界问题（以 3×3 的窗口为例）.....	17
图 3.5 中值滤波的拓展方式.....	17
图 3.6 分布类型判别流程.....	19
图 3.7 混合高斯模型的参数更新.....	21
图 3.8 新高斯分布建立流程.....	23
图 3.9 可疑区域的判决.....	25
图 3.10 目标提取流程.....	27
图 3.11 林火识别算法效果图.....	28

## 第四章

图 4.1 森林防火监控系统结构模型.....	34
图 4.2 系统硬件结构图.....	36
图 4.3 DVS 和 DVR 的方案对比.....	38
图 4.4 视频传输模型.....	40
图 4.5 软件需求用例图.....	41
图 4.6 红外森林防火监控软件模型.....	42
图 4.7 系统资源及设备管理.....	43
图 4.8 林火监控系统的数据流信息.....	44
图 4.9 客户端消息流程图.....	48

## 第五章

图 5.1 鉴权模块流程.....	51
图 5.2 参数配置文件.....	54
图 5.3 参数配置模块流程.....	55
图 5.4 林火识别模块与其他模块的关系.....	57
图 5.5 中值滤波流程.....	58

图 5.6 林火监控系统分析报告 UML 序列图 ..... 62

图 5.7 林火监控报告流程 ..... 62

图 5.8 UML 描述 E-R 模型 ..... 64

图 5.9 日志模块流程图 ..... 68

第六章

图 6.1 测试场景模型 ..... 71

图 6.2 主界面的测试效果 ..... 73

图 6.3 登陆鉴权效果图 ..... 73

图 6.4 参数配置模块验证 ..... 74

图 6.5 没有烟火时的效果 ..... 74

图 6.6 林火识别模块检测效果 ..... 75

图 6.7 林火分析报告的效果图 ..... 76

图 6.8 数据库模块验证 ..... 76

图 6.9 日志文件 System.log 的内容 ..... 77

## 插表清单

## 第四章

表 4.1 林火监控系统方案比较 .....	33
表 4.2 云台协议 .....	35
表 4.3 云台控制字段 .....	35

## 第五章

表 5.1 登陆消息表 .....	52
表 5.2 林火监控软件的参数 .....	54
表 5.3 CFileConfig 类的成员函数 .....	56
表 5.4 CGaussMixModel 的成员变量和成员函数 .....	58
表 5.5 RPT 文件访问函数 .....	63
表 5.6 本地区域信息 (LocalAreaInfo) .....	64
表 5.7 基站信息 (StationInfo) .....	64
表 5.8 用户信息 (UserInfo) .....	65
表 5.9 用户对基站权限 (StationPermissions) .....	65
表 5.10 用户日志 (UserLog) .....	65
表 5.11 监控视频 (SurveillanceVideo) .....	65

## 第六章

表 6.1 红外探头组件参数表 .....	71
表 6.2 测试指标 .....	72
表 6.3 功能测试结果 .....	77



## 第一章 绪论

### 1.1 引言

森林火灾是一种突发性强、破坏性大、救助困难的自然灾害<sup>[1]</sup>。从 1950 年至 1987 年的 38 年间，全国年均发生森林火灾 15838 次，火灾发生率为 13.9 次/10 万公顷，年均受害森林面积 94 万公顷，森林火灾受害率平均 8.5%，是世界同期平均水平的 8 倍，全国因森林火灾年均伤亡 800 人。据有关专家测算，这一时期全国共烧毁木材 9.55 亿立方米。按每米价值 150 元计算，全国直接经济损失达 1433 亿元，占全国各类火灾总损失的 56.5%。如何做好森林防火工作，即有效预防和扑救森林火灾，确保人民生命财产安全还是一个大难题，只能做到早发现早扑救，尽量将森林火灾造成的损失降到最小。然而森林火灾形势十分严峻，过去近百年的时间里，全球气温上升了 0.5 到 0.8℃，尤其是在最近的 50 年里上升幅度较大。专家预测未来 10 到 15 年，平均气温上升的幅度更大。随着林区气温升高、林区可燃物增加、火源管理十分困难等情况，使得森林火灾的危险性进一步增加。面对如此严峻的森林防火现状，要求人们运用更多的先进技术和方法，采取最有力的措施，尽可能对森林大火的发生和蔓延进行最大限度控制。

### 1.2 选题背景

森林火灾具有突发性、随机性、短时间内能造成巨大损失的特点<sup>[2]</sup>。一旦有火警发生，拥有准确且畅通的火情信息，明确合理的扑救方案是今后防火、灭火的一个趋势。红外成像监控系统能够在短时间内提供火灾发生时的温度影像信息和分析结果，帮助决策者在短时间做出正确的人员、设备配置方案。

红外成像技术<sup>[3]</sup>是目前唯一的一种可以将温度信息瞬间可视化，并加以验证的检测技术。红外成像仪通过非接触方式量化温度，在专业的红外林火监控软件的帮助下，短时间内便可自动完成分析。现在，红外成像技术已成为早期监测领域最有效的检测手段，它能够在林火发生的早期快速、准确、安全的检测林火。据统计，在红外检测上每投资 1 美元，将得到 4 美元的回报。红外图像反映的是目标与背景自身的红外辐射能量的差异，相较可见光图像，红外图像不易受强光和烟雾干扰，能在夜间工作，适应全天候工作的要求。红外成像技术是利用景物对红外光的辐射差成像，可以不依赖外部光线。将红外技术运用到森林防火工程中的能够充分发挥红外成像技术的优势，达到很好的火灾检测效果。

### 1.3 研究现状

近年，欧盟研制出的一套“森林火灾自动观察系统”<sup>[4]</sup>，它由 4 个可以监视 500m 到 10km 范围的黑白成像仪和一个热敏与烟感应器组成，可以在 30s 内发现烟火，40s 内确定起火区域。如在葡萄牙和法国南部的马赛等地，就安装了三组这种情况的摄像和感应装置，每一组装置可以监视 1 万平方米森林。一旦发现异常，在 1 分钟内电脑就可建立连接并将拍摄的图像通过宽带网或移动电话传送到数据分析中心，由其迅速发出火警通报，这套系统的准确率高达 95%以上，欧盟已决定把这项成果推向工业生产和市场。ECP 公司的森林火灾监控系统，利用计算机视觉和模式识别的理论，采用模式识别算法，可以对 4 千米外的林火在短时间内进行识别并发出警报。Bosque 公司的 BSDS 系统采用红外和普通成像仪的双波段监控，在准确识别森林火灾的同时还可以区别其它现象的干扰，误报率较低。

我国对森林防火的研究是在 1987 年大兴安岭“5.6”大火<sup>[5]</sup>后，国家加强了对森林防火工作的投入力度。针对我国森林资源丰富、环境复杂，传统的瞭望塔、护林塔和飞机检测森林火情的方式已经不再适用，因此我国采用森林防火遥感图像资源分类的遥感技术的研究，将遥感技术、卫星定位系统和地理信息系统用于林火检测工作。这种方式主要依赖 GPS 定位系统<sup>[6]</sup>，但是 GPS 卫星对森林监控区域每天只扫描 4-14.2 次，每次时间不超过 30 分钟，因此卫星不能对林区进行全天实时检测，其提供的信息不能做到实时性。同时，人工目视解译遥感图像，人为的因素比较多，难以完全正确把握土地类型图斑，仍然有很多的图斑是难以判读的，需要到现场核实，工作量十分大。由于森林监控精确度要求十分高，因此要求遥感图像分辨率高、精度高，然而分辨率高、精度高的遥感影像价格比较高。

### 1.4 主要研究内容

本文主要介绍林火监控系统中林火监控软件的设计与实现，针对当前对林火监控设备的需求，并结合项目的可行性，本文设计了一种使用红外成像仪的林火监控系统。从课题设计的过程和结果看，主要涉及到如下内容：（1）深入分析林火识别算法。（2）根据实际情况构建林火识别系统，主要包括前端的监控端、传输网络和控制中心三个部分的设计。详细设计这三个部分，明确三个部分的功能和外部接口。（3）按照用户需求设计林火识别系统的软件部分。主要是对客户端软件的设计，包含软件整体的概要设计和各个模块的详细设计。

## 1.5 论文组织与结构

论文对红外理论、红外成像技术等知识做了介绍，对比了几种林火识别算法，选择了适合复杂森林环境的混合高斯建模法，并在前人基础上对混合高斯模型的建立、更新有了更多补充，结合传统的视频监控网络设计了适合红外成像的智能识别监控网络，重点阐述了林火监控软件的设计与实现，最后搭建测试环境对林火监控软件的各个模块做了测试。文章的具体组织结构如下：

与本文的主要工作相对应，本文的章节安排如下：

第一章：绪论，介绍了课题的研究背景，分析了国内外研究现状，最后对本文采用的森林防火监控系统进行简单介绍。

第二章：森林防火中的红外技术，分析了红外辐射、红外成像的理论，简述了红外成像仪的发展，详细阐述了红外技术在森林防火中的运用。

第三章：林火识别算法的设计，这一章包含如下内容：首先分析了森林中火灾发生的条件；然后分析了林火烟雾的特征；其次分析了现有的林火识别算法，分析了各类算法的优缺点；再次结合实际情况分析了视频格式；最后详细阐述了本文中的林火识别算法并做了验证。

第四章：红外林火监控系统的设计，提出了红外林火监控系统的功能需求，对比了几种监控系统方案；并借鉴这几种方案，提出了本课题中的红外林火监控系统方案；详细阐述了红外林火监控系统的软硬件和整体流程。

第五章：林火监控软件的设计与实现，将软件分为界面、温度分析、林火识别、分析结果和数据库等模块。利用软件开发流程设计了各个模块，用软件建模语言描述了各个模块的结构层次，并详细阐述了各个软件模块的实现方法。

第六章：林火监控软件的测试和分析，搭建了测试环境，测试了林火监控软件的各个模块，并做了测试分析。

第七章：结束语，对本文设计的森林防火监控系统进行总结，并提出了整个系统需要改进和优化的地方。





## 第二章 森林防火中的红外技术

### 2.1 红外辐射理论

自然界中,一切温度高于绝对零度-273℃的物体都能向外辐射能量。1800年,赫胥尔首先发现了红外辐射,经过几代科学家 100 多年的探索、实验与研究,总结出揭示红外辐射本质的辐射定律。

#### (1) 基尔霍夫定律

1985年,基尔霍夫研究指出,物体的出射辐射度<sup>[7]</sup> $M_{\lambda T}$ 和吸收能力 $\alpha_{\lambda T}$ 的比值与物体的性质无关,该比值对所有物体来说是波长和温度的普遍适应函数,都等于同一温度下绝对黑体的出射辐射度 $M_b$ 。其表达式(2-1)为:

$$\frac{M_{\lambda T}}{\alpha_{\lambda T}} = M_b = f(\lambda, T) \quad (2-1)$$

该定律表明,当几个物体处于同一温度时,各物体发射红外线的的能力正比于它吸收的红外线的的能力;如果该物体不能发射某一波长的辐射能,也决不能吸收此波长的辐射能。根据这一定律可以推断出,性能好的反射体或透明体,必然是性能差的辐射体。绝对黑体就是定义为吸收和发射所有波段的辐射体,简单讲就是在任何情况下对一切波长的入射辐射吸收率都等于1的物体,也就是说全吸收。显然,因为自然界中实际存在的任何物体对不同波长的入射辐射都有一定的反射(吸收率不等于1),所以,黑体只是人们抽象出来的一种理想化的物体模型。但黑体热辐射的基本规律是红外研究及应用的基础,它揭示了黑体发射的红外热辐射随温度及波长变化的定量关系。

#### (2) 辐射的光谱分布规律-普朗克辐射定律

普朗克提出了量子假设<sup>[8]</sup>,在量子假设的前提下,得出了黑体辐射的普朗克公式。一个绝对温度为 $T(K)$ 的黑体,单位表面积在波长 $\lambda$ 附近单位波长间隔内向整个半球空间发射的辐射功率(简称为光谱辐射度)与波长、温度满足式(2-2)关系:

$$M_b(\lambda, T) = c_1 \lambda^{-5} [\exp(c_2 / \lambda T) - 1]^{-1} \quad (2-2)$$

其中 $c_1$ 是第一辐射常数 $3.74 \times 10^{-16} (W \cdot m^2)$ , $c_2$ 是第二辐射常数 $1.44 \times 10^{-2} (m \cdot K)$ , $M_b$ 是绝对黑体的辐射出射度 $(W \cdot cm^{-2} \cdot \mu m^{-1})$ , $\lambda$ 是波长 $(\mu m)$ , $T$ 是绝对温度 $(K)$ 。

普朗克辐射定律是所有定量计算红外辐射的基础。从这个定律可以得出黑体辐射的几个特点:

①温度越高,所有波长上的光谱辐射出射度越大;

②辐射出射度是随温度的升高而迅速增大的；

③光谱辐射出射度的峰值波长随温度升高向短波方向移动。

### (3) 辐射功率随温度的变化规律-斯蒂芬-玻耳兹曼定律

斯蒂芬-玻耳兹曼定律描述的是黑体<sup>[9]</sup>单位表面积向整个半球空间发射的所有波长的总辐射功率  $M_{bT}$  (简称为全辐射度)随其温度的变化规律。因此,该定律为普朗克辐射定律对波长积分得到式:

$$M_{bT} = \int_0^\infty M_b(\lambda, T) d\lambda = \sigma T^4 \quad (2-3)$$

上式中,  $M_{bT}$  为温度为  $T$  的黑体的总辐射能力,  $\sigma$  为斯蒂芬-玻耳兹曼常数, 近似值为  $5.673 \times 10^{-12} W/(cm^2 K^4)$ ,  $T$  是黑体的热力学温度( $K$ ),  $\lambda$  为波长( $\mu m$ )。从这一定律可以看出, 物体的温度越高, 其红外辐射能量越多。

上述三个红外辐射定律揭示了红外辐射的本质特性, 为红外成像、测温技术奠定了理论基础。

## 2.2 红外成像的原理

自然界中的一切物体, 只要它的温度高于绝对温度<sup>[10]</sup>( $-273^\circ C$ )就存在分子和原子无规则的运动, 其表面就不断地辐射红外线, 只要收集并探测这些辐射能, 就可以通过重新排列来自探测器的信号形成与景物辐射分布相对应的热图像, 热图像反映物体表面的红外辐射场, 即温度场。红外成像仪就是探测这种物体表面辐射的不为人眼所见的红外线的设备, 它的主要任务是将观察平面内的物体发出的红外辐射转换成可见光图像, 便于观测和识别, 红外成像的模型所下图所示,

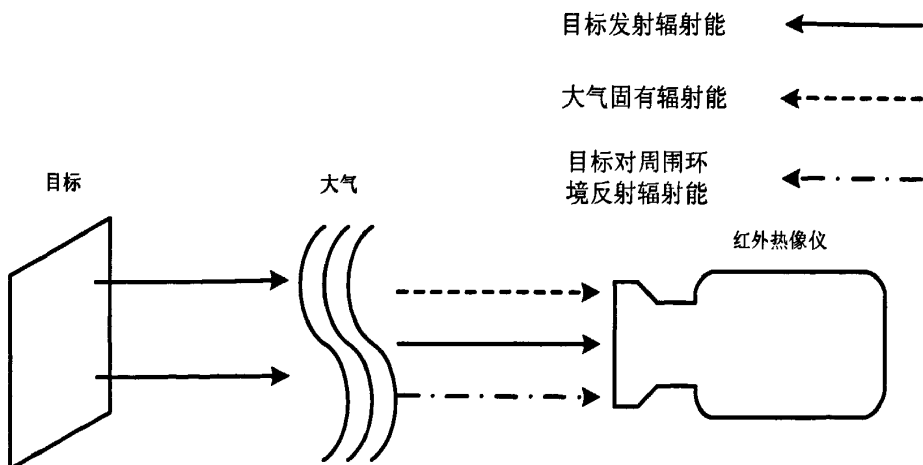


图 2.1 红外成像模型

红外图像与可见光图像的成像机理不同, 人眼所观察到的图像一般表示物体反射可见光以及对颜色的敏感能力。表面光滑的物体由于反射率高, 成像效果就

亮；表面粗糙的物体由于反射率低就显得暗，颜色不同的物体区分度很大。而热成像系统所显示的图像一般反映物体自身辐射的能力，表面光滑的发射率较低，表面粗糙的发射率较高，因此同一温度物体的热像，光滑区域较暗，而粗糙区域显得较亮。红外图像感受和反映的是目标与背景自身的红外辐射能量的差异，而可见光图像感受和反映的是目标及背景反射来自太阳或其它物体光线强弱的差异。这两者都与构成目标及背景的材料、颜色及表面光亮度有关。由于红外图像和可见光图像的成像机理存在着上述本质上的差异，所以它们具有各自的特点。

红外成像技术是利用景物对红外光的辐射差成像，将红外辐射转换为可见光图像，利用场景内物体本身各部分辐射的差异获取物体图像的细节，所以红外图像的获取可不依赖于外部光线。热图像与物体表面的热分布场相对应，实质上是被测目标物体红外辐射的热像分布图，与可见光图像相比，红外热像图缺少层次感，分辨率较低，因此在实际应用中，为更有效地判断被测目标的红外热分布场，常对原始热像图进行相关处理增强其可视性，如滤波去噪，图像亮度、对比度的调整，实时校正，伪彩色处理等技术。

## 2.3 红外成像仪的发展

红外成像仪的发展主要经历了三个阶段<sup>[11]</sup>。1964年，美国TI公司研制成功第一代的热红外成像装置，这种装置利用光学元件运动机械，对目标的热辐射进行图像分解扫描，然后应用光电探测器进行光电转换，最后形成视频图像信号，并在荧屏上显示，这是现代红外成像仪<sup>[12]</sup>的雏形。六十年代中期，在红外前视装置的基础上，开发了具有温度测量功能的热红外成像装置。这种第二代红外成像装置，通常称为热像仪。九十年代出现了致冷型和非致冷型的焦平面红外成像产品，这是第三代的红外成像仪，可以进行大规模的工业化生产。红外探测器的发展是整机发展的关键因素。红外探测器已从第一代的线阵列经历了第二代的二维延迟与积分长波红外的扫描阵列（TDI），发展到了今天的第三代高密度高性能的红外焦平面阵列。

## 2.4 红外技术与森林防火

红外线是一种波长 $0.78-1000\mu m$ 的电磁波。所有物体都会向外发射热辐射，热辐射是红外线的实质体现，表现了物体的温度状况。利用红外设备拍摄的图像体现的是其温度分布状况<sup>[13]</sup>，也就是说物体或物体的一部分热辐射越强，其温度越高；反之，物体或物体的一部分热辐射越低，其温度也就越低。表现在图像上，

温度越高的物体或部分物体，灰度值越大；温度越低的物体或部分物体，灰度值越小。温度稍有变化物体向外发射的热辐射能量就会有很大变化，即物体从正常温度到燃点或者燃烧向外辐射的能量就会有很大变化。因此使用红外线探测物体发射出来的红外能量能够在火灾发生的早期阶段<sup>[14]</sup>准确的定位火情，达到早期检测，减少损失的目的。

通常情况下，森林中的温度在 $-40^{\circ}\text{C}$ - $60^{\circ}\text{C}$ 之间，但是发生火灾后，森林中的火焰温度会达到 $600^{\circ}\text{C}$ - $1400^{\circ}\text{C}$ ，森林与火焰的温度对比反差很大。因此根据图像中灰度值的变化状态可以得到林区温度的变化，通过相应的分析并综合考虑火焰和森林的特征，分离出烟火区域并提示告警。使用红外图像除了能够得到火灾的大致位置外，还能够判断出火灾的范围、性质等情况。

森林中火灾包含 5 个阶段<sup>[15]</sup>：正常阶段、温度上升阶段、初期阶段、发展阶段、熄灭阶段。正常阶段是指此时森林中温度处于正常状态；温度上升阶段是指可燃物处在阴燃的状态，可燃物内部温度持续上升，时刻有可能燃烧；初期阶段是指温度超过可燃物燃点，燃烧开始发生，此时火焰较小并伴随着烟雾；发展阶段是指火焰已经燃烧并呈现火焰面积扩大的态势；熄灭阶段指火灾因为人为扑火、可燃物烧尽或者天气等原因造成火焰熄灭。当前，普遍采用人工巡逻或者可见光成像等方式检测火灾，这些方式是检测处于开始或者发展阶段的火灾。这是一种火灾发生后的补救措施，此时火灾已经产生，只能减低损失。红外线检测主要集中在火灾发生的第二个阶段，即温度上升阶段<sup>[16]</sup>。这时可燃物处于阴燃或者初期燃烧阶段，火灾面积小，火势容易控制。另外由于存在红外窗口，红外线辐射能够穿透浓雾，监控检测目标。红外线有很长的波长因此其探测距离远、探测面积大，有利于森林等大空间区域的监控。

## 2.5 本章小结

本章对阐述了红外技术，分析了红外辐射理论，介绍了红外基本概念和定律。介绍了红外成像原理和红外成像仪的发展。提出了红外技术在森林防火监控系统的运用。分析了采用红外采集设备检测森林的优势，得出了红外监控是火灾早期检测手段的结论。

## 第三章 林火识别算法的设计

### 3.1 森林火灾发生的条件

火焰燃烧的三个条件是：火源、燃烧物、助燃物。燃烧物通常以气相、液相和固相三种形态存在，大气中的氧气一般作为助燃物。森林火灾中，初始火源大多数为固体可燃物如树枝、杂草或人造物质，这些固体可燃物在一定强度的热源或内部自燃的作用下，会先发生阴燃。固体可燃物的燃烧分为三个阶段：温度上升阶段、初期阶段、发展阶段、熄灭阶段。在初期阶段，因为可燃物与氧气的比例不同有不同的燃烧方式：

(1) 可燃物数量不充足时，就不能够持续燃烧，火源燃烧完后自行熄灭。

(2) 可燃物缺少氧气，以阴燃的方式燃烧。阴燃是物质没有火焰的缓慢燃烧，通常表现为产生可燃气体、烟雾和温度升高的现象。可燃物继续缓慢燃烧，直到氧气或可燃物耗尽熄灭。

(3) 如果存在足够的可燃物，而且具有良好的通风条件，火灾就以迅速的方式蔓延到周围。

随着温度的升高和热分解速度的加快<sup>[17]</sup>，可燃气体浓度加大，当温度和浓度达到一定值时，阴燃向有焰燃烧转变，于是出现明火。燃烧也进入到发展阶段，这个阶段燃烧区域发出大量的热量，并会产生大量的烟。大量热量通过可燃物的直接燃烧、热对流或者热量的传导，使燃烧区域向周围扩散，也就是通常说的火蔓延，火蔓延导致了过火面积的扩大。当达到一定的温度时，燃烧区域周边所有可燃物都会开始燃烧，温度升高很快，最高温度会达到 1400℃。在火灾全面发展阶段后期，随着森林中可燃物的不断减少、降雨或人为灭火，火灾燃烧速度递减，燃烧区域的温度逐渐下降。

### 3.2 林火烟雾的特征分析

火焰在燃烧的时候温度通常达到 900℃-1400℃，由于森林中燃烧物种类和燃烧条件十分复杂使得林火燃烧的过程十分复杂。林火在燃烧时在氧化反应的作用下除了产生燃烧剩余物外，还向空气中排放烟雾和大量的高温微粒。高温微粒让火焰以我们看得见的形式存在，火焰发出能被看得见的光线被叫做焰火焰。森林火灾会在早期产生大量的烟雾，烟雾特征是火灾早期检测的主要特征之一。火灾烟雾包含没有燃烧的可燃蒸汽和由于燃烧混入的未燃烧空气，气体燃烧物以及未

完全燃烧的液体、固体和凝结物小颗粒。

火焰烟雾的特征<sup>[18]</sup>包含如下：

#### (1) 火焰形状

火焰中炽热微粒所发出的光线之集合，勾划出火焰形状。由于火灾火焰是一个复杂的几何体，因此火焰形状与观察方位密切相关。

#### (2) 火焰闪烁

火灾火焰具有闪烁的物理特征。这种闪烁不仅在辐射强度上表现为（3~30）赫兹的频率，而且在火焰形状的波动上也有所体现。

#### (3) 火焰的颜色特征

火灾是在一定合适的条件下开始燃烧的，这其中的化学反应需要空气中的氧气，燃烧会产生燃烧物、释放出光和热量。火焰通常是红色的，当然火焰的颜色将随着温度的升高而改变，当温度升高时，火焰的颜色会从红色变成黄色，在达到 1400℃左右的时候会变成白色。这说明低温火焰释放出低饱和度的光，而高温火焰释放出高饱和度的光。同时，在白天或者是存在其他光源的影响，火焰的颜色会比在晚上或者是没有其他光源的存在有更强的饱和度。应该说明的是超过 1400℃或者由于燃烧元素关系的火焰会呈现蓝色火焰。

图像上的非火灾区域可能以具有火焰的颜色，而这些非火灾区域通常会被误判为是火焰。两种因素造成了这些非火灾区域的存在：①物体的颜色与火类似；②太阳的反射、灯光的存在造成的误判。为了有效的判断真正火灾的存在，经常采用火焰及烟雾的动态特征来消除误判。火焰的动态特征包括火焰突然的运动、形状的改变，蔓延的速率和红外光谱的摆动<sup>[19]</sup>。为了改进火灾识别的可靠性，采用火焰的无序特征以及火焰像素的数量状况判断是否是真正的火灾。检测相邻两帧图像的火焰像素数量可以确定目标图像是否有火焰出现。

#### (4) 烟雾的运动特征

通常情况下，火源位置相对固定，因此火灾烟雾的整体运动就会是连续的、平滑的而不是阶跃的，也就是说烟雾的运动拥有相对的稳定性，即使是当火灾处于快速蔓延的情况下，烟雾整体运动也是相对稳定的。

与烟雾的整体运动相反，烟雾局部运动却是不规则的。随着燃烧物燃尽和未燃物被点燃，烟雾区域不断扩散并移动位置。在无风的条件下，烟雾在热量驱动和浮力的作用下，运动方向由下至上。但是在有风力作用的情况下，烟雾除了受到热量的驱动，还会因受到空气气流的影响，而表现出不规则的运动方式。这种运动的不规则特征主要体现在：（1）在空气气流和热浮力的影响下，烟雾整体运动方向是不规则的。（2）烟雾局部运动也是不规则的，这种不规则性体现在烟雾

局部运动方向具有一定的统计分布特征。这种运动不规则特征是森林环境中其他事物所不具备的。

### (5) 烟雾的颜色特征

各种各样的燃烧材料会产生大量的烟。对多数烟来说，他们通常以灰色表现出来，这表示烟的 R、G、B 值相等或者相似。根据这一颜色特征，分析烟雾对背景图像的影响，可以得到区分烟雾和其他运动物体的依据。可以将灰色归纳为两种灰色，即深灰色和浅灰色。

## 3.3 现有林火识别算法分析

目前，基于图像的烟火识别算法主要有：差分分析法，单高斯背景建模检测法以及混合高斯建模检测法。

### (1) 差分分析法

差分分析法就是对拍摄同一区域并且连续时刻的图像的各个像素相减<sup>[20]</sup>得到图像的可疑运动区域。而得到的差分图像  $diff$  是一个二值图像，它是连续图像  $p_1$  和  $p_2$  之间较大灰度差值的区域，用 0 来代表非运动区域，用 1 代表运动区域。即：

$$diff(i, j) = \begin{cases} 0, \text{非运动区域} \\ 1, \text{运动区域} \end{cases}, (0 \leq i < H, 0 \leq j < W) \quad (3-1)$$

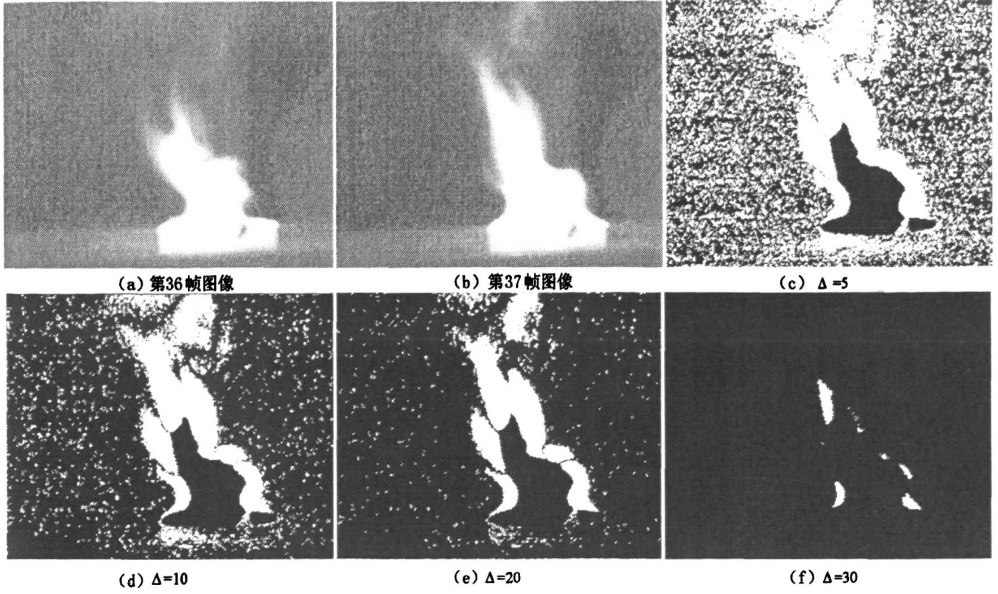
对于采集到的灰度图像，可以用一个  $W \times H$  的矩阵表示， $W$  是图像的宽度， $H$  是图像的高度。差值图像  $diff$  是一个  $W \times H$  的矩阵。分析  $diff$  的矩阵，并生成以  $diff$  矩阵为基础的二维图像。对于差值图像  $diff$  的元素  $diff(i, j)$ ，在利用  $diff(i, j) = p_2(i, j) - p_1(i, j)$  计算差值图像时会出现负值。但是  $diff(i, j)$  的取值范围是 0 或 1，因此取  $diff(i, j) = |p_2(i, j) - p_1(i, j)|$ ，这表示  $diff$  仅表示两幅图像的差值的绝对值，即像素是否存在改变。因此可以认为图像中位于  $(i, j)$  坐标的点满足下式：

$$\sqrt{diff(i, j, R)^2 + diff(i, j, G)^2 + diff(i, j, B)^2} \leq \Delta \quad (3-2)$$

在灰度图像中满足下式：

$$Abs(diff(i, j)) \leq \Delta \quad (3-3)$$

其中  $\Delta$  是一个阈值，因此满足上式的像素就是非运动点  $diff(i, j) = 1$ ，否则就是运动点  $diff(i, j) = 0$ 。实验中，在白天采集一组视频序列，利用差分分析法视频，得到第 36、37 帧的处理结果，效果如下图所示，

图 3.1 不同阈值  $\Delta$  下差分分析法效果

发现检测的效果不是很理想，图像中有很多的干扰区域存在，这些干扰区域有大有小，面积大的干扰是由背景物体热辐射变化产生，面积小的干扰一般是噪声引起的。通过改变阈值  $\Delta$  可以改变处理效果。通过比较后面四幅图可以看出  $\Delta$  取不同的值时，图像运动区域的分析结果是不同的。 $\Delta$  取值很小时，分析结果存在许多噪声； $\Delta$  取的过大时，虽然降低了噪声，但是很多运动点的信息也一并丢失了。

## (2) 单高斯背景建模检测法

单高斯背景建模法<sup>[21]</sup>属于背景建模法。背景建模法的基本思想是对一段图像序列的背景进行建模，在背景模型建立后，将当前的图像与背景模型进行运算，根据运算结果确定待检测目标。单高斯背景模型就是将图像中每一个像素点的颜色值看成是一个随机过程  $x$ <sup>[22]</sup>，同时假设该点的每一像素值出现的概率服从高斯分布。单高斯分布满足下式：

$$N(x, \mu, \sigma) = \frac{1}{\sqrt{(2\pi)|\sigma|}} \exp\left[-\frac{1}{2}(x - \mu)^T \sigma^{-1}(x - \mu)\right] \quad (3-4)$$

其中  $x$  是样本， $\mu$  是期望， $\sigma$  是方差。因此该高斯模型的期望和方差可以表示成：

$$\mu_j = \frac{1}{A} \sum_{t=1}^A \mu_{j,t} \quad (3-5)$$

$$\sigma_j^2 = \frac{1}{A} \sum_{t=1}^A (\mu_{j,t} - \mu_j)^2 \quad (3-6)$$

其中  $A$  表示图像的像素点总数， $j$  表示图像的像素编号， $t$  表示图像编号即图像的帧数。单高斯建模检测法的步骤是：首先，对图像做滤波处理，滤除了干扰噪声之后，对图像序列的所有像素建立单高斯模型，运用  $\mu$  和  $\sigma$  的计算公式，算出每



个像素点的期望和方差。然后，利用下式做前景和背景的判决<sup>[23]</sup>,

$$|pix_{j,t} - \mu_j| < \lambda \quad (3-7)$$

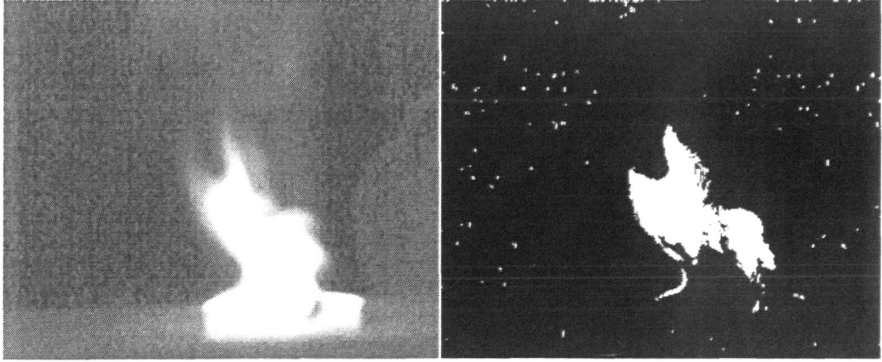
其中  $pix_{j,t}$  是当前点,  $\lambda$  是判决阈值。定义差值图像  $diff(i)$ ,  $diff(i)$  的值由上式的判决来决定, 当判决为真则  $diff(i)$  的值是 0, 表示该点是背景点, 否则该点是前景点。

$$diff_{j,t} = \begin{cases} 0, & |pix_{j,t} - \mu_j| < \lambda \\ 1, & |pix_{j,t} - \mu_j| \geq \lambda \end{cases} \quad (3-8)$$

通常情况下,  $\lambda$  满足三倍标准差原则, 即

$$|pix_{j,t} - \mu_j| < 3\sigma_j \quad (3-9)$$

在判决之后, 考虑到林火烟雾具有运动特征, 即林火烟雾有向周围蔓延的特性。运用前景点面积的相对变化进行判决, 若前景点相对变化持续上升, 使用前景点与总点数的比值作为衡量依据, 当比值持续上升时说明前景点的数量在上升, 证明有火灾存在, 并且提取目标。若比值没有持续上升, 还要对背景模型进行更新。使用单高斯建模检测法的效果如下,



(a) 原始图像 (b) 单高斯背景建模处理效果

图 3.2 单高斯模型运行效果

从上图可以看出在简单的场景下单高斯建模法能够判断火焰的存在, 但是由于林区的环境非常复杂, 这种复杂性体现在森林环境的变化往往带有随机性、经常性, 这种复杂的环境并不能通过一个高斯模型来描述。因此使用单高斯背景建模检测法检测林火, 会经常存在误报的情况。

### (3) 混合高斯背景建模检测法

混合高斯建模检测法<sup>[24]</sup>的产生源于森林环境中, 背景运动导致温度分布的变化, 单高斯模型很难描述这种复杂的背景。某一背景区域像素的正常改变, 会被单高斯模型误判为可疑目标, 因此引入混合高斯模型, 将背景区域的正常运动依然归为背景, 排除其他事物对背景模型的干扰, 比如, 能够排除飞鸟的运动和树枝的摇动等干扰。混合高斯背景模型在单高斯模型的基础上做了延伸, 通过多个高斯概率密度函数的加权平均来平滑地近似任意形状的密度分布函数。混合高斯

模型可由下面二式描述：

$$P(X=x)=\sum_{i=1}^K \omega_i N(x, \mu_i, \sigma_i) \quad (3-10)$$

$$N(x, \mu_i, \Sigma_i) = \frac{1}{(2\pi)^{\frac{n}{2}} |\Sigma_i|^{\frac{1}{2}}} \exp[-\frac{1}{2}(x-\mu_i)^T \Sigma_i^{-1}(x-\mu_i)] \quad (3-11)$$

其中  $x$  是  $n$  维随机变量,  $N(x, \mu_i, \sigma_i)$  代表当前帧某像素的高斯分布,  $K$  表示高斯分布的个数,  $\omega_i$  是每个高斯分布的加权值,  $\mu_i$  表示某一个高斯分布的期望,  $\Sigma_i = \sigma_i^2 I$  ( $I$  是单位协方差矩阵) 是第  $i$  个分部的协方差。为了满足概率密度函数分布的要求,  $\omega_i$  必须满足:  $\sum_{i=1}^K \omega_i = 1$ 。

本文结合混合高斯模型的优势, 同时运用林火烟雾的特征, 实现了一种红外森林火灾识别算法, 该方法在运行过程中准确度高、误报率低。在后面的章节中将详细介绍混合高斯建模检测法。

### 3.4 视频的格式

采用 MPEG-4/H.264 格式<sup>[25]</sup>作为客户端接收到的编码格式。MPEG-4 不仅是针对一定比特率下的视频、音频编码, 更关注与多媒体系统的交互性, 它是一个数据速率很低的多媒体通信标准。MPEG-4 的关键在于“支持基于内容”的编码和解码功能, 也就是对场景中使用分割算法抽取的单独的物理对象进行编码和解码。

MPEG-4 标准继承了以前的标准 (MPEG-1/2 和 H.261/3) 的基本框架, 即典型的三步: 预测编码、变换量化和熵编码。新的压缩编码都是基于优化的思想进行设计, 把先前的标准中某些技术加以改进。例如在原来的基础上提出 1/4 和 1/8 像素精度的运动补偿技术, 大大提高了预测编码的性能。MPEG-4 标准不仅仅给出了具体的压缩算法, 它是针对数字电视、交付多媒体应用、视频监控等整合及压缩技术的需要而制定的。MPEG-4 将多种多媒体应用集成在一个完整的框架里, 为不同的应用提供了相应的类别和档次。

H.264 作为 MPEG-4 标准的第十部分, 采用了预测编码加变换编码的混合编码模式, 它集中了以往各个编码标准的优点, 并吸收了标准制定过程中积累的经验, 获得了比以往其他编码方式好的多的压缩性能。H.264 标准的最大优势在于具有很高的数据压缩比, 在同等图像质量的前提条件下, H.264 编码标准的压缩比是 MPEG-4 的 1.5-2 倍。H.264 采用利于网络传输的结构和语法, 有利于对误码和丢包的处理, 以满足不同的速率、不同解析度以及不同网络传输、存储场合的需求。

视频的解码技术主要运用在客户端上, 可以通过视频编码算法提供的相关接

口，即视频编码压缩参数，作调整以适应不同的环境需求，更好的发挥其性能。这些参数包括编码压缩算法、GOP 尺寸大小、GOP 类型、码流大小、帧率、分辨率等，这些参数的调整主要与网络带宽、网络延迟及抖动情况、存储带宽、存储空间等因素有关。

### 3.5 林火识别算法

本文的林火识别算法，以红外图像作为分析数据，设计了红外图像中林火识别算法的流程，参考了 Grimson<sup>[26][27]</sup>等人建立混合高斯模型的方法和 KaewTraKulPong<sup>[28]</sup>等人对算法运行效率的提升，并借鉴了姚会<sup>[29]</sup>、吴爱国等人的更新方式，改进了混合高斯模型的更新方式，并按照这种更新方式设计了背景模型更新流程，参考了左文明等人对连通区域的目标提取算法提取了烟火区域，从而大大提升了火灾判决的准确度。

#### 3.5.1 林火识别算法流程

森林防火监控系统的林火识别算法利用混合高斯模型背景建模、3 倍标准差原则判断林火烟雾<sup>[30]</sup>，林火识别算法的流程如图 3.3 所示，其具体步骤如下：

（1）接收视频，提取前  $t$  帧图像，对每个像素建立混合高斯背景模型，保存各个背景模型的期望、方差和权重。

（2）对图像做预处理，这里主要是做中值滤波，去除由于红外前端电路或传输过程中引发的干扰噪声。

（3）将该图像的像素与背景模型比较，如果可疑点<sup>[31]</sup>存在，再判断是否是可疑区域，若是可疑区域就提取目标并告警；如果可疑点不存在，或者是可疑点存在，但可疑区域不存在就更新背景模型。如果没有可疑区域直接处理下一帧图像。

（4）当确定图像中存在可疑区域后，通过加标记的方法提取可疑区域，并向操作人员发出告警。

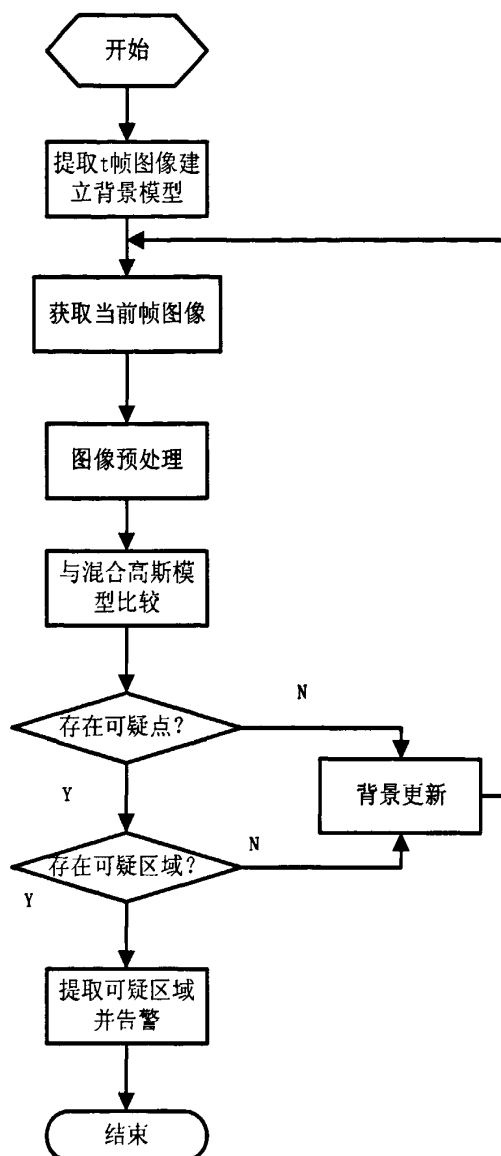


图 3.3 林火识别算法流程

### 3.5.2 图像预处理

图像在传输过程中，传输系统和外部噪声会给图像造成干扰，一般通过滤波的方法消除干扰噪声。滤波的方法有多种，选择中值滤波就可以获得很好的去噪效果。中值滤波<sup>[32]</sup>是将连续图像中的一个点的值用该点周围区域<sup>[33]</sup>（即窗口）中各点值的中值来代替。中值滤波的步骤是：首先，将窗口内的像素按照从小到大排列。然后取像素序列的中间值作为滤波器的输出。在有干扰的情况下，由于干扰值和周围的灰度值存在差异，因此干扰点在经过排序再取中值后会变成与周围像素灰度值一样，就去掉了干扰噪声。中值滤波的数学公式如下所示：

$$y_{i,j} = mid(y_{i+r}, y_{j+s}, r, s \in W, i, j \in I^2) \quad (3-12)$$

其中  $W$  是窗口范围， $I^2$  是图像上的任一点。

本文中的中值滤波算法的窗口不指定大小，提供一个接口由用户指定窗口的大小。当指定的窗口含有奇数个时，取按大小排列的中间值；含有偶数个元素时，取中间两个数的平均值。另外，中值滤波中还要注意的边界问题。所谓边界问题是指滤波窗口不完全包含在图像中。如下图所示，

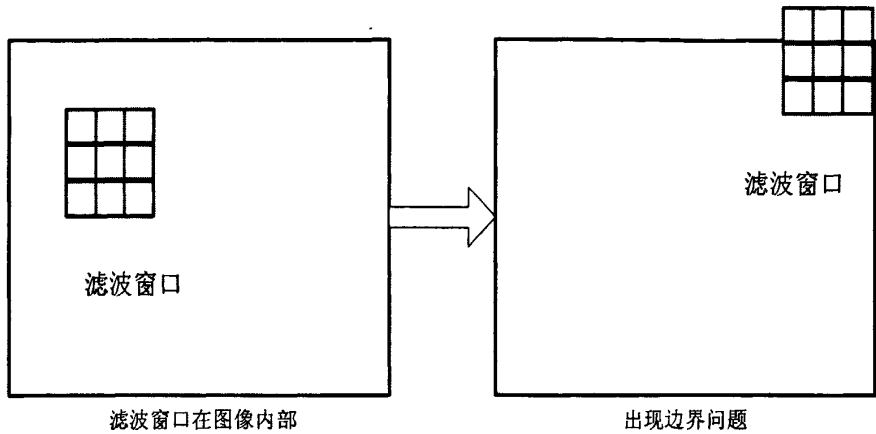


图 3.4 滤波中的边界问题（以  $3 \times 3$  的窗口为例）

解决这个问题可以利用拓展的方式，将尺寸是  $H \times W$  的图像拓展为  $(H+1) \times (W+1)$  的图像，新图像的边界值是原始图像边界值，如下图所示，每个正方形代表一个像素点，相同图案的像素点有相同的灰度值。

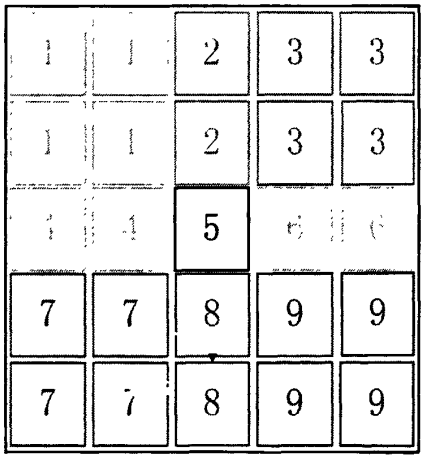


图 3.5 中值滤波的拓展方式

3.5.3 建立模型

(1) 建立背景模型

Grimson 等人建立的混合高斯模型可以克服背景比较复杂且能够适应背景的正常变化，利用多个高斯分布对图像中单个像素点建立模型。每个高斯分布拥有独立的权重值，而描述当前点的灰度值就用各个高斯分布的加权和。

$$P(X=x) = \sum_{i=1}^K \omega_i N(x, \mu_i, \sigma_i) \quad (3-13)$$

$$N(x, \mu_i, \Sigma_i) = \frac{1}{(2\pi)^{\frac{n}{2}} |\Sigma_i|^{\frac{1}{2}}} \exp[-\frac{1}{2}(x - \mu_i)^T \Sigma_i^{-1} (x - \mu_i)] \quad (3-14)$$

上面二式是混合高斯模型的数学描述。式中的  $K$  值通常由计算机的性能来决定， $K$  值越大，高斯分布越多，越能描述当前的分布情况，但处理能力要求也就越高。本文中确定  $K$  值主要遵循如下准则：（1）将背景模型中一个像素点所对应高斯分布个数最高确定为 5 个；（2）对背景模型中一个像素点所对应高斯分布最低个数不作限制；（3）按照过时分布的判别原则，对背景模型中一个像素点所对应的不可信分布进行判别，如果判别为过时分布，将其删除。混合高斯模型有三个参数需要初始化，即期望  $\mu_i$ 、方差  $\sigma_i$ 、权重值  $\omega_i$ ，取  $t$  帧像素点  $\{X_1, X_2, X_3, \dots, X_t\}$ ，用这些像素点初始化期望  $\mu_i$ 、方差  $\sigma_i$ 。在初始状态，只包含一个高斯分布，因此该高斯分布的权重值是 1。

当前像素与混合高斯是否匹配依据三倍标准差原则

$$|pix_{j,t} - \mu_j| < 3\sigma_j \quad (3-15)$$

其中  $pix_{j,t}$  表示第  $t$  帧图像的第  $j$  个像素。按照当前像素是否与混合高斯分布匹配定义了两种分布，即可信分布和不可信分布，可信分布与不可信分布之间的差距在于其权重值  $\omega$  的不同， $\omega$  由本高斯分布的更新频率关联。通过图 3.6 的流程，可以判决可信分布与不可信分布。

当可信分布长期无法匹配，其权重值会逐渐降低，最终变为不可信分布，直至被删除；而一个不可信分布随着与当前像素的匹配，权重值逐渐提升，从而变成一个可信分布。

可信分布与不可信分布的阈值选择可以通过引入 B 分布来解决。含有  $K$  个高斯分布的混合高斯分布按照  $\omega_i / \sigma_i$  的大小排列。当前像素点与各个高斯分布匹配比较，若不匹配  $\omega_i$  变小， $\sigma_i$  变大， $\omega_i / \sigma_i$  变小；否则， $\omega_i$  变大， $\sigma_i$  变小， $\omega_i / \sigma_i$  变大。因此混合高斯分布匹配次数越多， $\omega_i / \sigma_i$  越靠前，就越能够说明该高斯分布越接近背景。前  $B$  个分布被用作背景建模， $B$  的定义如下：

$$B = \arg \min_b (\sum_{i=1}^b \omega_i > T), (1 \leq B \leq K) \quad (3-16)$$

其中  $T$  表示属于背景分布的最小权值比例。当选择一个较小的  $T$  值时，背景模型通常是单模的，这时不能发挥混合高斯模型处理多模态模型的优势。这种情况下，只有使用最有可能的分布维持背景模型；当  $T$  较大时，由树叶摇动等引起的多模态分布将会导致部分图像信息被归入背景模型，因而排除了这些情况对前景的干扰。

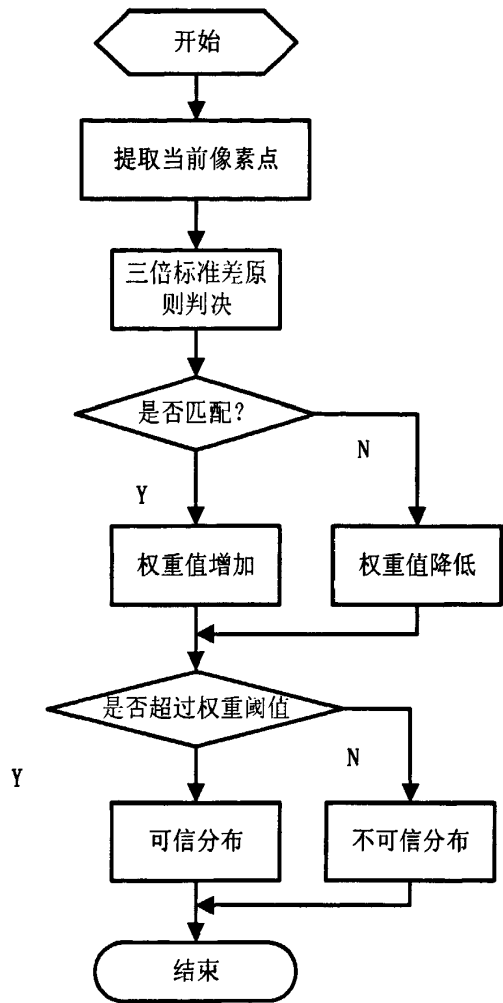


图 3.6 分布类型判别流程

3.5.4 背景模型的更新

野外森林环境十分复杂多变，这就要求混合高斯背景模型必须具有实时更新的功能。Grimson 的文章中的更新方法初始学习速率较慢，因此本文借鉴 KaewTraKulPong 的更新方法并参考了姚会等人的方法使用了一种新的更新方式。混合高斯背景模型的更新涉及到以下几个方面<sup>[34]</sup>：混合高斯模型参数的更新，即期望、方差、权值的更新；不可信高斯分布通过和灰度值的连续匹配，通过积累权重值变为可信高斯分布的更新；当前像素点灰度值匹配的高斯分布的均值和方差更新；新高斯分布的建立以及对过时高斯分布的删除等问题。

(1) 各个高斯分布参数的更新

高斯模型的参数包括  $\mu_i$ ， $\sigma_i$ ， $\omega_i$ ，在读取了像素点之后，要对高斯模型的三个参数做更新<sup>[35]</sup>，下面三式是参数更新的数学模型。

$$\mu_{i+1,t} = \frac{l * \mu_{i,t} + p_{i+1,t}}{l+1} \quad (3-17)$$

$$\sigma_{i+1,t}^2 = \frac{l * [\sigma_{i,t}^2 + (\mu_{i,t} - \mu_{i+1,t})^2] + (p_{i+1,t} - \mu_{i+1,t})^2}{l+1} \quad (3-18)$$

$$\omega_{i+1,t} = (1-\alpha)\omega_{i,t} + \alpha m_{i,t} \quad (3-19)$$

在第  $t$  帧图像中,  $\mu_{i+1,t}$  是更新后像素均值,  $\mu_{i,t}$  是更新前的均值。  $l$  是该高斯分布被更新的次数;  $p_{i+1,t}$  是当前的像素的灰度值。  $\sigma_{i+1,t}$  是更新后的方差,  $\sigma_{i,t}$  是更新前的方差。  $\omega_{i+1,t}$  是更新后的权重值,  $\omega_{i,t}$  是更新前的权重值,  $\alpha$  是学习速率;  $m_{i,t}$  由当前像素灰度值和当前高斯分布是否匹配决定, 如果两者匹配则  $m_{i,t}$  取 1; 两者不匹配  $m_{i,t}$  取 0。每个像素由  $K$  个高斯分布构成, 而每个高斯分布由期望和方差描述。通过对高斯分布的均值、方差权重值的更新, 可以使高斯模型能够适应环境的动态变化。另外定义参数学习速率  $\rho$  满足下式:

$$\rho = 1/(l+1) \approx \alpha/\omega \quad (3-20)$$

可信分布和不可信分布的区分在于其权重值的大小。如果一个不可信分布能够通过连续的和像素值相匹配, 更新其权重值, 达到可信分布和不可信分布的阈值, 则可以成为可信分布; 而一个可信分布如果长时间不能和像素值相匹配, 则其权重值会被更新的越来越小, 有可能被划分为不可信分布, 甚至被当成是过时分布被删除掉。高斯分布权重值的更新分为以下两种情况:

① 新建立高斯分布权重值的确定, 当然一个新建立的高斯分布的权重值是最低的。由于一个像素点的背景模型中所有高斯分布的权重值总和为 1, 这里可以先对当前存在的高斯分布进行权值的更新, 然后求得这些高斯分布的权值和, 用 1 减去这些高斯分布的权值和就是当前新建立的高斯分布的权重值。

② 现有混合高斯背景模型的权重值更新, 权重值的更新方式按照权值更新公式。

使用参数更新公式来更新参数有一个最大的问题就是运算量大<sup>[36]</sup>, 无法实时分析。通过分析 B 分布公式 (3-16) 发现, 混合高斯模型在开始运行的时候, 第一个高斯分布的权重值  $\omega_{i,t} = 1$ , 此时若  $\omega_{i,t} < T$ , 分布是不可信分布。假设图像中一直出现前景点, 那么利用权值更新公式的变型,

$$\omega_{i+1,t} = (1-\alpha)\omega_{i,t} \quad (3-21)$$

那么判定该区域为可疑区域要经过  $frame$  帧, 在不考虑增加新的高斯分布的情况下,  $frame$  满足:

$$frame = \log_{(1-\alpha)}^T \quad (3-22)$$

这时运算量大, 运算时间长, 因此考虑引入  $L$ -Window 进行更新。  $L$ -Window 分为两种方法。公式 I (3-23) 式用在处理帧数小于  $L$  时, 使用统计信息更新模型,



更新方法如下:

$$\begin{cases} \omega_{i,t} = \omega_{i,t-1} + \frac{1}{\sum_{i=1}^K ms_i + 1} (m_i - \omega_{i,t-1}) \\ \mu_{i,t} = \mu_{i,t-1} + \frac{m_i}{ms_i} (pix_{j,t} - \mu_{i,t-1}) \\ \sigma_{i,t} = \sigma_{i,t-1} + \frac{m_i}{ms_i} [(pix_{j,t} - \mu_{i,t-1})(pix_{j,t} - \mu_{i,t-1})^T - \sigma_{i,t-1}] \end{cases} \quad (3-23)$$

其中  $m_i$  满足下式:

$$\begin{cases} m_i = 0, pix_{j,t} \text{与高斯分布不匹配} \\ m_i = 1, pix_{j,t} \text{与高斯分布匹配} \end{cases} \quad (3-24)$$

$ms_i$  是像素点  $pix_{j,t}$  与第  $i$  个分布的匹配次数。当处理的视频帧数大于  $L$ , 采用下面的公式 II (3-25) 更新,

$$\begin{cases} \omega_{i,t} = \omega_{i,t-1} + \alpha(m_i - \omega_{i,t-1}) \\ \mu_{i,t} = \mu_{i,t-1} + \rho_i(pix_{j,t} - \mu_{i,t-1}) \\ \sigma_{i,t} = \sigma_{i,t-1} + \rho_i[(pix_{j,t} - \mu_{i,t-1})(pix_{j,t} - \mu_{i,t-1})^T - \sigma_{i,t-1}] \\ ms_i(t) = \sum_{a=1}^t m_i(a) \\ \rho_i = m_i(\frac{1-\alpha}{ms_i} + \alpha) \end{cases} \quad (3-25)$$

结合以上叙述, 混合高斯模型的参数更新如下所示,

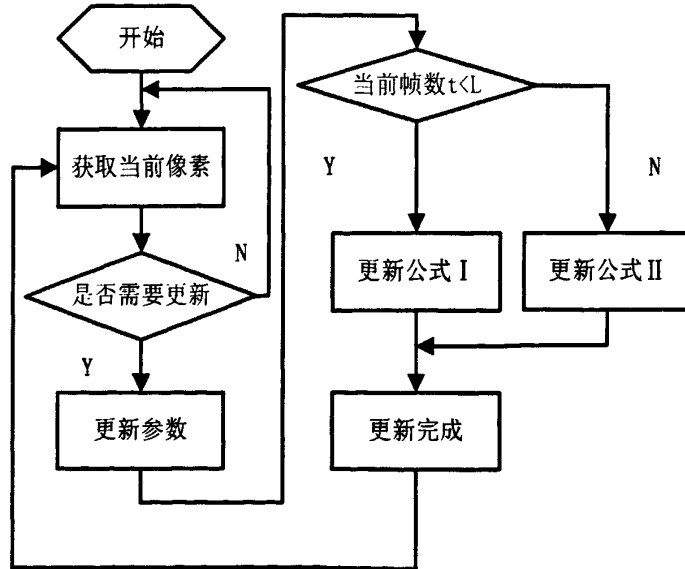


图 3.7 混合高斯模型的参数更新

## (2) 背景模型中过时高斯分布的删除

前面阐述了一个新高斯分布的建立，考虑到算法的复杂度，引入了新的分布，就会有过时分布的删除。根据高斯分布的权重值更新公式可以分析得出以下结论。如果一个高斯分布不能够长时间和场景灰度值相匹配，则其权重值会越来越小，该高斯分布不适应于当前环境的更新，可以判断其是过时分布。对于过时高斯分布的判别需要满足一下两个条件：

- ① 高斯分布是一个不可信的分。
- ② 高斯分布的权重值和方差满足如下公式。

$$\omega_i < \omega_{initial} \quad (3-26)$$

$$\frac{\omega_i}{\sigma_i} < \frac{\omega_{initial}}{\sigma_{initial}} \quad (3-27)$$

$\omega_i$  和  $\sigma_i$  分别代表的是该高斯分布当前的权重值和方差， $\omega_{initial}$  和  $\sigma_{initial}$  分别代表的是该高斯分布在新建立之时的权重值和方差。

如果满足以上两个条件，那么就可以判定当前高斯分布是一个过时分布，可以考虑将其删除，避免该高斯分布影响整个背景模型的学习速度。

### (3) 背景模型中新高斯分布的建立

背景模型建立后，可以对背景灰度值的细微改变做到动态适应。但是前面文章多次讲过，监控系统监控的目标是复杂多变的野外森林环境，随着监控环境的动态改变，如果仅仅存在一个高斯分布，很难对整个背景进行描述<sup>[37]</sup>。所以背景模型的更新过程中需要有新的高斯分布建立，这样背景模型不仅能够适应温度的细微变化，还能够适应干扰目标的温度变化。

新高斯分布的建立涉及到当前像素值是不是干扰目标点的判决，建立过程如图 3.8 所示。被判断为干扰目标点有两个步骤：

- ① 像素点的当前灰度值和像素点的背景高斯分布进行匹配比较，是否满足三倍标准差原则。
- ② 连续几帧的灰度图像中可疑目标点和所有像素点的数量比例做统计，如果在连续几帧图像中，这个比例呈现降低以及不变的情况，也就是说该像素点属于可疑点但不是可疑区域。

如果满足以上两个条件，就可以判定当前像素点为干扰目标点。确定干扰目标点后，为背景模型引入一个新的高斯分布。新建立的高斯分布的初始化如下：新高斯分布的均值以像素的当前灰度值为均值；确定新高斯分布的方差方法是：首先需要寻找当前图像的混合高斯模型中是否有以当前像素灰度值为均值的高斯分布存在，如果存在则以此高斯分布的方差均值为方差，如果不存在则以最相似的高斯分布方差作为新高斯分布的方差；新高斯分布权重值的确定，先要对原有的高斯分布进行权值的更新，根据该像素点灰度值对应的背景模型中所有分布的

权重值和取 1，减去其他的高斯分布的权重值，剩下的就是新高斯分布的权重值了。

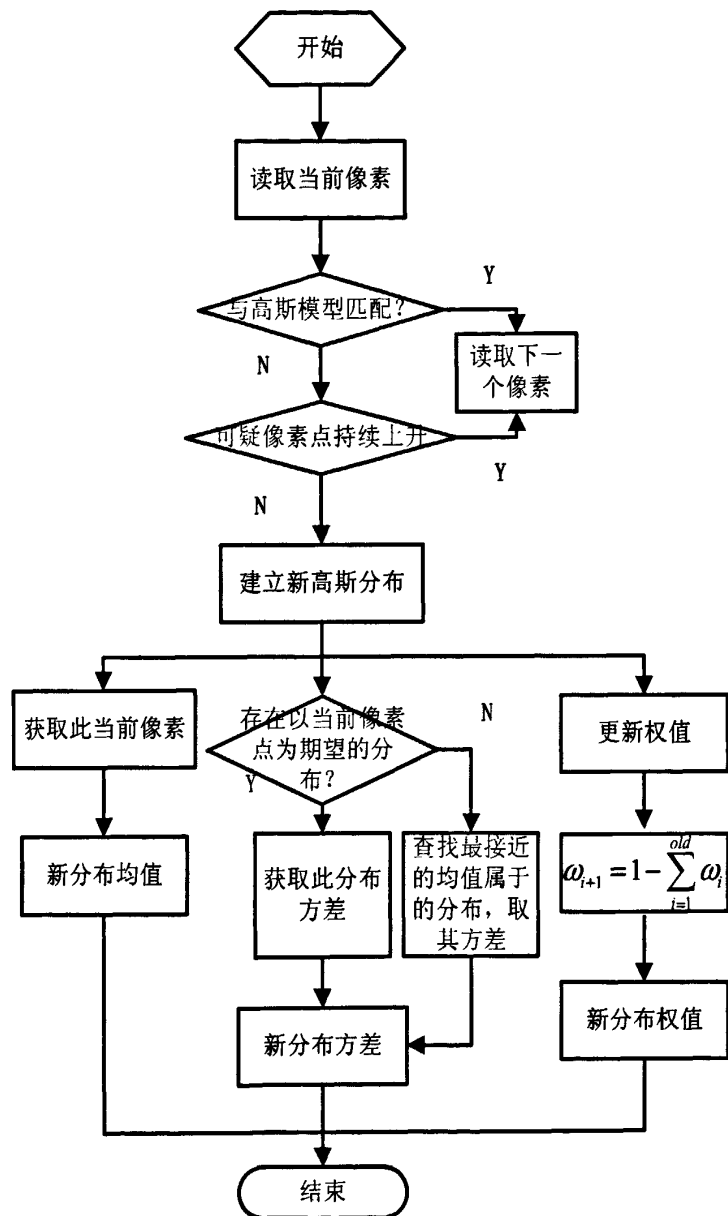


图 3.8 新高斯分布建立流程

需要注意的是，Grimson 的方法对背景的运动比较敏感，在检测一段时间后一般会出现图像中大部分像素都被检测为前景的现象。这主要是因为高斯分布拥有很大的方差，判断是否是可疑点的依据是三倍标准差原则，待测像素点允许与背景有一定程度的偏差。当待测像素点与背景相比，偏差不大时，大部分待测点不会被误判为前景点。然而，当待测像素点与背景持续匹配，根据方差  $\sigma_i$  的更新公式，相应的高斯分布方差  $\sigma_i$  会逐渐收敛即数值上会减小，根据三倍标准差原则当方差  $\sigma_i$  收敛到一个极小值时，图像中满足三倍标准差原则的点会很少，导致背景

点减少而前景点增加。因此，本文通过引入判决阈值  $C$  的方法排除这种情况，判决满足下式：

$$|pix_{j,t} - \mu_j| < C \quad (3-28)$$

其中当前像素在满足三倍标准差原则或上式时，都认为该像素点属于背景点。 $C$  的选择很重要， $C$  要满足条件  $FGThreshold < C < 3\sigma_{initial}$ 。 $FGThreshold$  是  $C$  的下限， $\sigma_{initial}$  是首次建立高斯模型时的方差值。

### 3.5.5 可疑区域的判决和提取

可疑区域的判决主要分成两步：第一步是对可疑点的判决；第二步是对可疑区域的判决。下面详细说明判决方法：

#### (1) 可疑点的判决

可疑点既可以属于运动的干扰点，又可以是烟火像素点。使用三倍标准差原则进行判决，判决方法是将高斯分布按照  $\omega_i / \sigma_i$  的大小从大到小依次排列。然后利用公式 (3-16)，确定  $B$  分布的值，前  $B$  个分布作为可信分布，其他分布作为不可信分布。再将当前像素与前  $B$  个分布的期望依次进行比较，如果有分布满足三倍标准差原则和阈值  $C$  的判定，那么就将该点视为背景点，否则视为可疑点。

#### (2) 可疑区域的判决

林区中环境复杂多变，经常出现干扰点影响林火的判决，因此还有必要考虑林火烟雾的动态特征。林火在燃烧的过程中，会向周围膨胀，反映在灰度图像上就是林火烟雾的面积在持续增加。基于以上可以认为可疑区域的判决主要分为两步：

①存在可疑点。运用上面的判定方法确定是否存在可疑点。

②可疑区域是否持续上升。依据林火的蔓延特征，对连续几帧的可疑目标点和总点数的比值做统计，如果这个比值持续上升，那么图像中存在可疑区域；否则该可疑点只是干扰点。在实现中为了减少运算量对每五帧图像的可疑目标点和总点数的比值做统计。可疑区域的判决流程如图 3.9 所示，

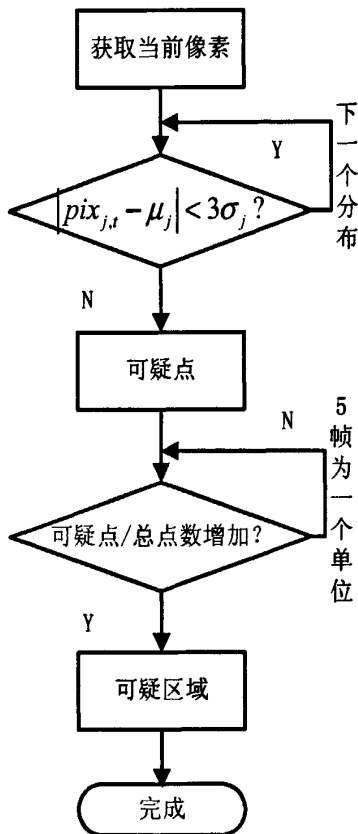


图 3.9 可疑区域的判决

在确定可疑区域之后，下一步的工作是对可疑区域进行提取。对可疑区域的提取可以分为以下三步：

首先，无需将图像中的像素点进行二值化处理，只是将被判决为可疑点的像素点进行标记处理即可。这种处理即给每个可疑点提供一个标记域，标记该像素点为可疑点。对可疑区域进行提取，首先需要扫描图像，将和非可疑点相邻的可疑点提取出来作为可疑区域的边界，而可疑区域边界遵循四连通原则，所以这些边界都是相连的。

然后，取得了可疑区域的边界点后，要先对图像做进一步去噪处理。采用下面的方式对图像做去噪处理，以每一个可疑点为中心的  $3 \times 3$  的区域里面统计可疑点的个数，如果可疑点的个数超过半数，那么就可以保留为可疑点，否则认为其是噪声，同时确定边界可疑点。前面预处理算法中已经对图像进行了中值滤波处理，但是中值滤波的对象是图像中的像素杂点，减少图像中的噪声，为以后进一步的处理做准备，而在可疑区域提取过程中的滤波处理对象是被定义为可疑区域边界点的可疑点，将不利于边界联通区域提取的边界点淘汰，简化边界复杂度。

最后，用四连通原则对可疑区域进行提取，可以用以下步骤来提取可疑区域：

- ①给每一个边界可疑点增加一个标志域，标志域的内容是起点、未检测点和

已检测点，分别用-1、0、1 来表示<sup>[38]</sup>。

②在灰度图中，从上到下，从左到右搜寻一个未检测点，找到以后把像素点的标志域更改为起点，为这个新的跟踪建立一个  $3 \times 3$  的外接矩形，初始化其上、下、左、右 4 个边界，转到③，如果没有找到未检测点则转到⑦。

③从起点开始，按照四连通的原则，在其上下左右 4 个邻接点寻找未检测点，不存在孤立的边界点，找到未检测点后，将其标志域改为已检测点，并将其当作当前操作点，同时需要判断当前点是否对图形的外接矩形有影响（即改变外接矩形的四个边的位置），如果有影响就需要对外接矩形进行改变扩充。

④从当前操作点开始，按照四连通原则，在其上下左右 4 个邻接点中寻找未检测点或起点，如果找到就转第⑤步，否则转⑥步。

⑤如果找到的是未检测点，则将其标志域更改为已检测点，将其作为当前操作点转回第④步；如果找到的是起点，要判断这一点和当前操作点是否互相找到对方，如果找到对方，仍以原来的当前操作点开始，转回④；反之，则将标志域起点更改为已检测点，转到⑥。

⑥将第②步中建立的外接矩形的四个边界值确定，并转向②寻找新的可疑区域。

⑦整个图形的可疑区域寻找完毕。

⑧对上面几步中形成的矩形进行处理。当一个矩形被包含在另一个矩形之中时，要将被包含的矩形进行舍弃，只保留外面的一个矩形；当两个矩形出现重叠的现象时或者是两个的矩形的边界想接触的时候，要用这两个矩形的公共外接矩形来将这两个矩形<sup>[39]</sup>包含起来，当然遵从于第一个原则，就必须将被包含的这两个矩形进行舍弃了。

目标提取的流程如图 3.10 所示：

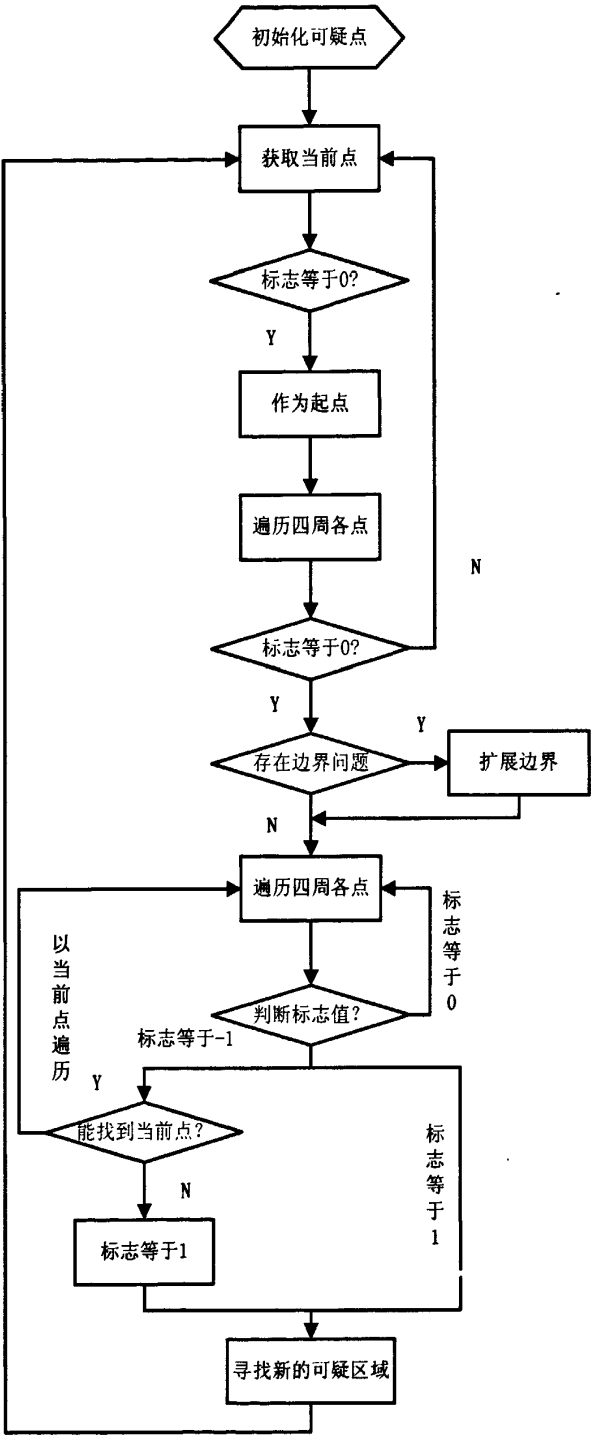


图 3.10 目标提取流程

此外，考虑到边界提取四连通原则的要求，即边界点越少就越有利于提高算法性能，在滤波环节中过滤了一些边缘的可疑点，所以在最后确定可疑区域的外接矩形时，考虑到林火和烟雾的动态性，需要对外界矩形进行扩大化处理，这样能够有效的使可疑区域外接矩形最大限度的包含可疑区域。

### 3.5.6 林火识别算法的验证

以上几节对混合高斯模型检测法作了详细的叙述。下面是在 Visual Studio 2005 中验证混合模型检测法，验证方法如下：

- (1) 对当前图像做预处理，使用中值滤波滤除噪声。
- (2) 建立混合高斯模型，包括初始化混合高斯模型的各个参数，将中值滤波窗口设为  $3 \times 3$ ，建模帧数设为 20frame，高斯分布上限定位 5 个分布，阈值  $T$  定为 0.8。
- (3) 调用混合高斯模型中的更新机制对背景模型更新。
- (4) 判决可疑区域，首先判决可疑点，当可疑点持续上升时判决可疑区域。
- (5) 利用红外成像仪拍摄 8 位深度，分辨率是  $320 \times 240$  的视频，使用林火识别算法进行检测，验证效果如下图所示，

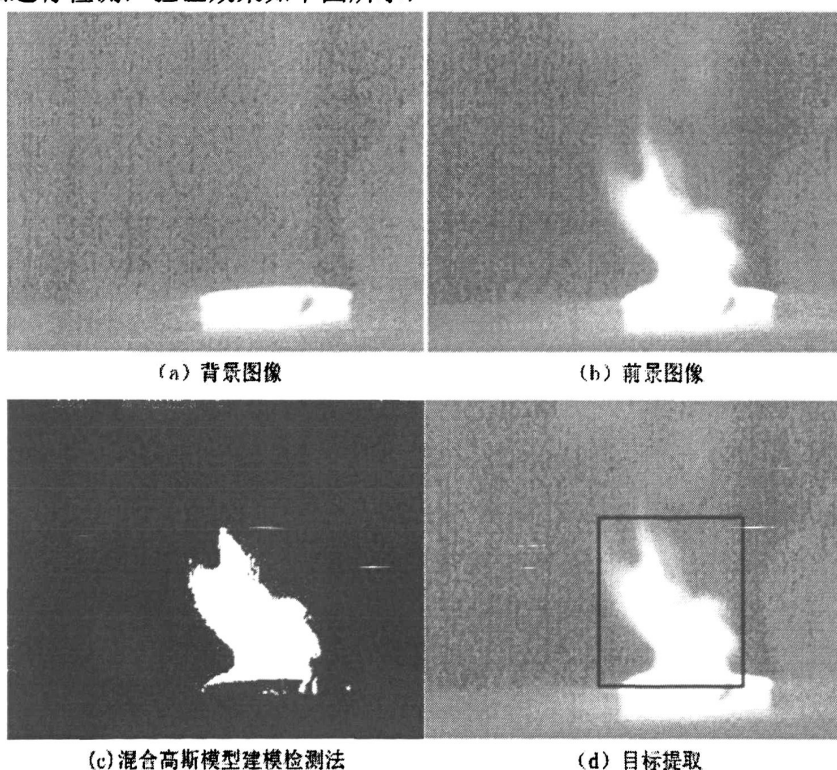


图 3.11 林火识别算法效果图

(6) 识别效果分析。可以发现本算法能够准确识别火焰，并且能够提取火焰。另外注意到，图 3.11 (c) 中干扰噪声的数量也比前两种方法有大幅降低。

## 3.6 本章小结

本章阐述了林火识别算法，利用混合高斯模型对森林背景创建模型。结合高



斯模型的匹配原则和林火烟雾的蔓延特征，判决林火烟雾是否存在。并介绍了可疑目标区域的提取方法。同时对林火识别算法做了验证并分析了结果。



## 第四章 红外林火监控系统的设计

### 4.1 林火监控系统需求分析

通过分析传统的视频监控系统，并结合森林环境和用户对森林防火监控系统的需求，确定红外林火监控系统<sup>[40]</sup>应具备下面几个特点：全天候监控；监控范围广；无线远程传输；智能早期识别林火烟雾并告警；数据方便管理；软件界面友好。

#### （1）全天候监控

拍摄设备要求白天、夜间 24 小时工作。要能够排除浓雾等天气造成的干扰，能够在复杂多变的森林环境中不间断的监控图像。

#### （2）监控范围广

监控摄像机的拍摄距离尽量远。摄像机安装在森林中的制高点，并配有云台负责控制摄像机的运动。云台能够方便上下左右转动，让摄像机的拍摄范围更广。

#### （3）无线远程传输

因为监控地点与监控中心距离较远，所以使用无线远程传输。要求无线网络传输距离足够远，延迟低并且传输速率快。

#### （4）智能识别林火烟雾

能够在无人留守的情况下，运用视频分析算法自动判断是否存在林火烟雾。算法能够有很高的准确度，误报率低，并且能够按照算法功能需求的定义随时更新。

#### （5）管理数据方便

能够长期备份日常的监控数据，且数据不会因某种情况出现丢失。

#### （6）软件界面友好

软件交互性强，容易操作，运行稳定，更新方便，并能够注重操作感受，让操作效率更高。

### 4.2 林火监控系统的方案分析

当前的可见光型林火监控系统是图像型林火监控系统的主流<sup>[41]</sup>，但是可见光型林火监控系统有如下诸多缺点：

#### （1）检测滞后

森林中树木茂密，而火焰能够在阴燃的情况下产生，可见光摄像机利用的是

可见光反射到镜头再成像的原理，无法捕捉到被烟雾或树叶遮盖的烟火。当被可见光摄像机发现，火灾可能已经形成蔓延的态势，已经造成了无法挽回的损失。

### （2）抗干扰性差

森林中水汽多，容易形成雾天，而可见光摄像机在浓雾天气的可视距离比平时大大降低，因此长期使用可见光摄像机监控森林的效果不好。

### （3）夜间效果差

夜间可见光摄像机捕捉不到影像，虽然有的可见光摄像机拥有夜视功能但是效果不好、拍摄距离也很有限。同时夜视功能也没有办法解决前两个问题。

相较可见光摄像机，红外成像仪有两大优点：（1）红外图像反映了物体的温度分布，温差越大，拍摄效果越明显。森林中的可燃物在燃烧之前温度会迅速上升，出现阴燃现象。红外成像仪拍摄的图像反映了物体的温度分布情况。利用这一点，红外成像仪能够及时扑捉到阴燃时的火灾隐患，属于早期的火灾检测。同时，背景和前景的温差越大，红外线的成像效果越好。因此红外成像仪可以敏锐的拍摄到被遮盖物体的温度分布。使用红外成像仪监控森林区域，就能够拍摄到被树叶、浓雾遮盖的物体。（2）可见光摄像机在夜间不能扑捉到图像，有的可见光摄像机拥有夜视功能，但这类采用 LED 灯的摄像机拍摄距离十分有限，而红外成像仪在夜间能够有很好的监控效果。夜间烟火温度和周围环境温度相比差别更大，拍摄的烟火效果更加明显、成像效果更好，更有利于对烟火的监控。

红外林火监控系统方案原型来自于视频监控系统，因此林火监控系统的方案带有视频监控系统的色彩。当前的视频监控系统分为两类：模拟监控系统和数字监控系统。模拟监控系统中的各个设备都是模拟器件，提供了简单的软件，只允许设备设置和视频浏览的功能。数字监控系统的前端采集设备输出的是数字信号，以网络作为传输手段。数字监控系统将监控视频和多画面显示的功能进行整合，正在逐渐取代传统的模拟监控系统。

红外林火监控系统是以数字视频监控系统为基础，因此与数字视频监控系统既有联系又存在区别。与数字视频监控系统一样，林火监控系统增加了视频采集路数，提升了多画面图像的显示速率，同时增加了对云台和镜头的控制。图像在压缩后通过传输网络，再由终端软件解码，显示图像。

林火监控系统又存在自己独特的方案，林火识别算法是林火监控系统区别于一般监控系统的关键。林火监控系统方案的选择依据是综合考虑林火识别的准确度、林区的复杂情况以及开发成本。结合林火识别算法本文提出了三种林火监控方案：前端分析、嵌入分析、后端分析。下面详细比较这几种方案：

#### （1）前端分析方案

前端分析方案是对传统的模拟监控系统的改进，是在监控摄像机附近加装林火识别单元。使用视频分配器将摄像机输出的信号分成 2 路，一路进入后端视频压缩模块；另一路进入林火识别单元进行分析处理，当林火识别单元检测到有林火烟雾存在就通过网络接口向监控中心报警。

(2) 嵌入分析方案

嵌入分析方案是布置在摄像机或者是视频服务器上，实现林火识别。林火识别模块作为一个单独的模块集成在摄像机或视频服务器上，摄像机拍摄的数据直接传给林火识别模块分析。

(3) 后端分析方案

后端分析方案也是一种增加林火识别模块的方案，但是林火识别模块布置在后端的客户端上。摄像机将视频信号拍摄之后，经过压缩处理传输给客户端。客户端调用林火识别算法对视频进行分析，之后决定是否报警。因为林火识别模块要占用大量的处理资源，所以采用这种方式能显著减轻监控前端的处理压力。

下表是三种方案优缺点对比：

表 4.1 林火监控系统方案比较

方案	描述	优点	缺点
前端分析	识别模块独立出来，布置于监控前端	直接对未压缩视频做分析。独立于系统，兼容性好。	(1) 为了支持传统的模拟监控系统的一种过渡方案，识别模块不具备压缩功能成本高。(2) 不能和整个系统兼容，报警信息的输出要考虑发送端是否支持。
嵌入分析	识别模块嵌入在前端设备中，布置于监控前端	有火灾时才将告警信息传给 NVR，节省网络资源。	(1) 开发周期长，前端系统高度集成，容易出现故障。(2) 视频分析软件升级繁琐，不利于维护。(3) 前端设备芯片处理能力有限，限制了识别算法。
后端分析	识别模块布置于客户端	识别模块位于后端，便于集中部署。识别模块由软件运行，模块升级方便。降低前端对系统的要求，成本低	(1) 对 NVR 的要求高。 (2) 客户端处理能力有限。

本文中的图像采集设备使用的是红外成像仪，因此前端分析方案并不适用。嵌入分析方案虽然减轻了传输的压力，但是由于集成度高，研发周期长且易出故障不利于后期维护；同时前端芯片处理能力和内存有限，可能导致判决延迟或者数据溢出，也不适合林火监控。后端分析方案将林火识别模块布置于客户端上，模块升级容易且不需要前端进行处理，降低了成本。后端分析方案的缺点是路数增多会增加传输设备的处理负荷，但是通过可以增加传输设备的数量和配置即可解决这个问题；另外客户端的处理能力也可以通过提高配置来解决。因此本文采

用后端分析方案。

### 4.3 林火监控系统总体方案设计

本节将根据上节的分析提出红外林火监控系统的方案设计。森林防火监控系统的监控前端包括若干个监测站<sup>[42]</sup>和与之配套的 DVS。DVS(Digital Video Server)即视频服务器,是一种压缩、处理视频音频的网络传输设备。每个检测站可设置在瞭望台或制高点上,向四周进行 360 度的搜索。某个检测站发现火情目标后,将视频数据通过无线发射机向外发送,在传输过程中会有衰减,经过无线网络放大后再将信息发送到监控中心,监控中心的接收机收到监测站发来的数据后,转发数据,由客户端显示器显示林区的状况,系统会自动向操作员报警。林火监控系统的结构模型如下图所示,

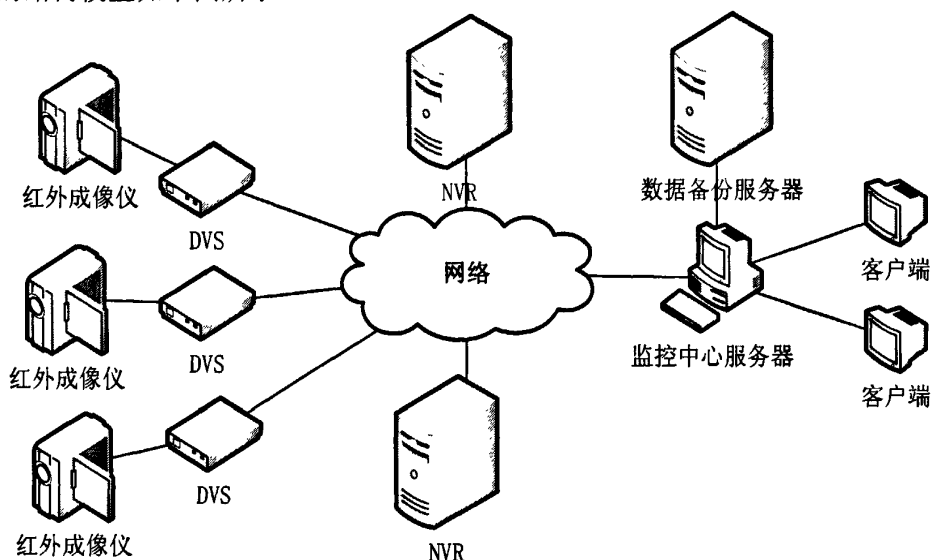


图 4.1 森林防火监控系统结构模型

按照地理位置的不同,将林火监控系统分成监控前端、监控中心以及传输网络三个部分<sup>[43]</sup>。监控前端由摄像机、云台、DVS 等构成。监控中心包含中心服务器、NVR(Network Video Recorder,即网络视频录像机)、数据备份服务器以及客户端等。从设计和实现的角度看,林火监控系统又包含硬件和软件,下面分别阐述林火监控系统的硬件和软件方案。

4.4 林火监控系统硬件设计

4.4.1 监控前端硬件设计

监控前端能够顺利的采集目标物体辐射的红外线，捕捉物体温度变化产生的红外辐射变化。监控前端分为以下几个部分：

(1) 云台的设计

通常在森林中的制高点建一个塔楼，再在上面安装红外成像仪，利用云台控制成像仪的转动<sup>[44]</sup>。云台能够带动成像仪水平和垂直转动，可以利用云台完成成像仪的开启和关闭以及指定速度的水平、垂直运动。在林火监控系统中云台要满足三个条件：

① 良好的防震性

云台要有很好的抗震性，通常云台被安置在山顶处，风力的作用会导致成像仪的晃动，引发控制中心接收到抖动严重的画面，这会降低林火识别的准确度

② 云台远程可控

在监控端能够实现对云台的远程控制。利用云台的解码器和解码协议能够控制云台的水平、垂直运动方向和速度。

③ 提供现场数据

云台要能够利用解码协议向控制中心传回方位角和俯仰角等信息，利用传回的数据定位火灾发生的位置。

综合上述需求设计了一种云台解码协议，编码格式如下：

表 4.2 云台协议

同步字段	地址	控制	水平数据段	垂直数据段	校验字段
1 字节	1 字节	1 字节	7 字节	7 字节	1 字节

所有的字段以 16 进制来表示，同步字节的值是 FF。地址字段是接收或是发送端的逻辑地址。校验字段是从同步字段到数据字段的字节数和。控制字段包含 8bit，具体格式如下表所示：

表 4.3 云台控制字段

7 位	6 位	5 位	4 位	3 位	2 位	1 位	0 位
特殊位	开关	手动/自动扫描	向下	向上	向左	向右	预留
1 是自动模式	1 是开，0 是关	1 是自动，0 是手动	1 有效	1 有效	1 有效	1 有效	通常置为 0

第 7 位是特殊位，如果是 1 表示第 6 位和第 5 位都是 1，这样就不用查询这两位，如果是 0 还需要继续查询。第 6 位控制成像仪的开关，第 6 位确定是手动/自动的扫描。从第 5 到第 2 位控制摄像头的转动方向。水平角度和垂直角度各用 1

字节的浮点数表示，其中水平角度的范围是从 0.000° 至 360.000°，垂直角度从 -90.000° 至 90.000°。例如：FF 01 4A 30 31 30 2E 32 30 32 30 32 35 2E 31 30 30 11 表示选择手动模式，垂直方向向上转动 25.1° 水平方向向右转动 10.202°。

(2) 红外成像仪的设计

使用自主研发的红外成像仪作为采集设备，该红外成像仪以 TMS320 DM6437 DSP 处理器为核心，红外成像仪的硬件结构如下图所示。系统主要由视频前端、核心 DSP 处理器、存储器、外围接口、液晶显示屏、电源等部分构成。

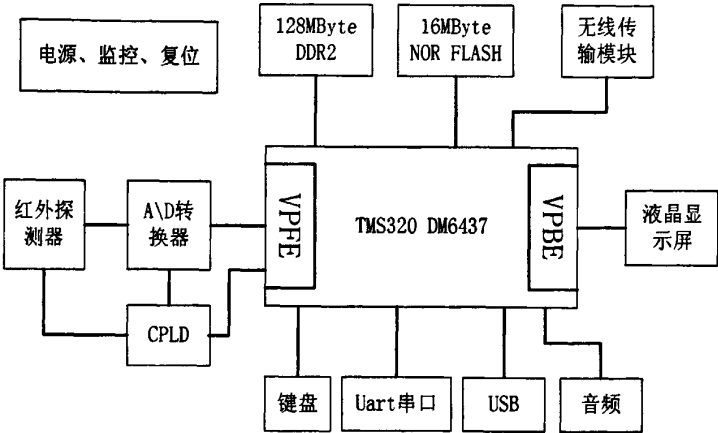


图 4.2 系统硬件结构图

系统各个模块的功能如下：

①TMS320 DM6437 核心处理器

在该方案中，整个系统的数据处理任务全部由 DSP 处理器来完成，主要包括图像数据的采集、处理、显示，以及红外测温处理，为了满足系统对实时性和运行高复杂度算法的要求，对 DSP 的性能有较高的要求。

②存储模块

DSP 子系统的存储模块包括 128Mbyte DDR2 SDRAM、16Mbyte NOR FLASH 和 SD 卡。其中，SDRAM 作为片内存储器的扩展，主要用于：

- 1) DSP 的可执行代码的运行空间；
- 2) 缓冲传感器或视频源的输入图像数据；
- 3) 用做 VPFE 处理或缩放图像数据的中间 buffer；
- 4) 用做 OSD 播放 buffer；
- 5) 用做图像处理过程中大量原始数据的中间 buffer；
- 6) 视频编解码过程中的中间数据 buffer。

Flash 则用于固化参数和代码，并引导系统的启动；SD 卡则用于存储采集到的有用图像及相关数据。

③液晶显示屏



图像液晶显示器将采用彩色高分辨率液晶屏,分辨率为 640×480。DM6437 的视频处理后端中视频编码器将数字图像进行编码,生成 PAL 或 NTSC 制式的复合视频信号,由液晶屏显示。

#### ④视频前端

视频前端包括红外探测器、A/D 转换器和 CPLD,主要完成红外热图像的采集和数字化。红外探测器收集红外辐射并将其转换为模拟电压值进行输出,完成图像的采集。A/D 转换器将该电压值变换为数字信号,完成图像的数字化,然后依次输入 DSP 的视频处理前端。CPLD 则主要负责红外探测器的驱动、A/D 转换器、DSP 的视频处理前端之间的同步。

#### ⑤外围接口

外围接口包括键盘、UART 串口、USB 及音频接口等,这些接口使成像仪的操作根据方便灵活。

### (3) 防护罩的设计

防护罩也是森林防火监控系统中的必要设备,防护罩的主要功能是防尘和防破坏。在野外使用的防护罩密封性能一定要好,保证雨水不能进入防护罩内部侵蚀摄像机。并且这一类的防护罩要求能抵挡各种气候,比如风、雨、雪等。同时也可以配备排风扇、加热器、雨刮器,可以更好的保护摄像机设备。当天气较热时,排风扇自动工作;气温太冷时加热器自动工作;当防护罩上有雨水污物时,可以通过控制系统启动雨刮器。

### (4) 视频服务器的设计

视频服务器(DVS),主要是对模拟视频信号进行编码压缩<sup>[45]</sup>,并提供网络传输功能。DVS 适合在监控点比较分散的环境中,DVS 有下面的关键指标:图像质量,延时性,网络适应性,QoS,开放性。通常情况下,DVS 拥有 1-8 个视频输入接口,用来连接摄像机信号输入,一个或两个网络接口,用来连接网络。DVS 内置 Web 服务器、压缩芯片以及操作系统,可实现视频的数字化、编码压缩及网络存储。另外还有报警输入输出接口、串行接口、音频接口等。

DVS 拥有断点续传(ANR)、冗余不间断工作以及以太网供电等功能。在客户端访问前端监控视频时,NVR(被用作媒体服务器)在给多个客户端发送视频时资源占用高并且多路视频分发也会造成带宽浪费。因此 DVS 提供了组播的功能。这些功能提高了 DVS 以及整个网络传输的可靠性。相比较普通的数字视频录像机(DVR,即 Digital Video Recorder),DVS 拥有如下的优势:

#### ① 功能角色

DVS 实质上可以看作是 DVR 的一个功能分拆,即将编码压缩器件从 DVR 中

完全独立出来，分布部署在前端，然后在后端与其他设备配合实现视频的存储转发，再加上磁盘存储设备，构成了完整的网络监控系统，可以说，在 DVS 系统中，分工更加精细，这样的好处是设备更专业、稳定性更高。

### ② 开放性

DVR 系统集成了视频编码压缩、本地存储及网络传输等所有功能，这种一体式的设备给用户一个相对完整的解决方案，正是这种结构，导致了 DVR 不开放性，因为它可以不需要与其他平台、存储系统等协同工作。而 DVS 本身仅是一个视频采集编码设备，必须采用开放式架构和标准的协议，产品要求“透明化”，因此 DVS 在开放性、集成性方面比 DVR 更好。

### ③ 稳定性

DVR 的操作系统、电源、硬盘、主板等环节均是故障点，任何一个故障点失效将需要整机停机，影响使用。而 DVS 通常采用嵌入式处理器及操作系统，无硬盘及其他计算机组件，因此，单体设备上，DVS 的稳定性能要强于 DVR。同时，DVS 的后端配合设备，如存储设备和 NVR 服务器等，这类设备技术成熟因而工作稳定。

### ④ 成本因素

对于信号传输部分，DVR 系统要求所有摄像机的视频输入、控制信号等集中连接到 DVR 设备，此方式对于监控点位比较密集的应用比较有利，而如果监控点位分布比较分散，那么所有视频、控制线需要全部连接到 DVR，在管线铺设及施工成本上偏高。在此场合下，存储多数是内部硬盘存储，或直连阵列存储，非集中的存储架构导致存储设备管理、维护成本偏高，而使用 DVS 集中存储在 NVR 上，在 NVR 设备可以集中部署磁盘阵列设备，采用多种存储架构，从而可以获得较好的性价比。

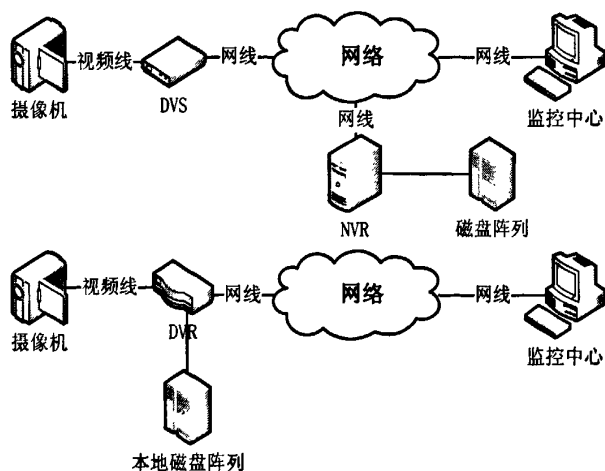


图 4.3 DVS 和 DVR 的方案对比

## 4.4.2 监控中心硬件设计

监控中心负责接收并分析处理通过网络传输的视频分析数据，其硬件主要由服务器、不间断电源供应系统、空调系统构成。

### (1) 服务器

服务器包括中心服务器、客户端、媒体服务器、数据备份服务器。服务器的主要功能是安装各种软件，包括安装监控软件、管理软件和连续电源供应系统的软件。这些主机对硬件配置要求不同。其中媒体服务器需要有较强的处理和数据转发能力，而数据备份服务器要求有较大的存储能力来保存视频数据和日志数据。

### (2) 空调系统

空调系统主要用来保证监控中心能够在一个正常的工作温度下持续工作。随着监控场合对空调系统的温度控制精度不断提高，越来越多的智能空调被投入使用。这类空调系统能够远程控制调节空调中压缩机、风机等模块的状态。在空调出现故障时，能够迅速报警通知操作人员，这就大大提高了空调系统和整个监控中心的稳定程度。

### (3) 连续电源供应系统

连续电源供应系统<sup>[46]</sup>，即 UPS，其应用十分广泛。最初，UPS 系统只是正常供电出现故障时，向需要电源的设备进行供电。但是随着需求的增加，UPS 不但能够提供应急电源，而且还能通过智能化，减少人为失误所造成故障。UPS 的智能化是通过网络接口将 UPS 和网络相连，同时在监控主机上安装相应的监控软件，能够对监控中心的电源状况进行实时的监控；而在电源出现异常时，不仅 UPS 能够发出警报，而且能够通过电脑主机上的监控软件发出警报，提醒监控人员处理故障。有了智能 UPS 系统，监控中心端的电源供应得到了有力的保障，同时也保障了监控中心端的正常工作。

## 4.4.3 传输网络硬件设计

前端采集的视频数据通过传输网络，传输给监控中心，以便后续的播放、分析、存储等处理。而云台控制信息也同样依赖网络传输到前端模块。利用传输网络传输监控画面的流程是：当监控前端收到编码压缩的码流后，按照 RTP 数据传输协议的报文格式装入 RTP 报文的数据负载段，并配置 RTP 报文头部的时间戳、同步信息等参数，之后再封装上 UDP 报头和 IP 报头，然后 IP 数据包通过网络向接收端；接收端收到 IP 包后先分析 RTP 包头判断版本、长度等信息，更新缓冲区的 RTP 信息<sup>[47]</sup>（如收到的字节数、视频帧数、包数等信息），再按照 RTP 时间戳

和包序列号等进行信源同步，整理 RTP 包顺序，重构视频帧，最后根据负载类型标识进行解码，将数据放入缓存供解码器解码输出。期间接收端周期性回送包含 QoS 反馈控制信息的 RTCP 包到数据发送端实现质量控制。整个视频传输模型如下图所示，

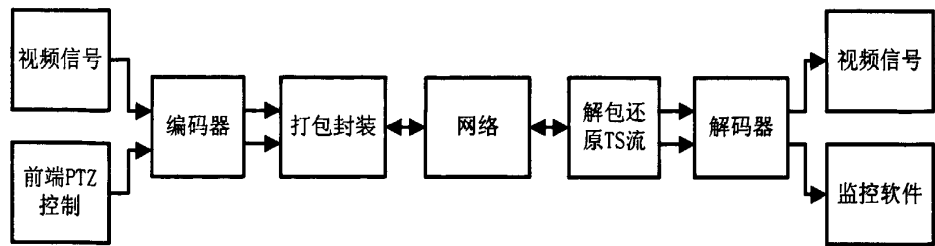


图 4.4 视频传输模型

有线传输方式拥有稳定可靠、传输速率高等特点，但是森林防火系统的监控前端通常都分布范围广、远离监控中心且地形复杂，这些因素大大提高了有线传输的施工成本。除了降低成本外无线传输还拥有如下的诸多优点：

（1）容易施工，通常林区的环境非常复杂，使用有线传输方式在布线的时候会遇到水泽或者悬崖等障碍，使布线十分困难，而利用无线方式只需考虑无线收发设备的性能。

（2）容易维护。通常情况下需要对有线传输线路做定期检查，并且一旦出现故障很难找到故障点。但是用无线传输方式就能够只检查无线传输收发设备，能够快速找出问题，排除故障。

（3）容易升级。在传输网络组建好之后有时要对网络升级，添加一些新的设备。使用有线网络就要重新施工铺设线缆，相反，使用无线网络传输就能够方便的添加收发设备。

在考虑到成本的前提下，无线传输方式通常又有两种方案：GSM 短消息和 GPRS 方式。虽然 GSM 短消息通信方式利用移动公网的资源，初期投资少并且后期的维护成本低，但是 GSM 短消息一次最多只能传送 160 个字符，且延迟很大，无法实时传输监控画面。而 GPRS 系统同样也是利用移动公网的资源，但是基于 IP 的网络，传输速率较高，理论上可以达到 171.2Kb/s，且传输时延小、系统性能稳定。同时 GPRS 是按照流量收费，因此通信资费低廉。现在中国移动的 GPRS 网络已经基本覆盖了全国，采用 GPRS 实现监控推广方便。GPRS 通信是按流量计费，因此通信资费低廉。

4.4.4 电源的设计

监控前端要尽量选择靠近固定电源的地方，电源线按照 220V 布线，经过适配

器转换成直流 12V 或者是交流 24V，电源线与视频数据线要保持一段距离。也可以使用一个大功率的直流稳压电源来供电，这时要选用线径较粗的线，否则监控前端不会正常工作。假如确实找不到固定电源的地方只能使用太阳能供电。

4.5 林火监控系统软件设计

4.5.1 林火监控系统软件的功能需求

林火监控系统的软件是运行在客户端的一个监控软件。通过它能够看到布置在野外的红外成像仪发送过来的红外图像，这些图像是灰度图像<sup>[48]</sup>。通过对灰度数据的分析，即采用建立混合高斯模型的方法判断当前帧是否存在林火烟雾。当发生火灾时，操作人员接到告警提示信息对火灾或者火灾隐患作相应的处理。运用数据库模块能够保存监控和火灾信息，为以后火灾的预防和统计做准备。

UML（Unified Modeling Language）即统一建模语言，是第三代的建模和规约语言，能够提供一种面向对象的描述方法。他包含三类常用的模型：

- (1) 功能模型，功能模型主要从使用者的角度分析软件的功能。
- (2) 对象模型，描述了各个主要对象的属性、操作，并制定了对象之间的关联。
- (3) 动态模型，以时间来衡量一个事物的行为。

通过对林火监控软件的需求进行分析，并运用 UML 中的用例图图 4.5 建立林火监控软件的需求用例。这种方法的好处是站在使用者的角度分析软件的功能，让软件的功能得到最大限度的利用。

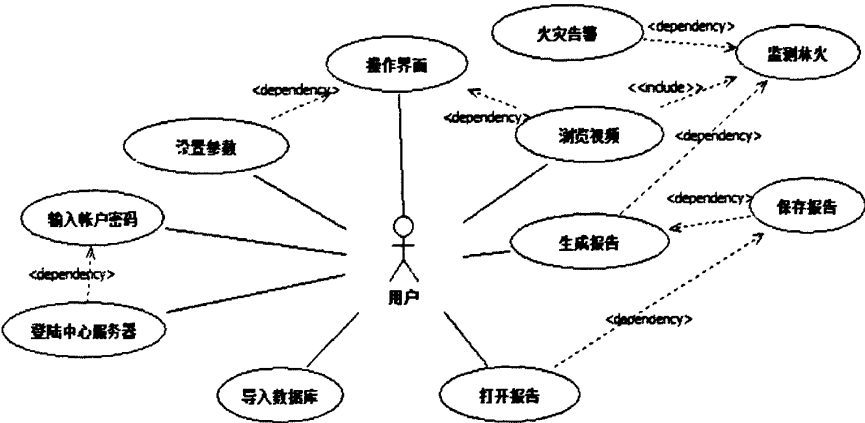


图 4.5 软件需求用例图

依照软件模块的设计原则可理解性、可组合性、可分解性、可保护性和可连续性，并综合分析林火软件需求，确定林火监控软件包含界面模块、鉴权模块、

林火识别模块、分析结果模块和数据库模块。红外森林防火监控软件模型如图 4.6 所示。各个模块独立运行，并且拥有接口，保持和周围模块的通信。

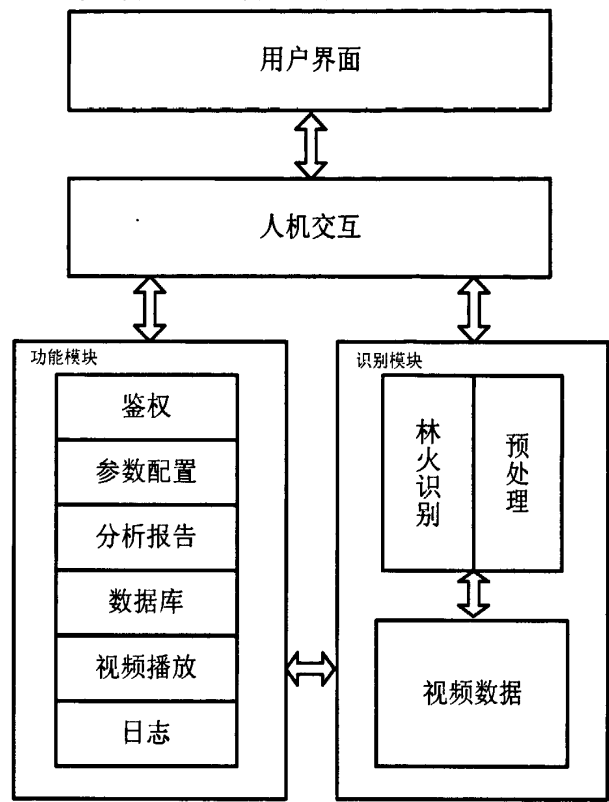


图 4.6 红外森林防火监控软件模型

4.5.2 林火监控系统软件设计

整个监控系统的软件集中在监控后端，主要包括：林火监控软件、系统管理软件、数据备份服务器软件、防火墙软件、UPS 软件。本文中的监控软件方案使用 Windows 平台。林火监控软件安装在客户端服务器上，该软件对监控区域远程监控，操作人员通过操作软件获取监控区域的信息，为预防火灾做好准备。下面分别阐述整个监控系统的软件

(1) 林火监控软件

利用无线传输网络，监控前端把监控区域的图像传输到监控中心。监控中心包括监控主机和外接的磁盘，林火监控软件在客户端主机上运行。外接磁盘用来存储以往的监控录像，方便操作人员复查。林火监控软件是运行在客户端的一个监视系统，通过它能够观察到由布置在野外的红外成像仪采集到的红外图像，这些图像是灰度图像。通过对灰度数据的预处理，并利用相关算法进行分析。当图像中有林火烟雾特征时，监控系统就会告警。操作人员通过告警提示信息对火灾或者火灾隐患作相应的处理，并且使用数据库保存火灾的信息，为以后火灾的预

防和统计做准备。在第五章中将更加详细的说明林火监控软件。

(2) 系统管理软件

控制中心服务器是一台服务器，是整个监控系统的调度中心。在这台服务器上部署了系统管理软件，系统管理软件用于管理整个林火监控系统。系统管理软件采用模块化的软件架构，依托数据库管理监控前端和客户端。数据库采用的是 Access 数据库。Access 是一个全面的数据库管理平台，提供了集成的商业智能工具，提升了数据管理方便性，同时，Access 数据库为关系型数据和结构化数据提供了更安全可靠存储功能。林火监控软件通过 ADO 访问数据库，数据库包含本地区与信息（LocalAreaInfo）、站点信息（StationInfo）、用户信息（UserInfo）、操作员信息（OperatorInfo）。系统管理软件的主要功能包含：

① 系统资源的管理

系统管理软件能够通过中心数据库对系统的所有软件资源、硬件设备、进行统一管理，如图 4.7 所示。所有资源及软硬件设备相关信息都保存在中心数据库，比如：红外成像仪、DVS、NVR 等。利用系统管理软件可以对设备做添加、修改、删除、建立连接、连接停止、数据备份等功能，用户利用系统管理软件提供的人机接口实现这些操作而不需要进入到数据库中。系统管理软件支持分布式、多中心的架构，这些中心拥有良好的同步机制和备份机制。

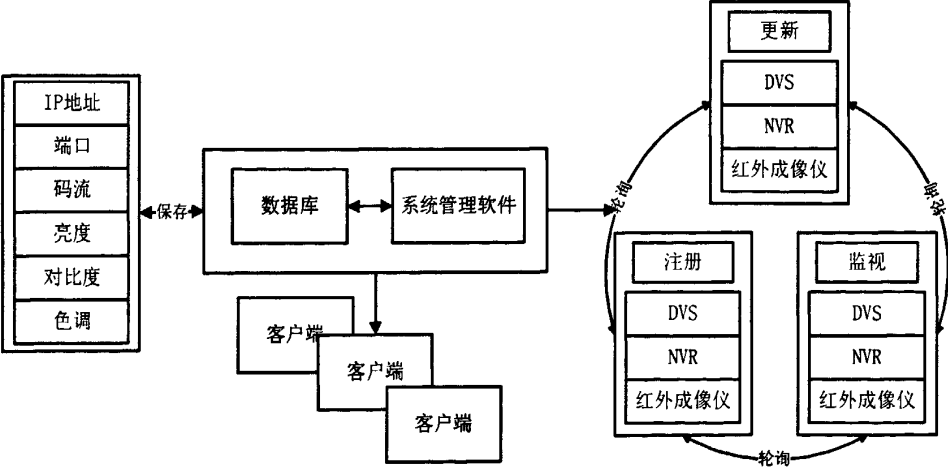


图 4.7 系统资源及设备管理

② 用户及权限管理

用户管理是对系统内的所有用户、用户组进行统一注册管理，所有用户信息集中保存在系统数据库中。用户的注册主要包括用户组、用户名称、密码的配置。而用户的权限由其自身权限与所在的用户组权限共同决定。用户权限即客户端的权限通常包括：用户的访问权限、控制设备的权限以及某些设备控制的优先级权限。

③ 网管功能

网管模块的功能在于可以自动检测并收集系统中所有设备及服务的状态信息，比如系统软硬件运行状态。服务状态、网络设备及存储设备状态，通过网管模块完成设备状态的采集，定期刷新给管理软件，并可以进一步将信息存储在系统日志中，或根据程序设定发送给客户端。比如，一旦发现采集的状态信息出现问题，管理软件会产生告警并发送给客户端。因此，可以说管理软件依靠检测各个设备的数据流状态维持着整个系统的稳定运行。林火监控系统中，视频流是最大的数据流，这些视频流可以分成时事浏览、视频备份等。此外还包含控制信息，比如：云台控制、系统报警信息等。林火监控系统的数据流信息如下图所示，

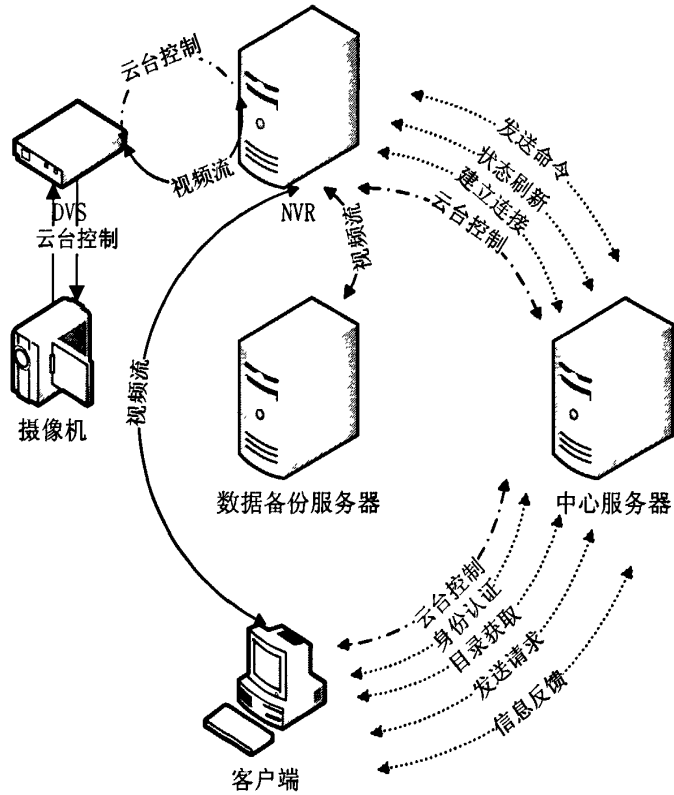


图 4.8 林火监控系统的数据流信息

(3) 数据备份服务器的软件设计

数据备份服务器通过局域网与监控主机直接相连，用来备份或存储系统日志、数据库和视频数据。使用数据备份服务器的好处是在以后避免意外原因引起的数据丢失，同时将数据和运行软件分开体现了分布式的原则，方便了其他运行主机对数据的访问，又因为与监控主机的分离提高了安全性。监控主机使用 socket 接口访问数据备份服务器，把相关的软件日志和数据库数据备份在服务器上。因此数据备份服务器软件的功能如下：①创建 socket 连接，将数据备份服务器和其他主机连接起来。②通过连接，数据备份服务器定时接收视频数据和日志文件，并



存储在硬盘上。在系统运行期间，只能根据系统的运行轨迹来推断错误出现的位置，这往往也是唯一的资料，特别是在高可靠性的情况下。这唯一的资料就是软件日志，利用日志文件能方便的定位软件出错位置，对以后软件的更新和调试提供了参考。这主要是为了方便以后调阅视频资料，并在林火发生后，能查阅视频，分析判断起火原因。③文件传输完毕，注销 socket 连接。

#### (4) 防火墙的设计

林火监控系统作为一种开放的系统，需要接受来自其他客户端的访问，因此面临着遭受网络攻击的可能性。林火监控数据对于分析林火发生起因十分重要。因此在设计的时候要考虑到数据安全的问题。防火墙技术是防范网络攻击最基本的手段之一，它是抵挡安全入侵的第一道防线。现有的是防火墙技术比较成熟，能够起到防止入侵的作用。

从狭义上说防火墙是指安装了防火墙软件的主机或者路由器系统；从广义上说防火墙还包括整个网络的安全策略和安全行为。防火墙分为以下四种：

①包过滤防火墙，包过滤防火墙一般在路由器上实现，用以过滤用户定义的内容，如 IP 地址。包过滤防火墙的工作原理是：系统在网络层检查数据包，与应用层无关。这样系统就具有很好的传输性能，很强的可扩展能力。但是，包过滤防火墙的安全性有一定的缺陷，因为系统对应用层信息无感知，也就是说，防火墙不理解通信的内容，所以可能被黑客所攻破。

②应用网关防火墙，应用网关防火墙检查所有应用层的信息包，并将检查的内容信息放入决策过程，从而提高网络的安全性。然而，应用网关防火墙是通过打破客户机/服务器模式实现的。每个客户机/服务器通信需要两个连接：一个是从客户端到防火墙，另一个是从防火墙到服务器。另外，每个代理需要一个不同的应用进程，或一个后台运行的服务程序，对每个新的应用必须添加针对此应用的服务程序，否则不能使用该服务。所以，应用网关防火墙具有可伸缩性差的缺点。

③状态检测防火墙，状态检测防火墙基本保持了简单包过滤防火墙的优点，性能比较好，同时对应用层是透明的，在此基础上，对于安全性有了大幅提升这种防火墙摒弃了简单包过滤防火墙仅仅考察进出网络的数据包，不关心数据包状态的缺点，在防火墙的核心部分建立状态连接表，维护了连接，将进出网络的数据当成一个个的事件来处理。可以这样说，状态检测包过滤防火墙规范了网络层和传输层行为，而应用代理型防火墙则是规范了特定的应用协议上的行为。

④复合型防火墙，复合型防火墙是指综合了状态检测与透明代理的新一代的防火墙，进一步基于 ASIC 架构，把防病毒、内容过滤整合到防火墙里，其中还包括 VPN、IDS 功能，多单元融为一体，是一种新突破。常规的防火墙并不能防

止隐蔽在网络流量里的攻击，在网络界面对应用层扫描，把防病毒、内容过滤与防火墙结合起来，这体现了网络与信息安全的新思路。它在网络边界实施 OSI 第七层的内容扫描，实现了实时在网络边缘布署病毒防护、内容过滤等应用层服务措施。

辨别一款防火墙性能的优劣，可以参考 RFC2544 标准，主要包括的方面是：网络吞吐量、丢包率、延迟、连接数等，其中吞吐量是重中之重。在成本允许的情况下选择下，应该选择吞吐量大的防火墙。

## 4.6 林火监控系统的流程

### 4.6.1 控制中心服务器的流程

控制中心服务器调度监控前端和客户端的运行，管理整个系统的资源，其流程如下所示：

(1) 控制中心服务器启动时首先从数据库中读出本地区域信息 (LocalAreaInfo) 并保留在内存。此后管理控制中心服务器的 ID 以及认证客户端登录时是否是本区域合法客户端均以 LocalAreaInfo 内的 ID 为依据，则结束程序。

(2) 控制中心服务器再从数据库读取基站信息并加载到内存，以确定前端基站的在线情况。因为视频服务器名和基站名相同，同时只有一路视频，所以视频通道名也与基站名相同。

(3) 建立命令通道以支持客户端、媒体服务器和数据备份服务器的链接。

(4) 开启基站链接线程，依次链接前端各基站。因为视频服务器是单路的，所以使用控制中心服务器相关模块完成视频服务器的接入和实时视频流的转发，并建立与视频服务器的 232 串口透明控制通道。然后获取对应云台的当前方位并保存到相应的站点信息表 (StationInfo) 中，接着分别获取镜头的脉冲行程和当前脉冲位置并保存到相应的基站信息中；建立该视频服务器与媒体服务器的链接，至此视频服务器接入成功。视频服务器支持断线自动重连。监控前端的视频服务器与控制中心服务器的登录和注销（或断线）不需要命令（注意修改基站信息中的在线标记），但控制中心服务器需要向在线的客户端发送视频服务器的在线消息，客户端收到后更新监控前端在线情况。

### 4.6.2 客户端的流程

客户端位于整个系统的后端，拥有登录控制中心服务器的权限以及获取前端

基站的访问。客户端的流程如下所示：

(1) 登录。客户端的登录窗口中输入的区域编号（即区域 ID）、管理服务器的 IP 地址、用户名、密码，只要登录成功便在硬盘保存，下一次登录窗口弹出时用户只需输入密码即可。客户端发送 LOGIN\_CLIENT 到控制中心服务器，若此时控制中心服务器和媒体服务器（NVR）链接还未成功，直接返回警告信息；若此时控制中心服务器和媒体服务器链接成功，则通过查询数据库认证其身份，若通过则返回客户端 LOGINED\_CLIENT 命令并将媒体服务器链接信息返回给客户端，若未通过，返回客户端 LOGINED\_ERROR 命令，且不添加媒体服务器的连接信息。客户端收到返回的 LOGINED\_CLIENT 后，如果登陆不成功，则将返回的提示信息提示用户并断开连接；如果登陆成功，则用刚收到的媒体服务器连接信息开始连接媒体服务器，连接不成功则断开重新登录。若媒体服务器连接成功，可关闭登录窗口，弹出应用程序主窗口。

(2) 获取本地区域信息和监控前端信息<sup>[49]</sup>。在主窗口所有初始化工作完成后发送 QUERY\_ALLAREA\_INFO，控制中心服务器收到后返回本区域信息 ALLAREA\_INFO，客户端接收该信息后紧接着向控制中心服务器发送 QUERY\_AREADVS\_INFO，控制中心服务器收到后返回 AREADVVS\_INFO 给客户端，客户端收到各个监控前端的信息后保留到内存，并在界面中显示出来，再向控制中心服务器发送注册命令 REGISTER\_CLIENT，控制中心服务器收到后向内存中的 ClientInfoBase 的 Map 中添加新项，表示有新的客户端登录上线，同时添加该用户登陆的系统行为日志。另外，控制中心服务器添加用户日志或者用户通过 USERLOG\_DVR 命令添加日志，在调用数据库操作的动态库时，UserName[50]都为空，动态库判断对于 UserLog 中 UserID[50]不等于“help”的需要查询用户信息表获取 UserName[50]信息后再保存该日志到数据库中。

(3) 获取对监控前端的访问权限。向控制中心服务器发送 QUERY\_PERMISSION 命令获取本用户对各基站的权限，控制中心服务器返回 DVS\_PERMISSION 命令，并向该客户端发送监控前端的区域编号、站点编号以及权限验证码。

(4) 接收监控视频（浏览信息保存、基站上线掉线要自动开启和关闭图像预览）。

(5) 客户端注销处理。客户端注销靠管理服务器的“掉线检测”，管理器检测到客户端掉线后，应该从 ClientInfoBase 的 Map 中删除该客户端，同时添加该用户已注销的行为日志和更新媒体转发队列及对讲相关逻辑，同时注意维护基站信息中当前云台控制权获得者的 ID。

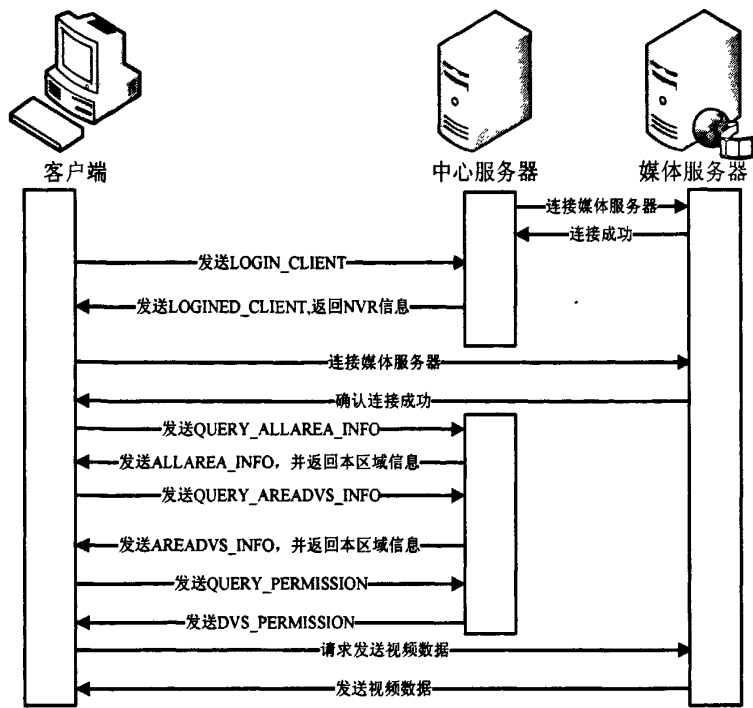


图 4.9 客户端消息流程图

客户端获取了各基站的权限后，各预览窗口便自动开启浏览前端图像的流程，用户也可手动停止某基站的监视。同时为了提高系统的资源利用率，减少不必要的资源浪费，禁止不同的窗口浏览相同基站的图像。并且基站掉线后需要搜索出预览该基站的窗口并且关闭该窗口的播放器。

监控前端的信息变化后，各在线客户端会收到控制中心服务器发来的UPDATE\_SITEINFO 命令，该命令传来了该基站的最新相关信息，客户端收到后需要更新显示。当基站上线后（REGISTER\_DVS 命令），控制中心服务器也会把该基站的最新信息传过来；当基站掉线后，搜索对应的查看窗口并关闭该窗口的播放器，还要修改内存中保存的基站信息中的在线标记，预览窗口和列表控件中的相关信息也需更新。

另外客户端在进行相关操作时及时填写行为日志结构 UserLog，并向管理服务器发送 USERLOG\_DVR 命令，管理服务器收到后添加到数据库。

4.7 本章小结

本章首先分析了当前的林火视频监控系统的功能需求。然后对目前运用广泛的可见光摄像头与采用红外成像设备的监控系统做了方案对比，得出了红外监控系统的优势。接着参考传统的视频监控系统，提出了红外林火监控系统的设计方案。最后，根据此方案设计了红外林火监控系统的流程。

## 第五章 林火监控软件的设计与实现

### 5.1 界面设计与实现

#### 5.1.1 界面的设计思想

GUI (图形界面接口) 是在计算机上利用图形方式显示的用户接口<sup>[50]</sup>。软件界面包含窗口、菜单、工具栏、对话框等元素。从文档数量的角度, 软件界面分为单文档窗口和多文档窗口。单文档窗口在一段时间内只支持对单一文档的操作, 而多文档界面可以同时多个文档操作。图形界面在使用方面要考虑对使用者的视觉和心理影响。随着软件开发同质化的影响, 优秀的界面设计有时比核心功能更受使用者青睐。用户界面的设计着眼于分析人在与周边事物接触的时候, 人在心理和行为方面的具体表现, 通过分析人的喜好设计出良好的人机交互方式。设计软件界面要求注意如下几点:

(1) 简洁明了。复杂的界面不但大量消耗系统资源, 而且增加了软件使用的难度, 不利于软件的推广。

(2) 风格一致, 各个窗口要以相同的风格显示出来, 这样既能体现软件的整体风格, 又能够区分开其他软件, 降低不必要的影响。

(3) 考虑用户的操作习惯, 并且能够让用户保存自己的操作习惯。让使用者能够在最短的时间掌握软件的使用方法, 提高效率。

(4) 稳定性, 对软件发生的异常状况或者是紧急信息主动提示告警。

#### 5.1.2 界面的实现

林火监控软件界面的设计原则是提供一种清晰、方便、安全可靠的软件界面, 增强使用者对软件的整体感受, 更进一步是追求一种更好的用户体验。因此使用 BCG 控件美化软件界面, 在视觉上让使用者赏心悦目。作为 C++ 语言描述的 BCG 扩展开发库, 其实现方法是对基本的 MFC 类进行扩展, 从面向对象的角度上说, 就是使用继承的方法, 创建一个原有对象的子类, 子类拥有父类的特性并且拥有自己的属性和方法。BCG 提供了众多子类供试用, 降低了软件界面美化的难度。同时在编码的初期阶段就可以继承到这些子类, 而不需要在软件编码完成后才考虑界面优化。软件界面设计是一种交叉式设计, 既要考虑软件设计流程、又要考虑软件对话框、菜单、工具栏等资源。这里运用软件流程为主, 软件资源为辅的方式介绍采用 BCG 实现林火监控软件界面设计, 其步骤如下:

### (1) 软件主体框架

BCG 提供了多种软件框架, 这些框架的风格包括 office2007、IE7 等。BCG 支持对已生成的工程文件进行美化。在 Visual Studio 2005 中添加软件框架分成两部, 首先, 添加需要的资源文件, 即 RC 文件。RC 文件中包含了软件中所用到的所有 Windows 资源, 包括: 字符集、图形、对话框等。然后, 对 OnInitDialog 函数作如下修改:

```

BOOL CFire_DetectionDlg::OnInitDialog()
{
    CBCGPDialog::OnInitDialog();
    CBCGPVisualManager2007::SetStyle( CBCGPVisualManager2007::VS2
007_LunaBlue);
    CBCGPVisualManager::SetDefaultManager (RUNTIME_CLASS
( CBCGPVisualManager2007));
    EnableVisualManagerStyle();
    return TRUE ; // return TRUE unless you set the focus to a control
}

```

最后, 将 CFrameWnd 和 CFormView 分别改成 CBCGFrameWnd 和 CBCGFormView。

### (2) 状态栏/菜单/工具栏的实现

在 MainFrame 类的头文件中包含有状态栏、菜单和工具栏的对象定义, 通过替换为 BCG 的类得到新的状态栏、菜单和工具栏。新的声明方式如下:

```

CBCGPStatusBar m_wndStatusBar; //状态栏
CBCGMenuBar m_wndMenuBar; //新的菜单栏
CBCGToolBar m_wndToolBar; //应用程序工具栏

```

### (3) 图标的选择

图标在选择的时候应该注意到两方面, 一是要可识别, 图标通常附在按钮上, 因此图标要尽量突出按钮的功能; 二是美观, 提升用户的使用感受。在 Windows 程序设计中, 图标文件一般都包含在 RC 文件中, 大小有 16×16 和 32×32 两种。方法是在利用 Visual Studio 2005 的资源管理器载入图标后, 需要再附着到按钮上, 代码如下:

```

((CButton*)GetDlgItem(IDC_BUTTON))->SetIcon(AfxGetApp()->LoadIcon(IDI
_ICON));

```

## 5.2 鉴权模块的设计与实现

### 5.2.1 鉴权模块的设计思想

鉴权又称作身份验证，是利用一些方法，确定用户的身份。林火监控软件中的身份验证利用了共享密钥的方法，这种方法的关键在于客户端和服务端都拥有某个账户的密码。在用户进行鉴权的时候，客户端将用户输入的密码发送给服务器，服务器检查用户提交的密码是否与服务器中的密码相一致，如果一致，则判断该用户鉴权成功；否则，该用户是非法用户。

林火监控软件的界面启动后，需要输入正确的用户名和密码才能登录，然后才能接收视频数据并对视频作实时分析。用户输入的账户密码传输给中心服务器的数据库。经过验证无误后，用户就可以接收视频数据了。具体流程如下所示：

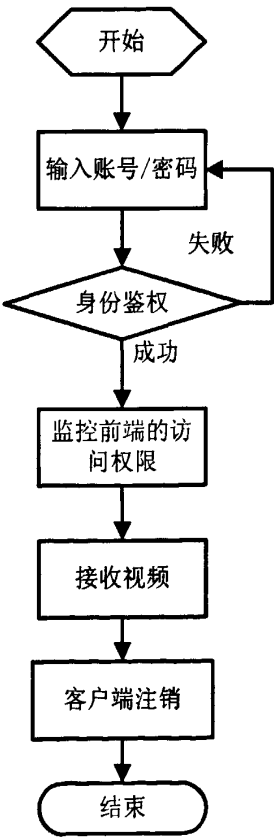


图 5.1 鉴权模块流程

### 5.2.2 鉴权模块的实现

客户端通过向中心服务器发送消息的形式保持与监控中心的信息交互，客户端向控制中心发送的消息包括三类：登录消息、询问消息、控制消息。

鉴权模块中运行的是登陆消息，消息的功能如下表所示：

表 5.1 登陆消息表

消息名称	编号	描述
LOGIN_CLIENT	1	申请身份验证
LOGINED_CLIENT	2	身份验证成功
LOGINED_ERROR	3	身份验证失败
REGISTER_CLIENT	4	注册新用户
CHANGPASSWORD	5	更改密码
CHANGPASS_RLT	6	更改密码成功

登陆消息包含客户端在登陆和注销时向中心服务器发送的消息。在实现中中使用 winsock 接口连接网络并传输数据。主要代码如下：

```
enum LogMessage
{
    LOGIN_CLIENT=1,
    LOGINED_CLIENT=2,
    LOGINED_ERROR=3,
    REGISTER_CLIENT= 4,
    LOGOUT_CLIENT=5,
    CHANGPASSWORD=6,
    CHANGPASS_RLT=7,
};
LogMessage lMsg;
//建立 socket
SOCKET clientsocket = socket(AF_INET,SOCK_STREAM,IPPROTO_TCP);
if(ServerSocket==INVALID_SOCKET)
{
    p->SetStatus(_T("error in create socket!"));
    return-1;
}
//连接对方主机
if( SOCKET_ERROR==connect(clientsocket,(SOCKADDR*)&their_addr,sizeof(t
heir_addr)))
{
    p->SetStatus(_T("connection is failed!"));
    return -1;
}
```



```

p->SetStatus( _T("connection is success!") );
//发送消息
switch (lMsg)
{
case LOGIN_CLIENT:
    SendBuffer=(const char FAR *)"LOGIN_CLIENT";
    break;
case LOGINED_CLIENT:
    SendBuffer=(const char FAR *)"LOGINED_CLIENT";
    break;
case LOGINED_ERROR:
    SendBuffer=(const char FAR *)"LOGINED_ERROR";
    break;
default:
    break;
}
Ret=send(clientSocket, SendBuffer, (int)strlen(SendBuffer), 0);
if(Ret==SOCKET_ERROR)
{
    p->SetStatus( _T("error in send!") );
}
//关闭 socket
closesocket(clientSocket);

```

客户端通过向中心服务器发送 LOGIN\_CLIENT 消息, 申请登录系统。而中心服务器通过比对输入的地域编号、账户、密码确定该用户是否拥有相应的权限。对于拥有权限的用户, 中心服务器会告知该用户并提供相应服务。在用户完成这些操作并且注销后, 中心服务器会收回这些权限。

## 5.3 参数配置的设计与实现

### 5.3.1 参数配置模块的设计思想

参数配置模块包含两部分一部分是配置文件, 另一部分是访问配置文件接口。配置文件中包含了软件的设置信息。配置文件包括区域和关键字两个部分。区域

(AreaName) 定义了一个范围。区域之间在逻辑上有很大的区别，包含在区域下的关键字 (KeyName) 都属于这个范围。配置文件的格式如图 5.2 所示，

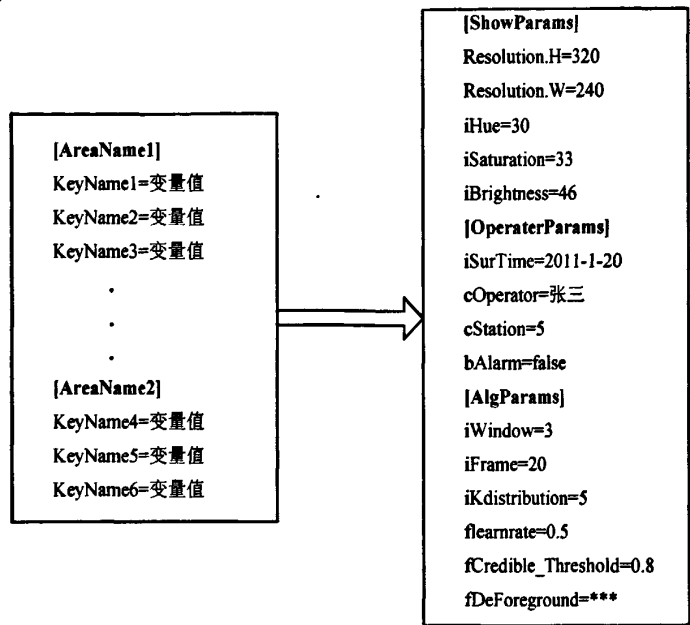


图 5.2 参数配置文件

访问配置文件接口一般在软件启动且登录成功后运行。它的主要功能是将前一次的配置信息保存在配置文件中，并在下一次软件初始化之后被调用。其流程如图 5.3 所示。林火监控软件中用到的参数如表 5.2 所示，

表 5.2 林火监控软件的参数

模块	数据项	变量	类型	长度	备注
显示参数	分辨率	Resolution	struct	8Byte	包含 weight 和 height 参数
	色调	iHue	int	4 Byte	
	饱和度	iSaturation	int	4 Byte	
	亮度	iBrightness	int	4 Byte	
操作参数	监控时间	iSurTime	int	4 Byte	主机时间为标准
	操作人员	cOperator	char[5]	5Byte	人员名称
	基站	cStation	char[15]	15 Byte	基站名称
	报警提示	bAlarm	bool	1 Byte	1 是告警, 0 是正常
算法参数	中值滤波窗口	iWindow	int	4 Byte	$Filter_w$ 默认 $3 \times 3$
	建模帧数 t	iFrame	int	4 Byte	默认 20frame
	高斯分布数 K 值	iKdistribution	int	4 Byte	K 越大, 处理能力要求越高, 默认 5
	学习速率 $\alpha$	flearnrate	float	4 Byte	默认 0.5
	可信与不可信阈值 T	fCredible_Threshold	float	4 Byte	类型与权重值一致, 默认 0.8
	判决阈值下限	fGThreshold	float	4 Byte	防止前景比例过大

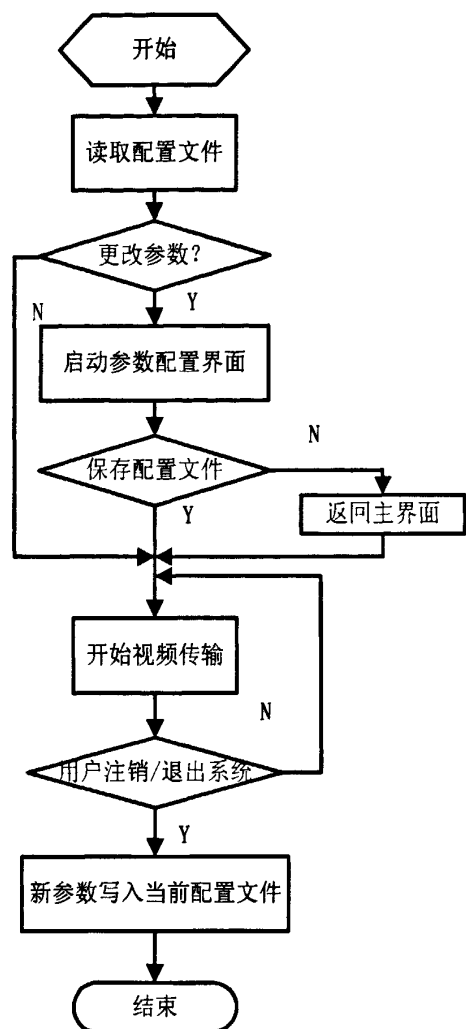


图 5.3 参数配置模块流程

5.3.2 参数配置模块的实现

访问参数配置文件的实现有两种方式：一种是使用 Windows SDK，SDK 是微软提供给软件开发人员的 API 接口，这些接口拥有较强的功能；另一种是自行定义 C++ 类，实现对配置文件的操作。虽然调用 SDK 能够快速实现一些功能，但是调用 SDK 会导致模块不可移植；同时，很多 SDK 为了通用性，在效率方面也有欠缺。本文使用自行设计的参数配置模块增加模块的移植性和效率。把参数配置模块分成两层，每一层是单独的一个类，两个类之间存在继承的关系，第一层 CFileConfig 是一个基础类，提供了对基本数据结构的访问；第二层是专门针对林火监控系统实现的类 CUserCfg，提供了访问参数配置文件 Config.dat 的接口。

参数配置模块的移植难度体现在对字符集的支持方面，常用的字符集有 ANSI 和 Unicode 两种。ANSI 是早期的编码，ANSI 采用 8 位编码能方便的对 26 个英文

字母编码。但为了满足对其他语言的支持，国际标准化组织制订了 Unicode 编码。很多系统如 Wince 系统只支持 Unicode，因此实现中要考虑即支持 ANSI 又兼容 Unicode。采用条件编译指令就能够达到这种效果。方法如下：

- (1) 对程序中所有用字符类型 char 声明的变量用 tchar 代替。
- (2) 如果需要 Unicode 编码，那么在每一个需要支持 Unicode 字符集的文件加入

```
#ifndef UNICODE
#define UNICODE
#endif
```

否则什么也不加。之后，系统在编译的时候会判断是否存在 UNICODE，若存在使用 wchar 定义，否则使用 char 定义。

基础类 CFileConfig 包含三类函数：访问函数、读取函数、存储函数、编辑函数，表 5.3 是这四类函数的定义。

表 5.3 CFileConfig 类的成员函数

类别	函数名	返回值	备注
访问函数	OpenCfgFile	BOOL	打开配置文件
	CloseCfgFile	BOOL	关闭配置文件
读取函数	GetFloatCfg	BOOL	读取 float 类型数据
	GetUShortCfg	BOOL	读取 unsigned short 类型数据
	GetULongCfg	BOOL	读取 unsigned long 类型数据
	GetTCharCfg	BOOL	保存 tchar 类型数据
存储函数	SetFloatCfg	BOOL	保存 float 类型数据
	SetUShortCfg	BOOL	保存 unsigned short 类型数据
	SetULongCfg	BOOL	保存 unsigned long 类型数据
	SetTCharCfg	BOOL	保存 tchar 类型数据
编辑函数	InsertAreaName	BOOL	插入新的区域
	InsertKeyName	BOOL	插入新的关键字
	DeleteAreaName	BOOL	删除区域
	DeleteKeyName	BOOL	删除关键字

CFileConfig 类封装成动态链接库，将封装好的库交给 CUserCfg 调用。CUserCfg 类是对 CFileConfig 的公有继承，CUserCfg 类主要访问林火监控软件的参数，CUserCfg 类在客户端成功登陆后声明，先使用 Open 函数打开文件并使用 Get 函数读取所有参数值，Get 函数是对原有 CFileConfig 中的读取函数进行封装。在客户端注销、软件关闭后，使用 Set 函数将新的参数值写入到 Config.dat 文件中保存，最后调用 Close 关闭文件。

## 5.4 林火识别模块设计与实现

### 5.4.1 林火识别模块设计

林火识别模块包含两个部分：模块接口和林火识别算法。模块接口是对外部其他模块而言，向外部模块提供服务的或者是获得外部模块的服务。通过这个借口林火识别模块的算法有了输入数据，并且能够向其他模块输出判决结果。

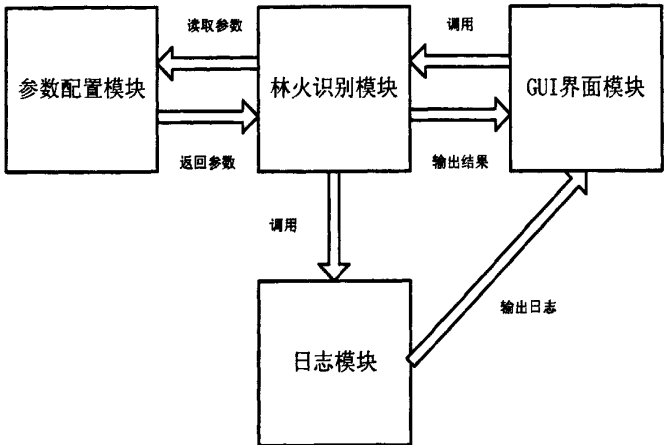


图 5.4 林火识别模块与其他模块的关系

林火识别算法主要包括几点：

- (1) 建立混合高斯模型，取前  $t$  帧图像建模。
- (2) 读取当前图像并用  $Filter_w$  的滤波器做预处理滤除噪声。将混合高斯分布按照  $\omega_i / \sigma_i$  从大到小排列，取当前像素与混合高斯模型各个分布进行比较，比较方法按照三倍标准差原则以及判决阈值  $C$  原则。若当前图像和背景模型匹配则转入步骤 (3)，否则转入步骤 (4)。
- (3) 按照第二章的介绍更新高斯分布的期望、方差、权重。同时删除过时分布，存在干扰点时建立新的高斯分布。
- (4) 停止多高斯分布背景模型的更新，连续比较几帧图像中可疑目标点在图像中的比例，按照火灾烟雾的动态特征，对可疑区域进行判断，如果是可疑区域转第 (5) 步，如果是干扰目标则转第 (3) 步。
- (5) 提取可疑区域，并对监控人员发出报警。

### 5.4.2 林火识别算法的实现

在接收到视频后，对视频做解码处理，然后读取单帧图像交给林火识别模块检测。检测过程如下：

- (1) 滤波处理

受数据采集和传输的影响，在对灰度图像做林火识别处理之前，要滤除图像中的干扰噪声，使用中值滤波对图像做处理，处理流程如图 5.5 所示，

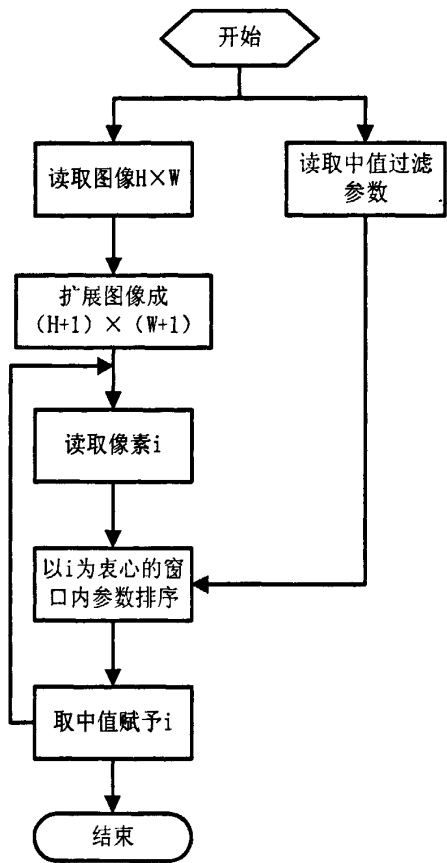


图 5.5 中值滤波流程

对  $H \times W$  图像作中值滤波处理，将图像扩展 1 个像素，即得到  $(H+1) \times (W+1)$  对接收到图像中的第  $i$  帧周围窗口内的点按照从大到小的方式排列，取中值作为第  $i$  点的像素值。依次遍历图像中的所有点，得到滤波处理后的图像。

(2) 烟火识别

将滤波处理之后图像，交给林火识别算法做下一步的处理。林火识别算法通过前面介绍过的建立混合高斯模型的方法实现。使用 C++ 语言描述林火识别算法，定义类 CGaussMixModel。CGaussMixModel 包含属性和方法，如下表所示：

表 5.4 CGaussMixModel 的成员变量和成员函数

类别	名称	类型/返回值	访问权限	描述
属性	iFilterWindow	int	Private	中值滤波窗口
	iFrame	int	Private	建模帧数
	iK_distributions	int	Private	高斯分布上限
	dLearnRate	double	Private	学习速率
	fCredible_Threshold	float	Private	可信与不可信阈值 T
	fFGThreshold	float	Private	防止前景比例过大的阈值 C
	GaussDistribution	struct	Private	高斯分布期望、方差、权重值
	CreateGaussMixModel	BackGroundState	Public	创建混合高斯分布

方法	UpdateGaussMixModel	int	Public	更新分布
	InsertNewGauss	bool	Public	建立新分布
	DeleteGauss	void	Public	删除过时分布
	DeleteGaussMixModel	bool	Public	删除混合高斯分布
	GetAllParameters	struct	Public	获取所有参数
	SetFilterWindow	void	Public	设置中值滤波窗口
	SetFrame	void	Public	设置建模帧数
	SetK_distributions	void	Public	设置高斯分布上限
	SetLearnRate	void	Public	设置学习速率
	SetCredible_Threshold	void	Public	设置可信与不可信阈值 T
	SetDeForeground	void	Public	设置防前景比例过大的阈值 C

视频在预处理之前,其相关信息包括通道、像素深度等包含在结构体 VideoInfo 中。视频信息结构体 VideoInfo 定义如下:

```
typedef struct VideoInfo
{
    int nSize; //VideoInfo 大小
    int nChannels; //通道
    int depth; //像素的位深度
    int width; //图像宽像素数
    int height; // 图像高像素数
    int imageSize; //图像数据大小(单位字节)
    char *imageData; //指向排列的图像数据
    int widthStep; //排列的图像行大小, 以字节为单位
} VideoInfo;
```

视频在接收并读取后,文件信息保存在 VideoInfo 中,有了这个结构体就能方便地对视频做分析。除了包含自身的构造、析构函数之外,类 CGaussMixModel 还包含众多成员函数。林火检测的基础是运用混合高斯模型,因此首先要建立混合高斯模型,定义如下函数建立混合高斯模型,

```
//创建背景模型
BackGroundState& CreateGaussMixModel (VideoInfo* newvideo,
GaussBGMPParams* pBGMPParams);
```

GaussBGMPParams 结构体包含了建立高斯分布的基本信息,其中包含有输入参数和输出参数,比如可信与不可信阈值 T,防止前景过大阈值 C 等都是输入参数而期望、权重、方差属于背景参数,具体定义如下

```
typedef struct GaussBGMPParams
{
```

```

float fCredible_Threshold;// 可信与不可信阈值 T,
float fFGThreshold;//防止前景过大阈值下限,
int idistributions;//分布数,
float fmean;//期望,
float weight;//权重,
float variance;//方差,
}GaussBGMPParams;

```

返回值是 BackGroundState 结构体, BackGroundState 结构体定义了当前背景模型的状态包括前景状态、背景状态等, 其定义如下:

```

typedef struct BackGroundState
{
    VideoInfo* background;//背景
    VideoInfo* foreground;// 前景
    void* storememory;//存储内存
    UpdateGaussMixModel update;//函数指针, 更新混合高斯模型
    DeleteGaussMixModel delete;//函数指针, 删除混合高斯模型
} BackGroundState;

```

混合高斯模型的更新主要出现在两个地方, 一次是在当前像素与高斯模型匹配的条件下, 这时判断出该像素不是可疑点, 要将该点归并到背景中去; 另一次是在出现可疑点但是不存在可疑区域的条件下。同样的, 可疑点并不一定能形成可疑区域, 还有可能是干扰点, 也将这些点归入背景模型。更新函数定义如下:

```

//更新混合高斯模型
int UpdateGaussMixModel(VideoInfo* curframe,//当前帧
    GaussBGM* bgmodel, //高斯模型
    double dLearnRate);//学习速率
//返回值是前景区域数目

```

GaussBGM 定义了高斯分布的数据结构, 包含了各个高斯分布的背景状态、当前帧数、高斯分布参数等信息, GaussBGM 结构体的定义如下:

```

typedef struct GaussBGM
{
    BackGroundState backstat;//背景状态
    GaussBGMPParams* params;//高斯分布参数
    int iframe;//帧数

```



```
bool bmatch;//是否匹配
} GaussBGM;
```

在混合高斯模型中，有的高斯分布长时间不能与当前像素匹配，这些分布就是不可信分布，不可信分布反映的是前景的变化需要从混合高斯模型中删除，用下面的函数删除：

```
//删除过时分布
```

```
void DeleteGauss (GaussBGM** pGauss);
```

以上是混合高斯模型主要成员函数的定义，在实现中使用中将 CGaussMixModel 声明成动态链接库，方法是在 FireDetect.h 文件中加入

```
#ifndef FIREDETECT_H
#define FIREDETECT_H
#endif

class _declspec(dllexport) CGaussMixModel
```

使用动态链接库的好处是：（1）方便林火识别模块的更新。不同与静态链接，动态链接库文件独立存在，只要将林火识别模块动态链接库文件覆盖原有文件就可以了。（2）与主界面模块分离，开发的时候不用考虑主界面或者平台的限制，可移植性好。（3）模块能够单独实现，单独测试、不需要依赖其他模块，开发难度降低，开发周期减少。

## 5.5 分析报告模块设计与实现

### 5.5.1 分析报告的设计

林火分析报告是由分析报告模块自动生成 Word 文档或者是自定义 RPT 文件，报告的内容包含视频拍摄时间、地点、操作人员、视频分析、分析时间、告警信息、说明信息以及操作员希望保存的视频监控信息，使用 UML 序列图描述了林火分析报告的创建顺序，如图 5.6 所示。

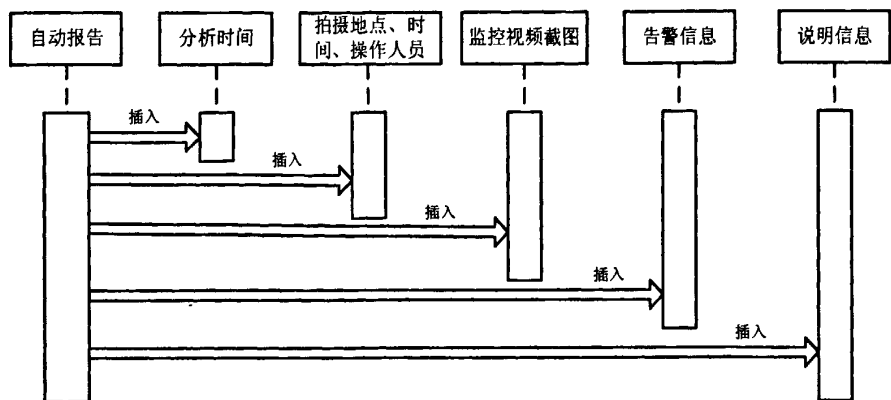


图 5.6 林火监控系统分析报告 UML 序列图

对报告的操作包括创建报告、插入信息、保存报告、打印报告。操作步骤是可以更改的，通过读取接口数据判断哪些步骤需要执行。创建报告要求输入报告的名称，指定文件格式并判断当前目录下有没有同名文件存在，如果没有就创建，否则继续输入文件名。在创建好空文档后，就能够插入内容。内容形式包含文字、图片、表格等。保存报告包括发送请求保存，然后将报告保存起来。打印报告要先对报告做预览操作，在报告整体修改好之后，运行打印操作。报告模块流程如图 5.7 所示。

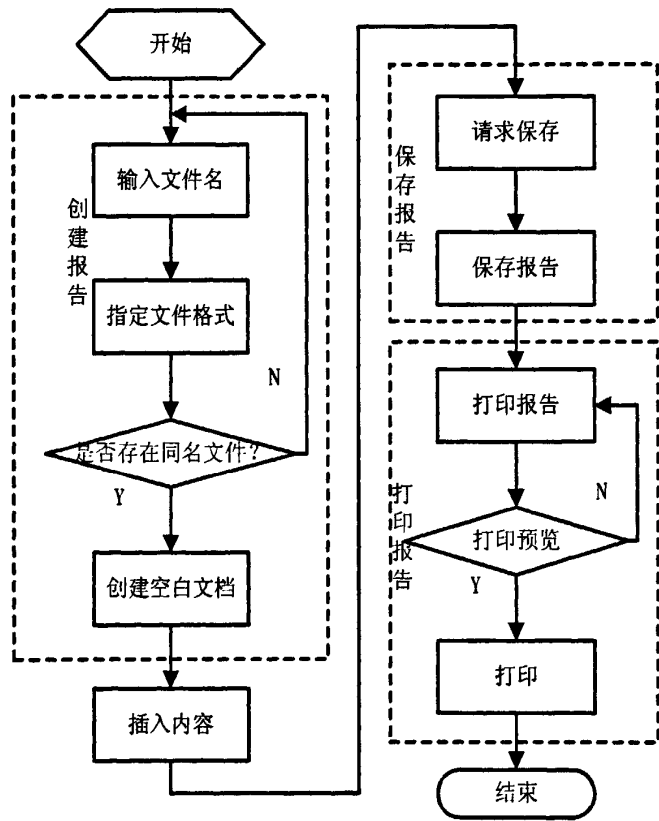


图 5.7 林火监控报告流程

5.5.2 分析报告的实现

分析报告有两种格式，因此在实现的时候也要考虑两种不同的格式。根据格式的不同分别实现两种报告。

RPT 文档主要包含三个部分：文字、图片和表格，三种内容用标识区分，调用不同的实现函数。因此在实现的时候三部分的调用函数都有所不同，具体见表 5.5 所示，

表 5.5 RPT 文件访问函数

操作名称	函数名	返回值	访问权限	描述
文字操作	InsertText	BOOL	Public	插入文字
	ChangeTextColor	void	Public	更改文字颜色
	ChangeTextFont	void	Public	更改字体
	AlignText	void	Public	对齐文本
图片操作	InsertPicture	BOOL	Public	插入图片
表格操作	InsertTable	BOOL	Public	插入表格
	InsertWordinTab	BOOL	Public	向表格插入文字
文件操作	InitialRPT	BOOL	Public	初始化 RPT 文件
	InputFileName	BOOL	Public	输入文件名
	SaveRPT	BOOL	Public	保存文件

生成 Word 文档格式的报告利用 COM 控件完成报告模块，使用 CWordOffice 提供的接口完成对报告文件的操作。首先创建空白 Word 文档，在文档中插入文字、图片和表格，最后利用多进程的方法调用 Word 进程。创建 Word 文档的方法是：首先要对各类对象初始化，最先调用 CreateDispatch 函数创建\_Application 实例，利用该实例的 GetDocuments 函数初始化 Documents 的实例，再初始化\_Document、\_Font、Selection 实例。然后，调用 \_Font 实例的 SetSize、SetName、SetColor 等函数改变文字的大小，字体和颜色。最后，调用 Selection 实例的 TypeText 函数插入文字；用 InlineShapes 实例的 AddPicture 函数插入图片；用 Table 实例的 Add 函数添加表格。在读取接口信息后，按照 UML 序列图中的顺序依次插入内容。之后保存文档完成操作。

5.6 数据库模块设计与实现

5.6.1 数据表的设计

林火监控软件的数据库包含有六个实体：本地区域信息、基站信息、用户信息、监控视频、用户对基站权限、用户日志，如图 5.8 所示。为了减少在今后数据维护的过程中，不会因插入、删除、更新产生异常，需要规范化的表结构。规范化的表结构设计能够避免数据库使用和维护中的问题，而且减少冗余信息的产

生，节省系统资源。运用 E-R 模型设计数据表，将六个实体作为基本出发点，把实体和联系看做数据模型的基本元素，这些基本元素结合形成用户视图。下图利用 UML 描述 E-R 模型，每个关系的第一个属性作为主键。

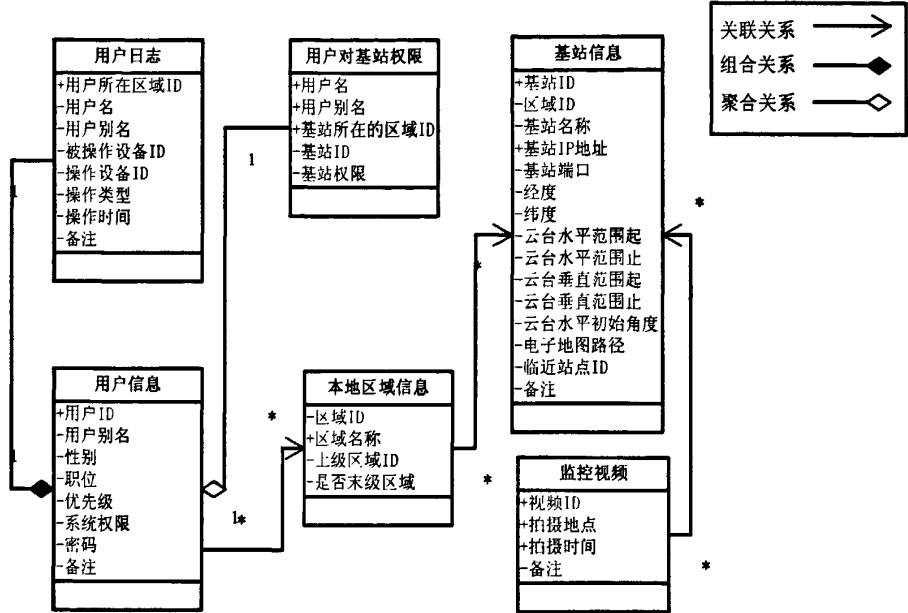


图 5.8 UML 描述 E-R 模型

林火监控软件的数据库需要使用 Access，Access 的好处是集成环境，能够方便的处理多样的数据。另外，Access 采用了统一的管理工具和 COM 接口，提升了 Access 配置是灵活性和可管理性。

表 5.6 本地区域信息 (LocalAreaInfo)

区域 ID	Char(4)	AreaId
区域名称	Char(30)	AreaName
上级区域 ID	Char(4) “0000” 代表无	HighAreaId
是否末级区域	Varchar(2), ‘0’-否, ‘1’-是	IfLastArea

表 5.7 基站信息 (StationInfo)

区域 ID	Char(4)	AreaID
基站 ID	Char(20)	StationID
基站名称	Char(32)	StationName
基站 ip 地址	Char(20)	StationIP
基站端口	int	StationPort
经度	Decimal(10,6) 6 位小数	Longitude
纬度	Decimal(10,6) 6 位小数	Latitude
云台类型	Int	PanID
云台水平可视范围(度)起	int	PanPlane_U
云台水平可视范围(度)止	int	PanPlane_D
云台俯仰可视范围(度)起	Int	PanVertical_U
云台俯仰可视范围(度)止	Int	PanVertical_D
云台水平初始角度	Decimal(10,6) 6 位小数	PanPlane_Init
电子地图路径	Varchar(100)	MapPath
临近站点 ID	Char(20)	Neighbor_ID

备注	Varchar(200)	Remark
----	--------------	--------

表 5.8 用户信息 (UserInfo)

用户 ID	Char (20)	UserID
用户名	Char(30)	UserName
性别	Char(2)	UserSex
职位	Char(10)	UserPlace
优先级	Char(32)	UserPriority
系统权限	Char(32) 32 位	SysPower
密码	Char(128)	UserPassWord
备注	Varchar(200)	Remark

表 5.9 用户对基站权限 (StationPermissions)

用户名	Char(20)	UserID
用户别名	Char(30)	UserName
基站所在的区域 ID	Char(4)	AreaId
基站 ID	Char(20)	StationID
基站权限	Char(32) 32 位	UserToStationPowe

表 5.10 用户日志 (UserLog)

用户所在区域 ID	Char(4)	AreaID
用户名	Char(20)	UserID
用户别名	Char(30)	UserName
被操作设备 ID	Char(20)	Equipment1
操作设备 ID	Char(20)	Equipment2
操作类型	Char(2)	OpID
操作时间	Datetime	OpTime
备注	Varchar (200)	Remark

表 5.11 监控视频 (SurveillanceVideo)

视频 ID	Char(20)	VideoID
拍摄日期	Datetime	ShootTime
拍摄地点	Char(20)	ShootLocation
备注	Varchar (200)	Remark

5.6.2 数据库模块的实现

作为 COM 的 ADO 是访问数据库的接口,ADO 凭借高速度和低内存占用成为微软支持的数据库平台下访问数据库最有效的手段。ADO 模型包括了下列对象: Connection、Command、Recordset、Field、Parameter、Error、Property、集合、事件。最常用的是 Connection、Recordset 及 Command 这三个对象。在 Visual Studio 2005 中使用 ADO 控件操作数据库的步骤是:

- (1) 在 Visual Studio 2005 中添加 ADO 控件。
- (2) 创建连接,声明 \_ConnectionPtr 的对象 pConnection,再调用 pConnection 的 CreateInstance 函数创建连接。
- (3) 打开数据库,字段 \_bstr\_t strConnect 设置为

“driver={data.mdb};server=data\_server;uid=\*\*\*;pwd=\*\*\*;Database=data”

(4) 创建并打开记录集。\_RecordsetPtr 的对象 pRecordset 调用 CreateInstance 函数创建记录集再调用 Open 函数打开记录集。访问数据库时要加入 try-catch 异常处理机制, 该机制能够降低软件的运行风险提高健壮性。这里需要在 try 语句中加入打开记录集的语句, 而在 catch 语句中插入处理方法以及提示告警信息。

(5) 插入、删除记录。pRecordset 调用 PutCollect 函数插入记录。调用 Delete 函数删除记录。同上, 这里也要加入 try-catch 语句。

(6) 对数据库访问结束后利用 Close 函数关闭记录集和连接。

## 5.7 其他模块的设计与实现

### 5.7.1 视频播放模块的设计与实现

使用 DirectShow 完成 MPEG-4/H264 的解码和播放, DirectShow 属于 DirectX 家族 (DirectX 还包括 DirectDraw、DirectInput、DirectSound、Direct3D 等组件)。在调用 DirectShow 开发视频播放模块之前需要安装 DirectX SDK。DirectX SDK 采用 COM 架构, 因此使用 DirectShow 开发应用程序开发周期比较短, 可靠性高。IMediaDet 接口是 DirectShow 的关键接口, 包含了从文件中读取重要信息的方法。

利用类 CCamera 实现视频显示的功能, CCamera 类的定义如下:

```
Class CCamera
{
private:
    bool bConnected;
    int iWidth, iHeight;
    long lBufferSize;
    VideoInfo * pFrame;
public:
    CCamera();
    virtual ~CCamera();
    bool OpenCamera(int iCamID, int iWidth, int iHeight);
    void CloseCamera();
    static int CameraCount();
    static int GetCameraName(int iCamID, char* cName, int iBufferSize);
    int GetWidth(){return iWidth;}
```

```
int GetHeight(){return iHeight;}  
VideoInfo * GetFrame();  
};
```

OpenCamera 函数打开编号是 iCamID 的摄像头, iWidth 是摄像头分辨率的宽度, iHeight 是分辨率的高度。使用结束后调用 CloseCamera 关闭摄像头, CloseCamera 也自动由析构函数调用。CameraCount 返回摄像头的数量, 输入摄像头编号给 GetCameraName 返回该摄像头的设备名称。GetWidth 和 GetHeight 分别返回图像的宽度和高度。QueryFrame 函数抓取一帧图像。

### 5.7.2 日志模块的设计与实现

一个日志系统根据其过程, 可以分为日志来源, 系统控制, 日志输出, 日志存储。根据这个过程, 我们可以将整个系统分为 4 个模块, 并加以抽象。

#### 1、日志来源

日志内容可以来源于任何其他系统, 但对日志系统来说, 这是个格式化的缓冲区。对于日志系统来说, 任何内容都是合法的。最重要的是, 日志系统必须提供一个简单的规则, 为后续的管理和检索提供方便性和灵活性。在传统的 printf 格式中, 是很难维护一个格式化的日志输出。文本方式对人来说阅读方便, 但不容易检索, 特别是在大量日志的情况下, 更不好维护。

#### 2、日志控制

系统控制的重点在于控制日志内容在日志系统中的流转过程。比如日志输出目的地, 比如日志的输出级别。在 apache 的 Logging 项目中, 他们提供了一个和平时不太一样的输出目的地, telnet。这和传统的 stdout、stderr、syslog 有很大的区别, 便于远程管理, 更大的潜力在于, 可以在运行期, 通过登录 telnet 来动态调整系统环境配置。

#### 3、日志输出

日志在控制台输出是比较常见的, 但如何考虑为系统的可靠性提供支持, 以及大量日志内容的情况下, 这个一般不予考虑的。在控制台输出的, 只会是非常核心的内容或者是致命的错误, 况且, 在有些情况下, 不一样会有控制台。我们一般在这种情况下, 都倾向于将日志输出到文件。但对一个完善的日志系统, 日志输出和日志存储又是有区别的。日志存储是日志输出到文件的一种方式。日志输出也是日志控制的一个内容。

#### 4、日志存储

日志存储在很多小型系统往往并不需要关注, 一个可靠性要求很高的系统中,

对日志存储却是极为苛刻。就是在现在的数据库系统中，也必须依赖日志的存在，来还原操作。考虑到以上几点，日志模块的流程如图 5.9 所示。

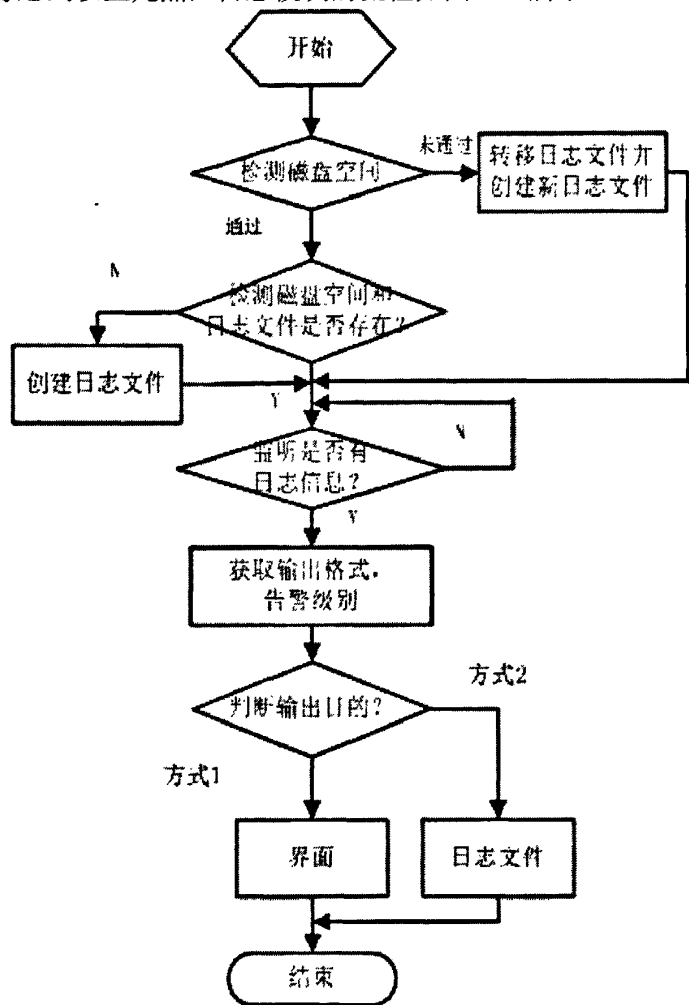


图 5.9 日志模块流程图

参考开源的 Log4j 模型设计了一个三层的分级日志模块。第一层是检测层，为了保障日志系统长期稳定的运行，日志文件过大会影响到日志文件 System.log 和整个系统的稳定，检测层通过检测剩余磁盘空间的方式确定是否需要转移日志文件。第二层是分级告警层，采用五级告警形式 FATAL、ERROR、WARM、INFO、DEBUG。调用的时候，选择合适的告警类型输出日志信息。日志模块第三层是输出层，面向日志文件和界面，提供了多种输出和显示方式。日志信息的格式主要由时间、告警级别、消息内容等组成，可以通过多种形式比如中/英文、显示方式输出告警信息。

日志模块通过类 CLog 实现，CLog 的描述如下所示：

```
Class CLog
{
```



```
public:
    CLogger(const char* filename);
    virtual ~CLogger();
public:
    enum LogLevel{FATAL,ERROR,WARM,INFO,DEBUG};
    enum Destination{UI,File};
    char* get_curr_time();
    void set_loglevel(LogLevel level);
    int Output_Log (tchar* msg,LogLevel level, Destination dest);
private:
    LogLevel level;
    Destination dest;
    tchar* strtime;
    tchar* buf;
    FILE* fp;
};
```

## 5.8 本章小结

本章依次介绍了用户界面 GUI、鉴权模块、参数配置模块、林火识别模块分析报告模块、数据库模块、视频播放模块、日志模块的设计，并根据设计方案分别实现各子模块的功能。



## 第六章 林火监控软件的测试和分析

### 6.1 测试准备工作

#### 6.1.1 测试环境的搭建

为验证红外检测软件<sup>[51]</sup>的功能，在实验室组建以下的测试场景，通过测试结果论证本文工作的正确性。如图 6. 1 所示，红外成像仪采集森林中的视频数据，通过视频接口传给 DVS，经过压缩后传输给监控主机。监控主机上的林火监控软件显示视频，并对视频数据做林火识别分析。

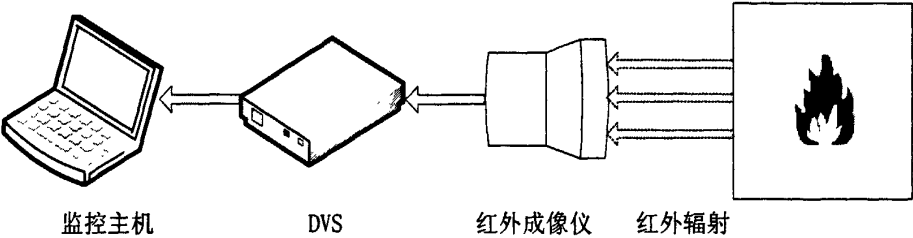


图 6.1 测试场景模型

① 林火监控软件运行主机软硬件环境：

硬件配置：Inter(R) Core(TM) i3 CPU 2.13GHz、内存 3G

操作系统：Windows 7

数据库：Access

② 红外探头组件，如图 6.1 所示。组件参数如表 6.1 所示，

表 6.1 红外探头组件参数表

工作波段	8~14 $\mu m$
视场	13.7° × 10.3°
调焦	电动
外接电源	12/220VAC 适配器, 50/60Hz
工作温度	-20℃~50℃
重量	2KG

#### 6.1.2 测试指标分析

软件测试指标分为负载测试、容量测试、功能测试三种类型，每种类型包含多种指标<sup>[52]</sup>：（1）功能测试，软件模块在运行中的功能指标，反映了各个模块功能是否正常。（2）负载测试，负载测试反映了软件模块在运行中要分配资源，包括 CPU 占用率、内存使用量等。（3）容量测试，也就是吞吐量测试，主要测试各个模块处理大量数据时的性能。按照表 6. 2 中的测试指标对每个模块都做了多

次测试，测试情况将在下面的几节中做具体分析。

表 6.2 测试指标

软件模块	功能测试	负载测试		容量测试
		CPU 占用率	内存使用量	吞吐量测试
软件界面	界面友好程度、控件可操作、提示信息完善程度	√	√	
鉴权	远程连接是否正常、能否正常发送、接收消息	√	√	
参数配置	能否正常读取、写入参数配置文件	√	√	
林火识别	能否准确判别烟火	√	√	
分析报告	能否正常生成报告，报告内容的是否全面，能否正常打开报告	√	√	√
数据库	能否连接到数据库文件中，能否正常读写数据	√	√	√
视频播放	能否正常在指定窗口中播放视频	√	√	
日志	能否正常读、写日志文件能否正常生成日志信息	√	√	√

6.2 模块功能验证

6.2.1 软件界面验证

软件界面的测试原则应该按照完整性、内容一致性、界面友好性、提示信息指导性等原则进行测试。（1）完整性，显示时应考虑数据自适应的宽度。测试数据的记录数超过一个显示区域，以验证满屏，其窗体是否有横向、纵向滚动条或换页打印，界面显示是否正常。（2）内容一致性，显示的内容应该与内存数据一致，并且多个界面的同一个数据应该一致。（3）界面友好性，对分析数据应按用户习惯进行分类、排序。对一些输入、输出、修改接口要在明显的位置上有“完成”提示信息。（4）提示信息指导性，用户提示信息应具有指导性，在软件处理关键业务的处理时，提示用户应用程序正在进行的处理，在处理结束后再提示用户处理完毕。同时，对输入参数的界面应在输入界面提供相应的规则描述，当输入数据不符合规则时应提示用户是否继续。测试的顺序按照登陆界面、主界面、参数配置对话框、分析报告界面、日志界面以及视频显示依次测试。由于篇幅所

限，只贴出主界面的测试效果图。

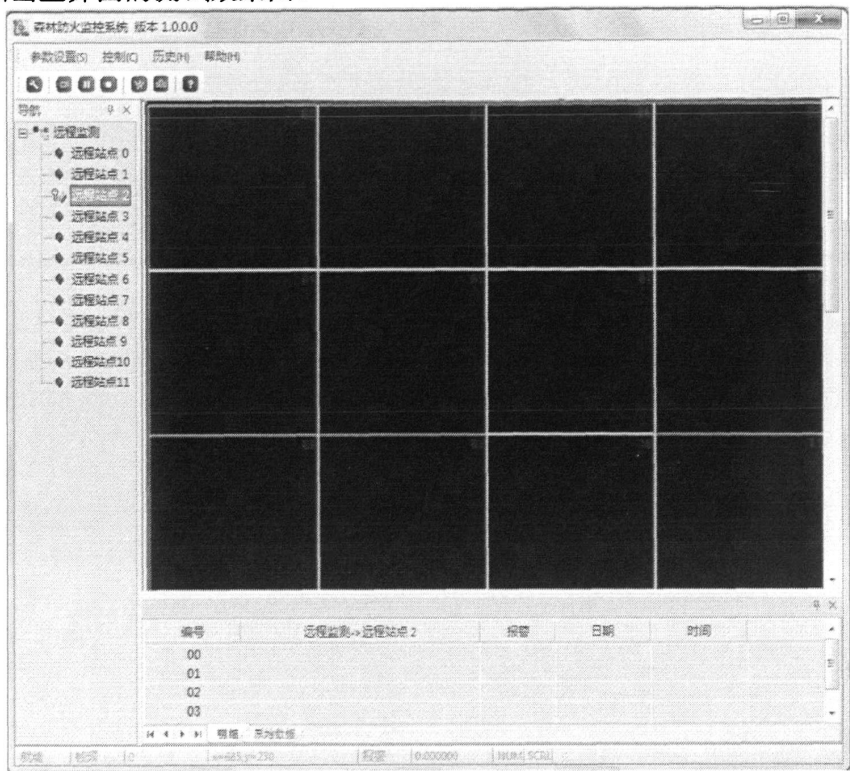


图 6.2 主界面的测试效果

6.2.2 鉴权模块的验证

鉴权模块的测试方法是在客户端和服务端分别部署客户端程序和服务器消息接收程序，两台主机连接在同一个局域网上。测试的步骤是：（1）打开客户端和服务器的测试程序；（2）在客户端程序中填写登录信息，确认后向服务器发送信息；（3）在服务器端接收用户端发送的信息。测试效果如下图所示，

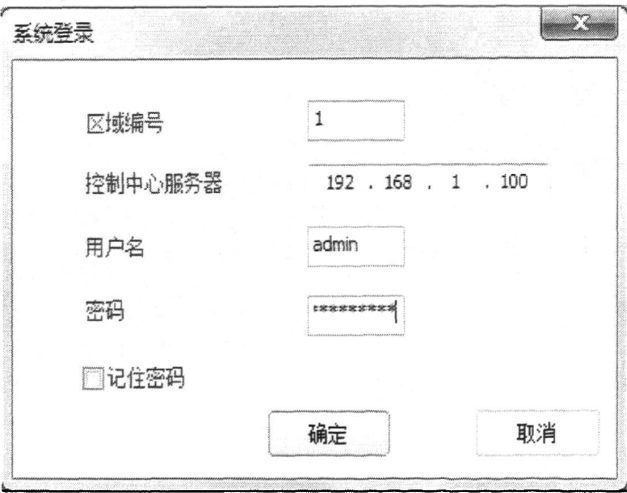


图 6.3 登陆鉴权效果图

6.2.3 参数配置模块验证

通过判定当前参数配置文件 Config.dat 是否存在来判定参数配置文件是否能稳定工作。通过分析可以知道，Config.dat 是否存在影响到后续的读写操作，在 Config.dat 文件不存在的时候要求能够提示操作员，添加配置文件；否则继续后面的操作。按照上一章对配置文件的操作，可以知道软件启动的时候读取配置文件，保存的时候写入配置文件，因此配置文件的操作是先读取再写入。按照此方式，验证配置文件效果如图 6.4 所示，

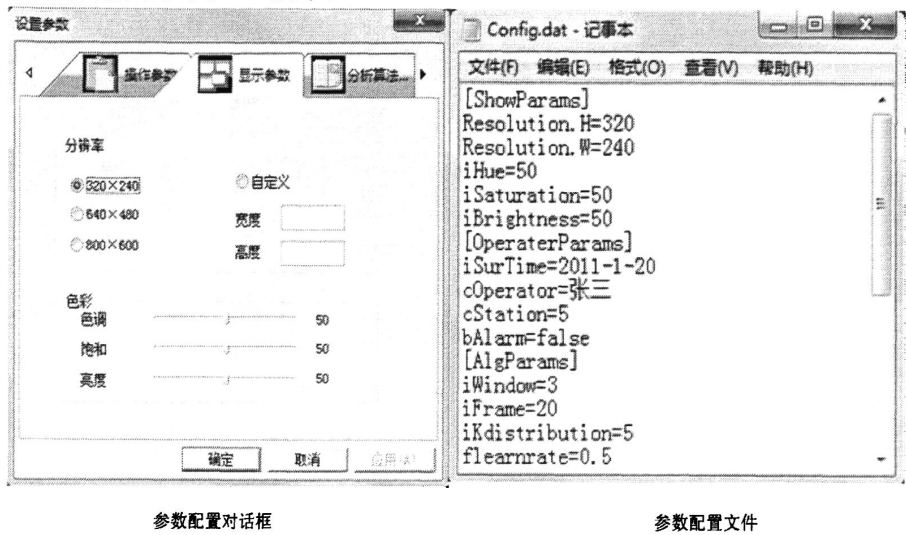


图 6.4 参数配置模块验证

6.2.4 林火识别模块的验证

本文考虑在两种情况下验证林火识别模块的准确性，①在没有出现烟火的情况下，用监控系统监控整片林区；②在出现烟火的时候利用林火识别算法进行判决。下面对比林火监控系统的监控效果：

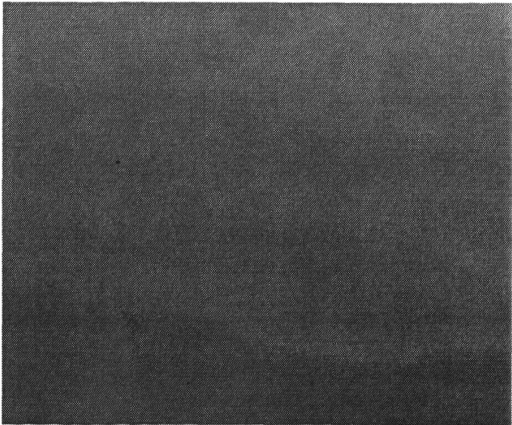


图 6.5 没有烟火时的效果

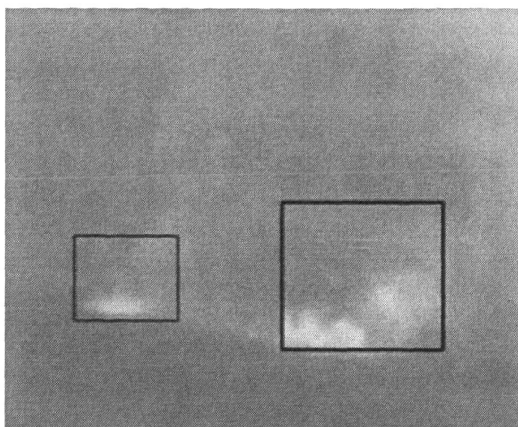


图 6.6 林火识别模块检测效果

下面对结果做以分析：在实际监控中，红外森林防火监控系统不仅能够有效的识别林火烟雾，而且能够对于干扰目标进行排除，下面对实际情况进行说明。①由于采用了红外采集设备，有效的提升了雾天中的识别准确度，能够灵敏的检测到森林环境中温度变化，并对温度持续上升迅速作出反应，在火灾发生早期探测出火险，降低了火焰蔓延时间，相应的增加了扑火救援的时间；②使用混合高斯模型建立了背景模型，动态的描述了林区的复杂背景环境。不但能有效的识别烟火，而且能够对环境中的汽车、灯光等干扰目标进行排除，降低了漏判率和误报率。

### 6.2.5 分析报告模块的验证

分析报告模块能够按照需要自动生成 RPT 文档，分析报告的测试包含两个方面：（1）报告文件的可操作性，报告文件要求是 RPT 文档，能否正常的打开、关闭以及更新文档。（2）报告内容完整性，分析报告的内容包含视频拍摄时间、地点、操作人员、视频截图、分析时间、告警信息、说明信息以及操作员希望保存的视频监控信息等。

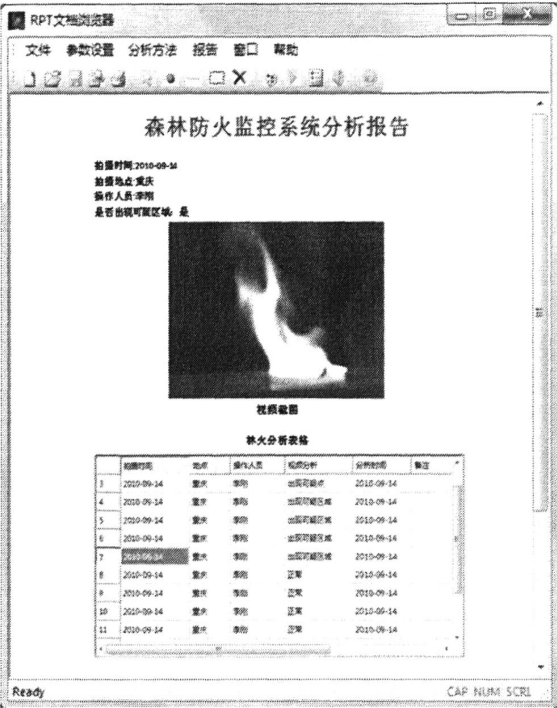


图 6.7 林火分析报告的效果图

经过测试发现分析报告能够顺利的创建、打开关闭以及更新报告文件。并且按照指定字体、排版格式生成分析报告，还能够利用多进程的方式显示分析报告。

6.2.6 数据库模块的验证

软件启动后，在监控一段时间后，选择“文件->导入数据库”就能够利用 ADO 接口将上一章中介绍的数据导入到数据库中。待 ADO 完成操作之后，通过 Access 验证数据库模块，在 Access 中可以看到数据表如图 6.8 所示。

UserInfo							
用户ID	用户名	用户性别	用户职位	优先级	系统权限	密码	备注
1	李刚	男	普通用户	5	Site_Management	*****	
2	赵明	男	普通用户	2	Query_Log	*****	
3	刘丽	女	普通用户	3	Site_Management	*****	
4	张猛	男	管理员	5	Site_Management	*****	
5	刘洋	男	普通用户	3	Site_Management	*****	
6	陈胜	男	普通用户	3	Query_Log	*****	
7	李梅	女	普通用户	3	Query_Log	*****	
8	吴平	女	管理员	5	Site_Management	*****	
9	张巡	男	普通用户	2	Query_Log	*****	
10	王铭	男	普通用户	1	Query_Log	*****	

图 6.8 数据库模块验证

经过测试发现该模块能够正常访问数据库文件，能够打开、关闭以及更新数据库文件。通过每条记录所对应的键名能够正确添加、修改、删除各条记录的数据。



6.2.7 日志模块的验证

软件运行的时候，软件的状态会发生变化。日志模块能够检测到软件运行的状态，也可以说日志模块是软件的“体温计”。本文在软件运行一段时间后查看输出日志，使用自带程序查看日志文件 System.log，查看效果如图 6.9 所示。

日志时间		告警级别	日志消息
[2010-10-08	23:09:49]	<INFO>	Initialize software ui and read config files
[2010-10-08	23:09:50]	<INFO>	Alloc memory for video module success
[2010-10-08	23:09:58]	<INFO>	Local--->server
[2010-10-08	23:10:02]	<WARM>	Computer is not in a net
[2010-10-08	23:10:14]	<INFO>	Local--->server
[2010-10-08	23:10:14]	<INFO>	Local--->server: LOGIN_CLIENT
[2010-10-08	23:10:14]	<INFO>	start certification
[2010-10-08	23:10:15]	<INFO>	server--->Local: LOGINED_CLIENT
[2010-10-08	23:10:16]	<INFO>	Local--->server:QUERY_ALLAREA_INFO
[2010-10-08	23:10:16]	<INFO>	server--->Local: ALLAREA_INFO
[2010-10-08	23:10:16]	<INFO>	Local--->server:QUERY_AREADVS_INFO
[2010-10-08	23:10:17]	<INFO>	server--->Local: AREADVS_INFO
[2010-10-08	23:10:17]	<INFO>	Local--->server:REGISTER_CLIENT
[2010-10-08	23:10:18]	<INFO>	Local--->server:QUERY_PERMISSION
[2010-10-08	23:10:18]	<INFO>	server--->Local:DVS_PERMISSION
[2010-10-08	23:36:20]	<INFO>	Local--->server:UPDATE_SITEINFO

图 6.9 日志文件 System.log 的内容

可以看到日志信息完整，能够体现软件的运行状况。通过查询相应告警级别的可以快速定位需要的信息，为今后软件的维护和更新提供了参考来源。

6.3 测试结果分析

本文中的软件功能测试使用黑盒测试，也就是说测试的时候不参考源代码，只作功能方面的测试。利用相应的测试指标判断各个模块的运行效果，下表显示了测试的统计结果。

表 6.3 功能测试结果

测试项目	功能指标	CPU 占用率	内存使用量	吞吐量测试	备注
软件界面	软件界面可靠性高、美观、易操作	8%	11,995KB	无	软件整体结构健壮、界面美观，且包含大量提示信息
鉴权	能够连接到服务器，并获得相应权限	12%	1,7323KB	无	顺利连接到服务器，并能与服务器互发消息
参数配置	能够正确读取配置文件中的配置值，并且能够随时修改参数	4%	1,648KB	无	参数存取的效率还带加强
林火识别	准确度高、误	21%	5,668KB	无	还需加入测温功能，

	报率低				提高准确度
分析报告	能够顺利生成 Word 文档，插入文字、图像和表格，并启动 Word 打开分析报告	15%	5,341KB	能满足当前数据量	Word 文档在保存为模版时或弹出 Normal.dat 对话框
数据库	能将 E-R 模型中的数据顺利导入数据库	8%	2,344KB	能满足当前数据量，但数据越多效率越低	目前还不支持存储动态图像
视频播放	能顺利解压视频并播放视频	14%	20,233KB	无	能够顺利播放从前端采集的图像并支持视频回放功能
日志	能够稳定的将软件信息输出到日志文件中，并且不存在覆盖现象	10%	1,895KB	大文件时效率偏低	日志文件过大时，打开速度慢，问题还待解决

6.4 本章小结

本章通过搭建实验测试环境，对软件系统的重要模块做了功能测试并对测试结果作了分析，并提出了改进意见，最后对系统的整体功能进行了测试，验证了软件功能满足设计要求。

## 第七章 结束语

### 7.1 全文总结

本文总体可以分成六个部分,首先,介绍了森林防火的背景,阐述了森林防火监控系统的功能。其次,深入分析了红外辐射理论、红外成像的原理以及红外焦平面探测器的发展状况。并且分析了红外技术与森林防火监控系统的交集,阐明了红外森林防火监控系统的优势。然后,介绍了林火识别算法,对比了常用的几种的林火识别算法,最终将混合高斯建模的方法与红外图像相结合,参考了前人做的工作并在实现中做了改进。接着,阐述了红外森林防火监控系统的需求,借鉴传统的监控方案,提出了红外防火监控系统的总体方案。并且详细介绍了林火监控系统的各个组成部分,描述了各个组成部分的运行流程。再次,设计了红外森林防火监控系统的软件。结合软件设计流程和面向对象方法,设计了并实现了界面模块、鉴权模块、参数配置模块、林火识别模块、分析报告模块、数据库模块、视频播放模块、日志模块。最后,搭建测试环境,并对各个软件模块做了单元测试。更进一步的对整个系统做了集成测试。软件运行稳定、烟火识别准确、执行效率高、可靠性强。

### 7.2 未来工作

随着红外技术的发展,使用红外的森林检测技术会在更多的领域得到使用。本文中的红外森林监控系统已能够满足对森林火灾监控的需求。但是还要对已有的红外森林监控系统做进一步的研究,以下是未来的工作计划:

(1) 研究效果更加优良的视频与处理方法,降低因为拍摄噪声以及传输带来的噪声。

(2) 采用更优良的林火识别算法,降低算法对系统的依赖程度。同时算法主要考虑以时空关系特征的算法。

(3) 研究红外测温算法和空间测量方法,将红外测温技术运用在林火监控中。

(4) 结合 GIS 以及 GPS 等定位方案,提供精确的火灾位置信息。并包含智能扑救指挥系统,该系统能够提出综合扑救方案。让扑火指挥者又可参考的决策方案。



## 致 谢

在论文完成之际，我要衷心地感谢我的导师黄俊副教授，本论文是在黄老师的悉心指导和关怀下完成的，在论文期间，黄老师提出了许多宝贵的意见和建议，使我受益匪浅。同时感谢黄老师给了我参与项目组研发这样一个实习和实践的机会，让我从书本中走出来，与项目组成员一起从事测试仪表的研发。这三年里，在学习和工作中，黄老师对我们悉心指导和帮助，在生活上黄老师更似朋友一样问候关怀。

同时也要感谢实验室的代少升老师、王小平老师，让我有机会加入红外成为红外项目组的一员，在我专心投入项目研发的岁月，有汗水也有喜悦，其中也多亏以上诸位老师对我学习上的悉心指导和工作生活上的问候关怀。在做开发过程中老师给予我很多指导与帮助，从他们身上，我能深深感受到严谨认真的钻研精神，也体会到很多为人做事的道理，这些让我受益良多。在做研发的过程中，遇到了很多问题很多困难，彷徨无措之时都得益于他们的鼓励和细心指导，这里我要感谢他们，因为每一次困惑的解答对我来说都是一次进步。感谢红外项目组的全体成员，至今还依旧深深记得加班调试代码的那段辛苦的日子；记得紧张的工作、持续的加班，疲惫之余大家相互鼓励带来的一次次感动；记得代码调通，所有模块均可以正常运行时候那一张张喜悦的面孔。在项目的开发过程中，大家都能相互研究探讨，彼此进步，整个实验室的老师同学，组成了一个很好的团队——相互合作相互进步的团队，短暂的岁月里，看到团队的迅速成长，在实验室度过这三年给我留下终生难忘的回忆。

感谢我的父母的支持与帮助，感谢他们为我所做出的一切。他们的支持和鼓励使我顺利完成了硕士学业。

感谢各位老师百忙中评审批阅我的论文，也真心感谢各位专家提出宝贵修改意见。



## 参考文献

- [1] 胡文奇.森林防火无线远程监控系统[D].北京邮电大学.2008.12.
- [2] 黄勤珍,谢红兵.森林防火自动监测系统[J].四川大学学报.2001.38(8):87-90.
- [3] 张建奇,方小平.红外物理[M].西安,西安电子科技大学出版社,2004.
- [4] 饶裕平,方陆明,柴红玲.林火视频监控中烟识别方法概述[J].四川林业科技.2009.1(30):81-85.
- [5] 黄承韬.省级森林火灾视频监控与指挥平台的建设[J].信息技术.2009.22(4):89-92.
- [6] 魏国良,游韦.GPS 技术在森林防火中的应用现状和前景展望[J].陕西林业科技.2009, 30(1):121-123.
- [7] 顾彤宇,顾彦方.基于 GIS 的永吉县森林防火指挥系统的设计[J].吉林林业科技.2009, 28 (2):52-53.
- [8] 许猛,李思普,张晓娟等.红外测温中环境辐射影响的分析[J].理论与实践.2007, 44(3):143-145.
- [9] 李云红等.红外热像仪精确测温技术[J].光学精密工程.2007, 59(9):267-268.
- [10] 刘缠牢,谭力勋等.红外图像伪彩色编码和处理[J].应用光学.2006, 54(5):79-82
- [11] 谷烽,姜云飞,毛明志.软件过程模型回顾与分析[J].现代计算机.2005, 87(5):28-30
- [12] 刘路放.Visual C++与面向对象程序设计教程[M].北京,高等教育出版社, 2002.
- [13] 李淑春.软件测试与测试技术[M].北京,清华大学出版社.2009.
- [14] 杨立.红外热像仪计算与误差分析[J].红外技术.1999, 21(4):20-24.
- [15] 贾涛,王耀南等.一种基于 ActiveX 技术的温度采集系统的实现[J].中南大学学报:自然科学版,2003,34(z1):241-244.
- [16] 刘群华,李春燕等.红外光电立靶数据库的设计与应用[J].光学技术,2007,32(3):471-475.
- [17] Amit M A, Lee K, Balick B. Extend surface temperature and emissivity retrieval to the mid-infrared(3-5)using the Multispectral Thermal Imager(MTI)[J].Remote Sensing of Environment.2005,98:141-151
- [18] 周金和,彭福堂.一种有选择的图像灰度化方法.计算机工程. 2006,20(32):198-200.
- [19] 吴爱国,杜春燕,李明.基于混合高斯模型与小波变换的火灾烟雾探测.仪器仪表学报.2008, 8(29):1622-1626.
- [20] 苏文杰.视频检测系统中的背景检测及更新算法研究[D].长安大学.2009.5

- [21]徐鹏,任波. 基于单高斯模型的森林火灾烟图像目标检测[J]. 计算机与现代化.2009, 30(2):18-20.
- [22]Ekin A , Tekalp A M. Shot type classification by dominant color for sports video segmentation and summarization[C].IEEE Intl conf. Acoustics ,speech, and signal processing ,HongKong ,2003,1(3):173-176.
- [23]Matsuyama T, Ohya T, Habe H. Background subtraction for non-stationary scenes[C]. In Proceedings of the 4th Asian conference on computer vision. Taipei, 2000.
- [24]陈焕钟,李榕,程剑光. 基于混合高斯模型的运动目标检测[J]. 激光杂志社.2009,4(30):32-33.
- [25]西刹子.智能网络视频监控技术详解与实践[M].北京, 清华大学出版社.2010,70-84.
- [26]Stauffer C, Grimson W.E.L. Learning Patterns of activity using real-time tracking[C].IEEE Transactions on Pattern Analysis and Machine Intelligence, 2000, 22(8):747-757.
- [27]Stauffer C, Grimson W.E.L. Adaptive Background Mixture Models for Real-time Tracking [C]. In Proc IEEE Conference on Computer Vision and Pattern Recognition, Fort Collins, Colorado, 1999, 2:246-252.
- [28]KaewTraKulPong.An Improved Adaptive Background Mixture Model for Real-time Tracking with Shadow Detection[C].In Proc 2nd European Workshop on Advanced Video Based Surveillance Systems, AVBS01.2001, 9.
- [29]姚会.基于改进的混合高斯模型的运动目标检测方法[J].厦门大学学报.2008, 47(4):505-510.
- [30]Chen Y T, Chen C S ,Huang C R, Hung Y P. Efficient hierarchical method for background subtraction[J].Pattern Recognition ,2007,40(10):2706-2715.
- [31]Zivkovic Z, Van der Heijden F.Recursive unsupervised learning of finite Mixture models[C].IEEE Trans Pattern and Mach Intell, 2004, 26(5):651-656.
- [32]刘伟,孙丽媛等.自适应中值滤波在数字图像处理中的应用[J].河北理工大学学报.2007. 29(4):111-120.
- [33]胡越黎,宣祥光等.图像的中值滤波算法及其 FPGA 实现[J].计算机测量与控制.2008.16(11):1672-1675.
- [34]朱孝政.一种新的混合高斯模型的学习算法[J].航空计算技术.2006, 4(36):121-126.



- [35]刘亚,艾海舟.一种基于背景模型的运动目标检测与跟踪算法[J].信息与控制.2002, 31(4):315-319.
- [36]Wren C R, Azarbytcjani A, Darrell T, Pentland A. Real-time tracking of the human body [J]. IEEE Trans.Pattern and Mach Intell, 1997, 19(7):780-785.
- [37]S Q Jiang, W Gao. A new method to segment playfield and its applications in match analysis in sports video[C].ACM Multimedia, New York , 2004.
- [38]任厚平,张永明等.基于混合高斯模型定位的火灾烟雾纹理特征提取[J]. 微计算机信息.2005. 21 (1):83-85.
- [39]Cemgil A T. A hybrid graphical model for robust feature extraction from video[C].In Proc of the conf on computer vision and pattern recognition, 2005.
- [40]陈辉.森林视频监控系统[J].林业资源管理.2004, 23(3):69-70.
- [41]Z G Zhu, G Y Xu, B Yang. A real-time vision system for automatic traffic monitoring [J].IEEE Trans Image and Vision computing. 2000, 30(11):781-794.
- [42]魏凯等.几种技术在视频监控中的应用[J].中国多媒体通信.2007.4:16-19.
- [43]王霓虹,滕志霞.网格技术在森林防火地理信息平台中的应用[J].东北林业大学学报.2009, 37(1):111-114.
- [44]崔波亮,王耀南.基于 GPRS 无线传输的图像监控终端实现[J].计算机测量与控制.2008, 16(12):1865-1867.
- [45]钟松峰.基于奥村模型的无线电检测站地貌检测范围分析方法.中国无线电.2005, 25(10):42-46.
- [46]郑托.农村 GSM 基站覆盖范围的测算[J].湖北邮电技术.2001,22(3):21-23.
- [47]杨世活.数字无线中继多点通信系统及组成[J].无线电通信技术.1992, 3(18):212-214.
- [48]肖吉军,陈利学.H.264 在网络视频监控系统中的应用研究[J].软件导刊.2009, 18(6):181-183.
- [49]Guptc S, Masoud O, Martin R F K. Detection and Classification of Vehicles [J]. IEEE Transaction on Intelligence Transportation Systems, 2002, 3(1):37-47.
- [50]谷烽,姜云飞,毛明志.软件过程模型回顾与分析[J].现代计算机.2005, 27(5):28-30.
- [51]李淑春.软件测试与测试技术[M].北京,清华大学出版社.2009:108-156.
- [52]梅林. Windows 环境下红外图像采集和处理系统[J].红外与激光工程.1999, 28(3):190-192.



## 附录 A 攻读学位期间科研成果及发表的论文

### A.1 参与主要科研项目及工作

1. 参与校企合作项目“红外热成像检测系统”的研发，参与软件开发工作。
2. 参与重庆市科委项目“基于红外热成像技术的森林防火监控系统”的研发。

### A.2 攻读硕士期间发表的论文

1. 《红外分析软件中 COM 技术的运用》[J].测控技术, 2010, 3, 75-78.
2. 《PC 端红外分析软件设计与实现》[J].电视技术, 2010, 3A, 103-105.



## 附录 B 引用代码段

```
#ifndef FIREDETECT_H
#define FIREDETECT_H
#endif
class _declspec(dllexport) CGaussMixModel
{
public:
    int iFilterWindow;
    int iFrame;
    int iK_distributions;
    double dLearnRate;
    float fCredible_Threshold;
    float fFGThreshold;
    GaussDistribution gaussdis;
private:
    typedef struct VideoInfo
    {
        int nSize; //VideoInfo 大小
        int nChannels; //通道
        int depth; //像素的位深度
        int width; //图像宽像素数
        int height; // 图像高像素数
        int imageSize; //图像数据大小(单位字节)
        char *imageData; //指向排列的图像数据
        int widthStep; //排列的图像行大小，以字节为单位
    } VideoInfo;

    typedef struct GaussBGMPParams
    {
        float fCredible_Threshold; // 可信与不可信阈值 T,
        float fFGThreshold; //防止前景过大阈值下限,
        int idistributions; //分布数,
```

```

    float fmean;//期望,
    float weight;//权重,
    float variance;//方差,
}GaussBGMPParams;

typedef struct BackGroundState
{
    VideoInfo* background;//背景
    VideoInfo* foreground;// 前景
    void* storememory;//存储内存
    UpdateGaussMixModel update;//更新混合高斯模型
    DeleteGaussMixModel delete;//删除混合高斯模型
} BackGroundState;

typedef struct GaussBGM
{
    BackGroundState backgstat;//背景状态
    GaussBGMPParams* params;//高斯分布参数
    int iframe;//帧数
    bool bmatch;//是否匹配
} GaussBGM;

public:
    BackGroundState& CreateGaussMixModel ( VideoInfo* newvideo,
    GaussBGMPParams* pBGMPParams ) ;//创建背景模型
    int UpdateGaussMixModel(VideoInfo* curframe,/GaussBGM*
    bgmodel,double dLearnRate); //更新高斯模型
    void DeleteGauss (GaussBGM** pGauss);//删除过时模型
};

```