

# 移动视频监控系统中 多功能显示终端子系统的设计与实现

## 摘 要

视频压缩技术的迅速发展和网络带宽的不断提高使得流媒体图像通过网络流畅地传输成为现实,具有广阔的应用前景。与此同时,移动网络技术的发展,网络传输速度的提高使得高质量图像也能通过移动网络进行传输。另外,智能手机及个人数字助理等终端处理能力的不断提高,使得使用这一类移动设备对视频流媒体进行解码成为可能。

移动视频监控系统的目标是建立一套无线网络环境下的视频监控系统,可以支持无线局域网和当前的 CDMA1X 网络,并可在移动设备上实现实时监控。本文所论述的多功能显示终端子系统是移动视频监控系统的组成部分,它包括固定显示终端和移动显示终端两种终端。其中,固定显示终端以有线方式接入系统,部署在普通 PC 设备上;移动显示终端部署在移动设备上,以无线方式接入系统。多功能显示终端通过这两种终端接入方式,实现了对监控终端的实时监控,同时可以接收监控终端发出的 GPS 定位信息,对监控终端进行实时定位。

本文首先对视频监控的相关技术进行了介绍和分析,提出了移动视频监控系统的整体解决方案;然后介绍了多功能显示终端子系统,包括需求分析、详细设计和实现等内容,其中,对系统所使用的视频缓冲机制进行了详细介绍;最后,通过对系统测试结果的分析,验证了系统功能,证明了系统的功能和性能达到了设计的要求。

本文最后对全文做出了总结,并描述了多功能显示终端子系统存在的不足和可能的进一步工作,最后总结了本人在硕士研究生期间的工作和成果。

**关键词:** 视频监控 无线 流媒体 视频缓冲

## Research and Implementation of Multifunction Display Terminal Subsystem of Mobile Video Surveillance System

### ABSTRACT

Technology of video codec and broad network is developing rapidly, because of that, transmission of high quality streaming media comes true. Meanwhile, mobile network is also developing, its speed is much higher than before. So high quality picture can be transmitted through mobile network, this makes new foreground for the application that can't use broad wired network to transmit video stream. Besides, Processors of intelligent handset and PDA are improved much than before, so it is possible to decode video stream on these equipment.

Mobile video surveillance system is based on the above. It aims to build a video surveillance system on wireless network. It can work on WLAN and CDMA1X, and realizes real time surveillance on mobile devices. The multifunction display terminal subsystem introduced in this thesis is a subsystem of mobile video surveillance system, it includes two kinds of terminals, fixed display terminal and mobile display terminal. The fixed display terminal is connected to the system by wired line and deployed on common PC. The mobile display terminal is connected to the system through wireless environment and deployed on mobile devices. By these two ways of connection, the multifunction display terminal realizes real time surveillance. It can also receive GPS information from surveillance terminals and get the real-time location of them.

The thesis firstly introduces relative technologies of video surveillance, proposed solution of mobile video surveillance system. Then introduces multifunction display terminal subsystem, include requirement analysis, design and implementation, and introduces the realization of video buffer mechanism. Finally, introduces the test result

of the system.

At last, the thesis makes a summary, and describes the shortage and the future work. And also sum up my work and achievement during my post graduate time.

**KEY WORDS:** video surveillance      wireless      Streaming Media  
video buffer

### 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： 陈琦 日期： 2008.2.28

### 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。（保密的学位论文在解密后遵守此规定）

保密论文注释：本学位论文属于保密在\_\_年解密后适用本授权书。非保密论文注释：本学位论文不属于保密范围，适用本授权书。

本人签名： 陈琦 日期： 2008.2.28

导师签名： Jing 日期： 2008.3.28



# 第一章 绪论

## 1.1 课题背景

视频监控一直是人们关注的热点之一，经历了本地模拟视频监控、本地数字视频监控、基于网络的远程数字视频监控的发展历程之后，现有的数字视频监控系统在远距离监控、系统部署，管理维护，历史记录存储管理等方面具有无可比拟的优势，以其直观、方便、信息内容丰富详实等特点被广泛应用于公安、电力、金融、交通、水利等部门。

数字多媒体监控系统代表了新一代视频监控技术的发展方向。它完全基于计算机，并以其为核心，运用最新的视频压缩技术、网络通信技术建立一套完整的监控体系，优化内部结构，提高整体性能和反映速度，满足新技术不断发展的需要，并向用户提供不同行业不同需求的数字多媒体监控管理解决方案。

然而随着需求的广泛扩展，现有的数字视频监控系统也面临着许多亟待解决的问题，首先，监视点和观察点在地理位置布置方面存在限制，监视点与观察点跨越距离太远则会造成传输性能不稳定，并且现有监控系统不支持终端的移动性；其次，现有监控系统所支持的终端接入方式单一，即只支持 ADSL 接入，或者只支持 CDMA 接入，不能同时支持多种接入方式，这样的话不利于监控业务的扩展；第三，监视模式单一，用户只能在监控中心才能得到监控画面或者实施控制；第四，现有监控系统部署时所需周期长，且不易灵活调整，不易撤销；第五，系统跨越地理区域太广则会因网络间融合问题导致信令、媒体数据传输不稳定。

视频压缩技术的迅速发展和网络带宽的不断提高使得流媒体图像通过网络流畅地传输成为现实，具有广阔的应用前景。与此同时，移动网络技术的发展，网络传输速度的提高使得高质量图像也能通过移动网络进行传输。另外，智能手机及个人数字助理等工具处理能力的不断提高，使得使用这一类移动设备对视频流媒体进行解码成为可能。

移动视频监控即是视频监控与各种移动设备上的流媒体技术相结合的产物，其最大的优势就是用户不必到专门的监控室，而是通过移动设备能随时随地查看监控画面。

本课题是研究一种多功能显示终端子系统，实现在移动显示终端以及固定显示终端上，对监视点随时随地的监视，实时获取监视点的视频信息。

## 1.2 课题研究内容

本论文基于“移动视频监控系统中多功能显示终端子系统”项目，是移动视频监控项目的一个子课题。主要研究移动视频监控系统中一种多功能显示终端，包括基于移动设备的移动显示终端和基于普通 PC 的固定显示终端。

在项目期间，本人对移动视频监控所需要的网络通信技术，视频压缩技术，流媒体技术和 GPS 定位技术进行了研究和设计工作。在项目进行过程中，我参与了显示终端子系统从系统分析、设计、编码实现到测试和系统联调的全过程，分别在移动设备 PDA 和固定有线终端上完成了系统的部署；设计了一种基于反馈的视频流媒体自适应缓冲机制，并完成学术论文一篇《一种基于反馈的视频流媒体自适应缓冲机制》，已被核心期刊《高技术通讯》录用。

## 1.3 本文的主要工作

本论文作为“移动视频监控系统中多功能显示终端子系统”项目的一个子课题，其具体工作主要包括：

1. 对多功能显示终端子系统的结构进行概要设计及详细设计；
2. 理解会话初始化协议(SIP)，并进行分析、设计及编码实现；
3. 理解实时传输协议/实时传输控制协议(RTP/RTCP)，并设计实现；
4. 寻找解决网络丢包及提高视频播放效果的方法。
5. 在基于 Windows Mobile 5.0 系统的 PDA 终端上以及基于 Windows 系统的固定有线终端上分别实现显示终端子系统，可以对监控采集终端进行控制及实时监视。

## 1.4 论文结构

本论文的结构和主要内容安排如下：

- |     |                                                             |
|-----|-------------------------------------------------------------|
| 第一章 | 绪论 介绍本论文的课题背景，课题目标和主要的研究工作等内容。                              |
| 第二章 | 简要介绍 SIP、RTP/RTCP、移动流媒体以及 GPS 定位的基本知识。                      |
| 第三章 | 介绍移动视频监控系统的总体解决方案，分析了该系统的体系结构、系统构成以及多功能显示终端子系统在整个系统中的地位与作用。 |
| 第四章 | 介绍多功能显示终端子系统的功能需求；并且对各项功能模块                                 |

间的交互进行了分析，为下一章的设计和实现奠定了基础。

- 第五章 介绍多功能显示终端子系统各部分的设计原理及实现，包括用户界面，终端管理器，信令模块，流管理器，播放器以及视频数据缓冲模块等。
- 第六章 介绍多功能显示终端子系统的测试与分析结果及系统中还需改进完善的功能。
- 第七章 对全文进行总结。

## 第二章 移动视频监控相关技术综述

### 2.1 会话初始化协议—SIP

#### 2.1.1 SIP 协议的背景和功能

SIP<sup>[1]</sup>是一种信令协议，由 IETF 在 1999 年提出。它是一种可扩展的请求/响应协议，用于两个通信端点之间的信令交互。它的主要目的是为了解决 IP 网中的信令控制，对于通信业有着重要的意义。

SIP 用于控制多媒体会话的建立和终结，可以动态调整会话属性，如媒体类型、编解码格式等。SIP 相对于目前已经存在一些信令协议有什么优势呢？具体来说，SIP 具有以下明显优点：

灵活性：SIP 是基于文本的一种协议，容易扩展。

高效性：小型协议，效率高。

稳定性：SIP 协议使用了多年，已经达到了比较稳定的状态。

安全性：它提供了加密和身份验证等一些功能。其扩展还提供其他类型的安全性功能。

标准化：SIP 已经成为一种标准，在通信行业内得到了广泛的采用。

SIP 可以提供的功能包括：

名字翻译和用户定位：将描述信息映射到定位信息。

特征协商：它允许各呼叫参与者在支持的特征上达成一致。

呼叫参与者管理：呼叫参与者能够将其它用户引入呼叫或取消到其它用户的连接。此外，用户可以被转移或置为呼叫保持。

呼叫特征改变：用户可以改变呼叫特征。例如，在声音呼叫过程中，用户可以根据需要开启视频功能<sup>[2]</sup>。

#### 2.1.2 SIP 网络组件

SIP 会话主要使用四个网络组件：用户代理、代理服务器，注册服务器和重定向服务器。

用户代理 (UA) 一般部署在终端用户设备上，用于创建和管理 SIP 会话。像移动电话、多媒体手持设备、PC 等一般均为用户代理。用户代理客户机(UAC)用于发出请求消息。用户代理服务器(UAS)用于对消息的响应。

代理服务器用于会话请求的转发，当代理服务器接受到 UA 的会话请求时，向 SIP 注册服务器查询接收方 UA 的地址信息。当接收方 UA 与发送方 UA 位于同一域中时，代理服务器将会话邀请信息直接发送给接收方 UA；当接收方 UA 与发送方 UA 位于不同域中时，代理服务器将会话邀请信息转发给另外的代理服务器。

SIP 注册服务器中存有域中所有用户代理的位置。在 SIP 通信中，UA 或者代理服务器会向注册服务器检索参与方的相关信息，并根据这些信息，做出相应的操作。

SIP 重定向服务器用于参与方的定位，用于将 SIP 会话邀请信息定向到外部域。在硬件部署上，重定向服务器，注册服务器和代理服务器可以部署在同一个机器上。

SIP 系统网络结构如图 2-1 所示：

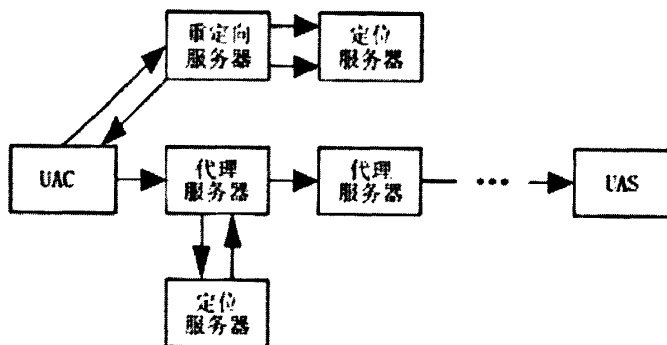


图 2-1 SIP 系统网络结构

### 2.1.3 SIP 消息的组成

SIP 消息可以分为两种类型：请求和响应。请求是从客户机发送到服务器，而响应是从服务器发送到客户机。

SIP 请求消息和响应消息均包含三个元素，其中请求消息包括请求行、头、消息体；响应消息包括状态行、头、消息体。请求行和头域定义了呼叫的本质，消息体则独立于 SIP 协议。SIP 定义了下述方法：

INVITE——邀请用户加入呼叫。

BYE——终止用户之间的呼叫。

OPTIONS——请求关于服务器能力的信息。

ACK——对 INVITE 响应的确认。

REGISTER——向服务器发送注册信息，让服务器知道用户的位置。

INFO——用于获取会话中相关信息的信令。

SIP 的响应消息主要包括六种类型, 分别为 1xx, 2xx, 3xx, 4xx, 5xx 和 6xx。每种类型都具有其特殊的意义。其中, 1xx 为 Information(信息), 代表用户代理或用户代理服务器的状态; 2xx 为 Successful(成功), 代表请求被接收; 3xx 为 Redirection(重定向), 用于向用户代理提供一个新位置或新的满足呼叫的服务器; 4xx 为 Request Failure(请求失败), 当用户请求无效时, 服务器会发送 4xx 消息; 5xx 为 Server Failure(服务器失败), 当错误为服务器本身的错误时, 服务器会发出 5xx 消息; 6xx 为 Global Failure(全局错误), 主要用于对特定用户的最终消息, 表示请求在任何位置都会失败。

## 2.2 RTP/RTCP 简介

### 2.2.1 实时传输协议 RTP

RTP<sup>[3]</sup> (Real-time Transport Protocol) 是一种实时传输协议, 主要用于传输多媒体数据流, 其可以对媒体数据流提供时间信息, 通过这些时间信息实现数据流的同步。RTP 本身并不能实现数据包的顺序传送, 而且也无法提供可靠的传送机制, 流量控制或者拥塞控制, 它是依靠 RTCP 来提供这些服务。

数据的实时到达是流媒体传输的关键, 只有实现了数据的实时到达才可以保证媒体的播放和回放。RTP 协议则提供了时间戳, 数据包序列号等结构用于控制实时数据的播放。在流媒体数据传输中, “时间戳”是非常重要的信息。发送端在发送数据时, 对数据包设置时间戳。在接受端收到数据包后, 就依照时间戳的标识, 恢复成原始的媒体数据。但 RTP 并不负责媒体数据的同步, 他只是传输层协议。同步是由属于应用层的协议来完成的。RTP 不具备传输层协议的完整功能, 没有任何机制来保证数据的实时传输, 也不支持资源预留以及服务质量的保证。RTP 协议与 RTCP 协议使用相邻的不同端口, 可以提高协议的灵活性, 使得处理相对简单。

RTP 属于传输层协议, 但在 OSI 体系结构中并没有单独的一层来对其进行实现。它通常只是根据一个具体的应用来提供服务。RTP 提供了协议框架, 上层程序可以根据具体的应用要求对其进行扩展。

应用程序运行于 RTP 之上, 而 RTP 运行在 UDP 的上层, 如图 2-2 所示。由应用程序产生的音视频数据被封装在 RTP 数据包中, 每个 RTP 数据包被封装在 UDP 消息段中, 然后再以 IP 包的形式进一步封装。

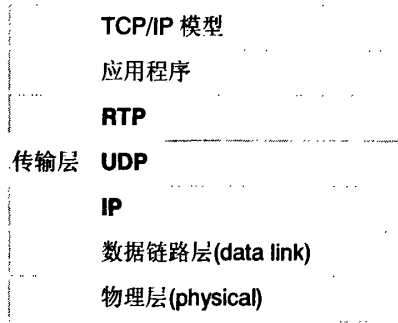


图 2-2 网络协议模型

### 2.2.2 实时传输控制协议 RTCP

RTCP<sup>[4]</sup>是实时传输控制协议，它与 RTP 协议一起，为媒体数据传输提供流量控制和拥塞控制服务。当一个 RTP 会话开始时，会分配两个相邻的端口：一个分配给 RTP，一个给 RTCP。在 RTP 会话期间，各参与者周期性地发送 RTCP 包。在 RTCP 数据包中，含有各种统计数据，服务器可以利用这些统计信息，对传输速率，甚至有效载荷类型等进行动态的改变。RTP 和 RTCP 配合使用，它们能以有效的反馈提高传输的效率，因此，这两个协议非常适合实时数据在网络上的传输。

RTCP 的功能主要包括四个方面：

- (1) 用反馈信息的方法来提高数据的传送质量，可以使用这种反馈来进行流量的拥塞控制，或者用来监视网络中出现的问题；
- (2) 为 RTP 源提供一个永久性的传送层标志；
- (3) 根据与会者的数量来调整 RTCP 包的发送率；
- (4) 传送会话控制信息。

## 2.3 移动流媒体应用

随着网络带宽及视频压缩技术的发展，移动流媒体的应用越来越广泛，到目前为止，移动流媒体应用主要经历了两个阶段<sup>[5]</sup>。第一个阶段由于各种技术的限制，采用声音加图片的模式。最初的无线带宽只有 12kb/s，推出的移动流媒体应用主要是天气预报、图片新闻等。在保证声音传输质量的前提下，采用几秒一帧的视频。在当时的技术情况下，声音只需 8kb/s 就可以达到很好的传输效果，而视频则占用剩余的带宽。像天气预报之类应用主要的信息传输目的是音频传输，视频只是用来提供辅助信息。而各种新闻等也同样如此。这类应用被称为获知性应用，获取信息是用户的主要目的，相对而言，娱乐的内容较少。

目前正在经历的是第二个阶段，无线网络的带宽达到了 20-60Kb/s。这种带

宽类似于互联网发展初期的拨号上网带宽，但由于编解码技术的发展，当初无法使用拨号上网来提供的多媒体服务在当前移动网络上则可以实现。如今，流媒体的应用种类很多，从内容上分，可以分为获知性应用和娱乐应用；而从时效上分，则可以分为直播、点播和下载等。

单就获知性应用来讲，第二个阶段和第一阶段的也并不相同，第二个阶段真正实现了流媒体服务，像上面举例的天气预报以及新闻节目，在声音正常的前提下，视频一般可以达到 3-10 帧/秒。同时比较受欢迎的应用还有视频监控。在道路拥堵的城市里，司机如果能通过手机提前获取到前方路段的交通情况，就可以避开拥挤路段，从而有效地减少阻塞的发生；上班族如果能通过手机监控家里的孩子和家庭财产的状况，则可以更加安心地专注与工作了。娱乐方面的应用则包括赛事直播、电视电影预告等。

移动流媒体的应用目前正处在第二阶段的发展之中，无线网络主要使用移动 2.5G，包括：GPRS、CDMA1X 网络，终端则需要支持数据业务的高档手机，因此，流媒体应用的普及还需要一段时间。但越来越多手机开始支持流媒体应用，而 3G 时代的到来，则可以进一步推动移动流媒体的发展。

## 2.4 GPS 卫星定位系统

GPS，即全球卫星定位系统，近年来得到了非常广泛的应用。随着其应用场景的不断增多，GPS 将会迅速渗透到人们的日常生活中来。

GPS 系统主要由空间部分、控制部分和用户部分这三大组成部分<sup>[6]</sup>。

空间部分是指 GPS 人造卫星。多颗导航卫星在卫星轨道面上不平均地分布着，而这些导航卫星就是动态的已知点，用户端通过计算其发送的“星历”，可以得到相应的导航定位信息。GPS 星历，实际上是一系列实时状态参数，这些参数被用来描述 GPS 卫星运动及轨道。民用 GPS 模块所接收到的广播星历是用于实时数据处理的预报星历，它是由 GPS 卫星通过导航电文直接向用户播发的。

地面控制部分是整个 GPS 系统最关键的部分。分布在地面的各个监控站负责提供所有的 GPS 卫星所播发的星历。GPS 信号的监测、数据的收集、导航电文的计算和注入、状态诊断以及轨道修正等均由地面系统负责。GPS 系统精确运转就是得益于地面监控系统的大量数据处理。

用户部分是指用户使用的设备上所携带的 GPS 定位模块，它负责接收、解调卫星的广播信号。GPS 模块属于被动定位，并不播发信号。通过计算与每个卫星的伪距离，求出自己的经度、纬度、高度和时间修正量这四个参数，从而计算出自己的位置。GPS 模块将得到的定位信息及辅助信息输出，供用户进行应用。



### 第三章 移动视频监控系統解决方案

#### 3.1 系統整体结构

##### 3.1.1 系統结构介绍

移动视频监控系統由监控服务器、监控终端以及多功能显示终端组成，监控服务器以固定 IP 方式接入系統；监控终端可以无线局域网、CDMA1X 网络或有線网络等方式接入到系統中；而对于显示终端，則可以有線方式或无线局域网方式接入到系統中。系統结构如图 3-1 所示。

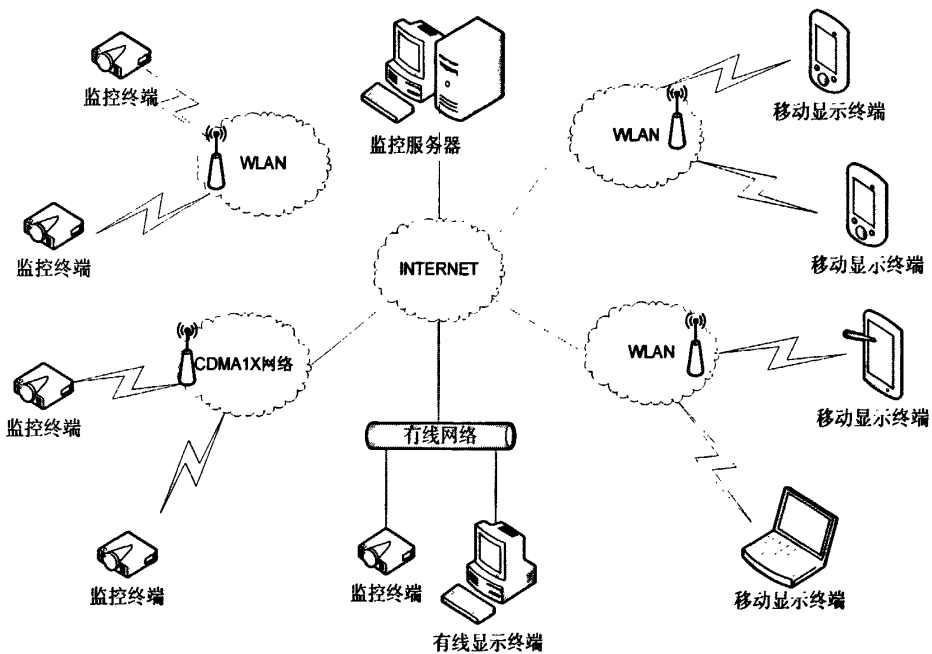


图 3-1 系統结构图

系統各部分功能如下：

表 3-1 系统各部分功能描述

系统组成部分	功能描述
监控服务器	对多个监控终端进行控制，并可接收监控终端发送过来的视频数据，呈现给用户。
监控终端	负责视频数据的采集。接收监控服务器或显示终端的控制命令，并做出响应。支持多种接入方式。
多功能显示终端	通过监控服务器与监控终端建立连接，并可对监控终端进行控制；接收监控终端发送的视频数据，呈现给用户。 显示终端可分为移动显示终端和固定显示终端。

### 3.1.2 系统结构划分

移动视频监控系统大致可以划分为三个子系统：

- 监控终端子系统：该系统主体是带有摄像功能的工控机，既可以通过 CDMA1X、WLAN 等无线方式也可以通过 Ethernet 等有线方式接入网络，用于监视某个对象或场所。
- 监控服务器子系统：该系统由服务器和数据库组成，服务器用于提供服务，数据库则用于存储、查询监控终端和显示终端的信息。服务器子系统使用固定的 IP 接入到 Internet，其主要功能是接受监控终端和显示终端的注册，控制监控终端并实时接收监控终端传送过来的监视画面。
- 多功能显示终端子系统：该系统包括固定显示终端和移动显示终端两部分。固定显示终端是以有线方式(如 Ethernet 等)接入网络的 PC 机；移动显示终端是以 WLAN 等无线方式接入网络的手提电脑、PDA 或智能手机等。显示终端子系统用于控制监控终端并实时接收从监控终端传过来的监视画面。

## 3.2 各子系统分析

### 3.2.1 监控终端

监控终端子系统的功能描述如图 3-2 所示，监控终端通过摄像头驱动模块从摄像头捕获视频信息，然后将此信息传给编码器模块进行相应格式(H.263、H.264)的编码，接着将已经编码的数据交给流控制器模块在网络上传输。GPS 接收器将接收到的位置信息传递给信令模块，信令模块将其发送出去。此外信令模块负责注册、建立连接、拆除连接、调整参数等信令操作，信令采用 SIP 协议在网络上传输。主控模块则负责将以上模块有机地整合起来并对它们进行控制。

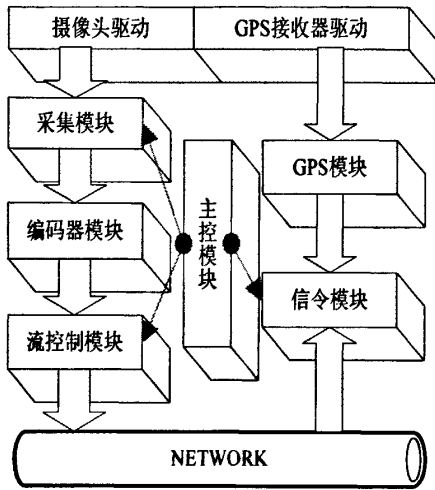


图 3-2 监控终端功能描述图

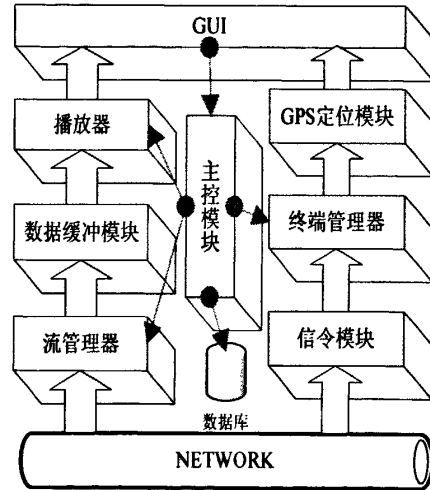


图 3-3 监控服务器功能描述图

### 3.2.2 监控服务器

监控服务器子系统的功能描述如图 3-3 所示，其信令模块除了具有监控终端子系统信令模块的全部功能外还具有为监控终端和显示终端提供注册的功能。此外，服务器子系统的另一个功能是接收视频画面，对比图 3-2 可以看出服务器接收画面的数据流与监控终端传送画面的数据流是相反的：服务器首先由流管理器模块将网络上的视频数据读取出来，然后交给数据缓冲模块进行缓冲，最后由播放器将视频画面在用户界面上呈现出来。另外，终端管理器模块负责信令的交互，并可以通过信令模块获取到监控终端的 GPS 信息，并将其在界面上呈现出来。服务器子系统的主控模块与监控终端子系统的主控模块功能基本相同，不同的是服务器子系统的主控模块在注册过程中增加了对数据库的控制。

### 3.2.3 多功能显示终端

多功能显示终端子系统的功能描述如图 3-4 所示，其模块构成与监控服务器子系统构成基本相同，但其并不具备接收监控终端注册的功能，而是在向监控服务器端注册时，获取监控终端的相关信息；另外，因为显示终端存储空间限制等原因，在显示终端并没有加入数据库存储功能。其数据流程为首先由流管理器将网络上视频数据读取出来，通过数据缓冲模块对视频数据进行缓冲，以平滑抖动、时延等影响，然后交由播放器进行解码播放，最后将视频画面在用户界面上呈现出来。

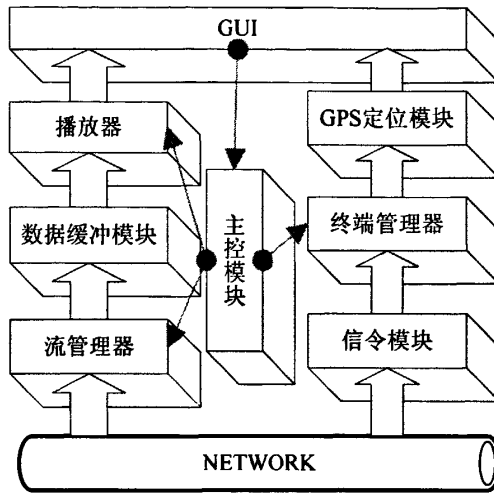


图 3-4 多功能显示终端功能描述图

## 第四章 多功能显示终端子系统功能需求

### 4.1 系统业务逻辑模型

#### 4.1.1 系统角色

在系统运行时，多功能显示终端首先向监控服务器注册并获取到其对应监控终端的信息，之后其与监控终端之间建立连接，进行实时监视，并可对监视过程中的各种参数进行修改。其与监控服务器及监控终端之间的用例关系如下：

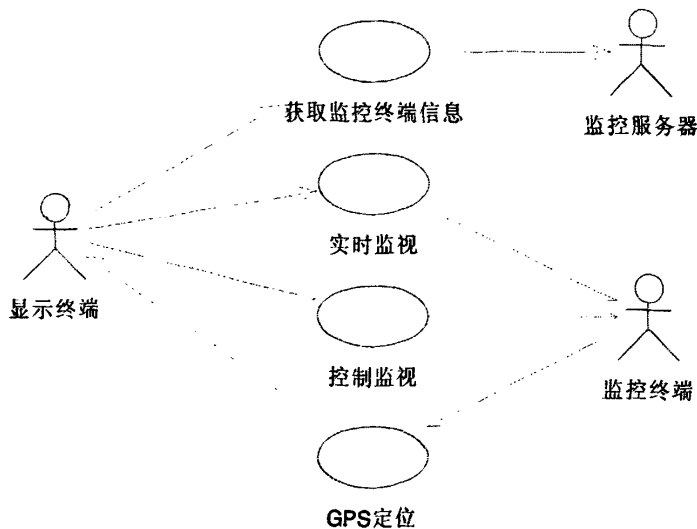


图 4-1 系统角色用例图

各用例说明如下：

表 4-1 获取监控终端信息用例说明

用例名称：获取监控终端信息
参与者：显示终端，监控服务器
简要说明：显示终端通过与监控服务器的信息交互，获取可用监控终端的信息
前置条件：显示终端与监控服务器间信令连接建立
基本事件流： 1. 显示终端向监控服务器发送注册消息 2. 监控服务器回复响应消息，其中包含显示终端对应的监控终端的信息 3. 用例终止

<b>异常事件流:</b> 1. 注册失败, 自动尝试 3 次连接, 若仍不成功, 提示注册失败 2. 显示终端返回到注册界面
<b>后置条件:</b> 监控终端信息在显示终端显示

表 4-2 实时监视用例说明

<b>用例名称:</b> 实时监视
<b>参与者:</b> 显示终端, 监控终端
<b>简要说明:</b> 显示终端与监控终端建立连接, 传递数据, 进行实时监视
<b>前置条件:</b> 显示终端获取到监控终端的信息
<b>基本事件流:</b> 1. 显示终端向监控终端发送请求消息 (INVITE) 2. 监控终端回复响应消息 (200OK) 给显示终端 3. 显示终端回复确认消息 (ACK) 给监控终端 4. 监控终端发送实时数据给显示终端, 进行实时监视 5. 用例终止
<b>异常事件流:</b> 1. 未收到回复消息, 自动尝试重新发送 3 次, 若仍不成功, 提示请求失败 2. 显示终端返回到请求界面
<b>后置条件:</b> 对监视对象进行实时监视

表 4-3 控制监视用例说明

<b>用例名称:</b> 控制监视
<b>参与者:</b> 显示终端, 监控终端
<b>简要说明:</b> 在监视过程中, 对监控终端的参数进行调节与控制
<b>前置条件:</b> 显示终端与监控终端建立连接, 进行实时监视
<b>基本事件流:</b> 1. 显示终端向监控终端发送修改参数的请求消息 2. 监控终端回复相应消息给显示终端 3. 显示终端回复确认消息给监控终端 4. 监控终端根据消息对相应参数进行修改 5. 用例终止
<b>异常事件流:</b> 1. 未收到回复消息, 自动尝试重新发送 3 次, 若仍不成功, 提示请求失败

2. 显示终端返回到参数修改请求页面
后置条件: 监控终端参数被修改

表 4-4 GPS 定位用例说明

用例名称: GPS 定位
参与者: 显示终端, 监控终端
简要说明: 在监视过程中, 动态显示监控终端的地理位置
前置条件: 显示终端与监控终端建立连接, 进行实时监视
基本事件流: <ol style="list-style-type: none"> <li>1. 监控终端向显示终端发送地理位置信息</li> <li>2. 显示终端接收到监控终端的地理位置后, 在地图上显示出来。</li> <li>3. 用例终止</li> </ol>
异常事件流: <ol style="list-style-type: none"> <li>1. 地理位置信息发送失败, 监控终端固定间隔向显示终端发送位置信息。</li> <li>2. 显示终端未收到位置信息, 则不会在地图上进行显示。</li> </ol>
后置条件: 监控终端参数被修改

#### 4.1.2 系统业务流程

显示终端通过与监控服务器以及监控终端之间的信令交互, 达到对监控终端进行控制的目的, 并通过多种信令的交互, 对监控终端进行不同的操作, 用来控制监视图像的发送以及修改监控终端的各项参数。主要的业务流程包括注册, 建立数据连接, 参数修改和停止监控等。以下分别对这几个业务流程进行了描述。

##### 4.1.2.1 注册

在显示终端启动后, 如果要获取监控终端的信息, 需要先向监控服务器发起注册(Register), 在监控服务器向显示终端发送的响应消息(200 OK)中, 会携带与该显示终端相对应的监控终端的信息, 包括 IP 地址, 端口号等。而要达到这一目的, 需要监控终端在显示终端之前就已经在监控服务器端注册。如果显示终端注册时, 并没有与之对应的监控终端注册的监控服务器, 则监控服务器向显示终端发送的响应消息中就不会携带任何监控终端的信息, 因而显示终端也就无法得到监控图像。该业务的流程图如图 4-2 所示。

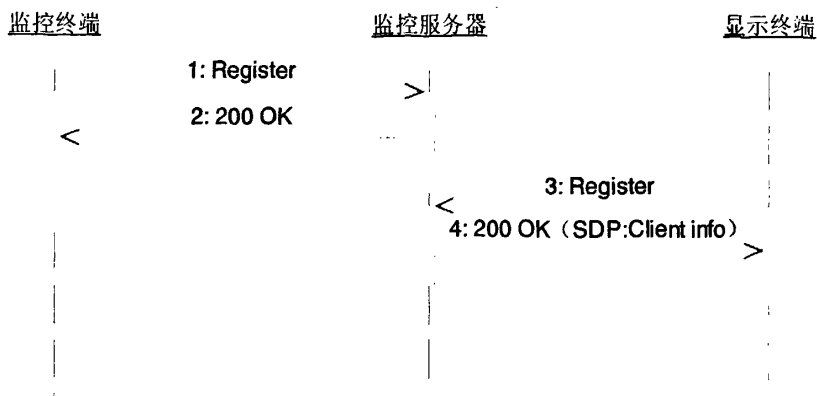


图 4-2 注册业务流程图

#### 4.1.2.2 建立数据连接

监控终端向监控服务器注册后，服务器记录下监控终端相关信息。显示终端向监控服务器注册时，监控服务器记录下该显示终端的相关信息，同时回复 200OK 消息，其中携带了与显示终端相对应的监控终端的信息。显示终端选定某一监控终端后，根据从监控服务器得到的信息，向该监控终端发送 INVITE 请求，之后通过 200OK 携带的 SDP<sup>[7]</sup>消息进行参数协商，显示终端回复 ACK 之后，协商过程完成，建立连接，进行数据传输。业务流程如图 4-3 所示。

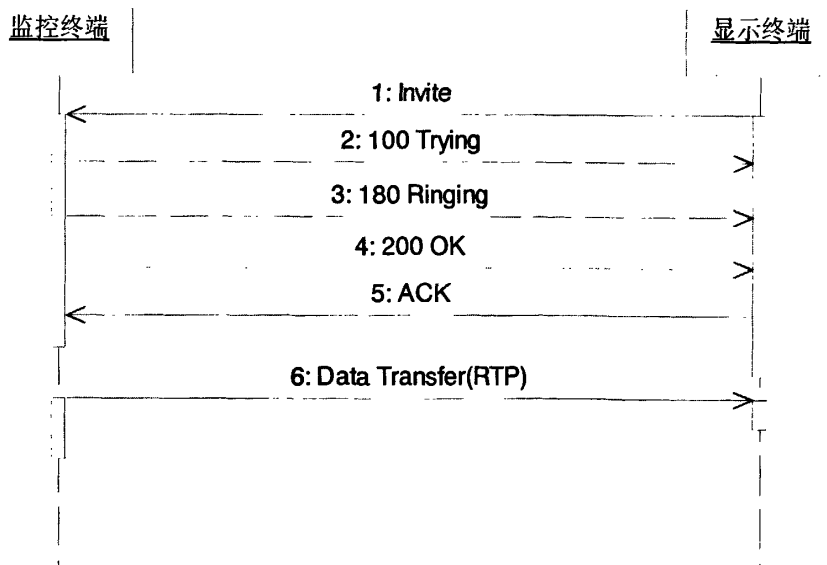


图 4-3 建立数据连接流程图



### 4.1.2.3 GPS 定位

GPS 信息由监控终端的 GPS 接收器来获取到自己的地理位置信息，然后通过 SIP 中的 Notify 信令发送到显示终端，显示终端接收到该信令后，从中提取出监控终端的位置信息并写入位置文件中，即可在地图上显示出来。

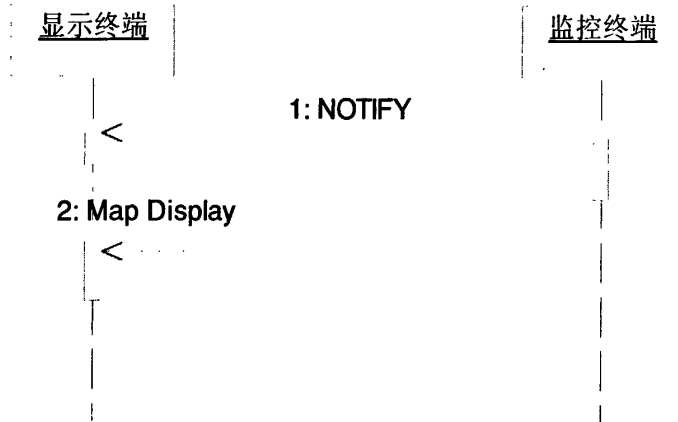


图 4-4 GPS 定位流程图

### 4.1.2.4 参数修改

显示终端要对某一监控终端的参数进行修改时，发送 INVITE(Re\_Invite)信令给监控终端，其携带的 SDP 消息中，含有所要修改的参数的信息。完成 200OK 及 ACK 的交互过程之后，监控终端对收到的参数进行分析，并做出相应的修改。之后继续进行数据的传输。这时的信令是不经过监控服务器的。

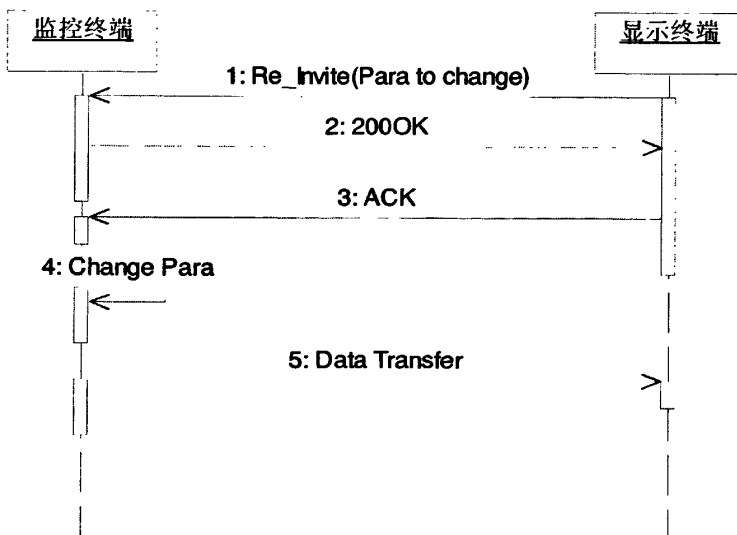


图 4-5 参数修改流程图

#### 4.1.2.5 停止监控

显示终端要停止监控时，发送 Bye 消息给对应的监控终端，监控终端收到消息后，首先停止数据的采集和发送，然后回复 200OK 给显示终端。显示终端收到 200OK 后，释放与该监控终端对应的资源，并做出相应的其他处理。业务流程如图 4-6 所示。

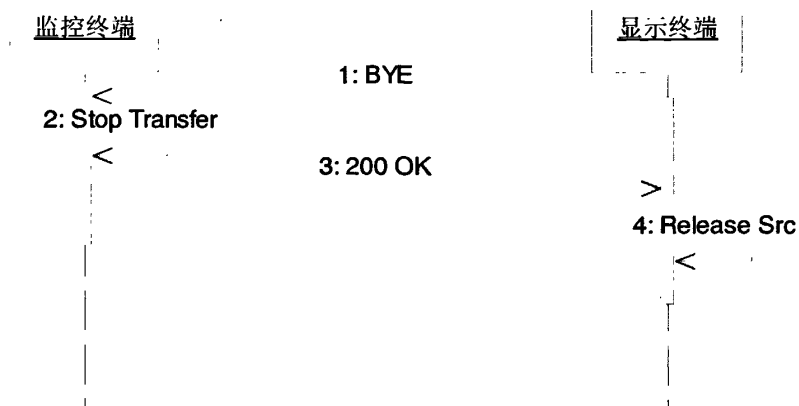


图 4-6 停止监控流程图

## 4.2 系统功能用例模型

### 4.2.1 系统功能角色

显示终端的功能角色可划分为主控模块(MainControl), 流管理模块(StreamHandler), 终端管理模块(Dogmanager)和播放模块(Player)。其中流管理模块负责对以 RTP 传输的数据流的管理; 终端管理模块负责 SIP 信令的发送与接收, 对监控终端进行管理, 并对收到的信令做出响应; 播放模块负责对接收到的数据进行解码播放(Decoder); 主控模块负责对以上各个模块的调度。模块功能角色关系如下:

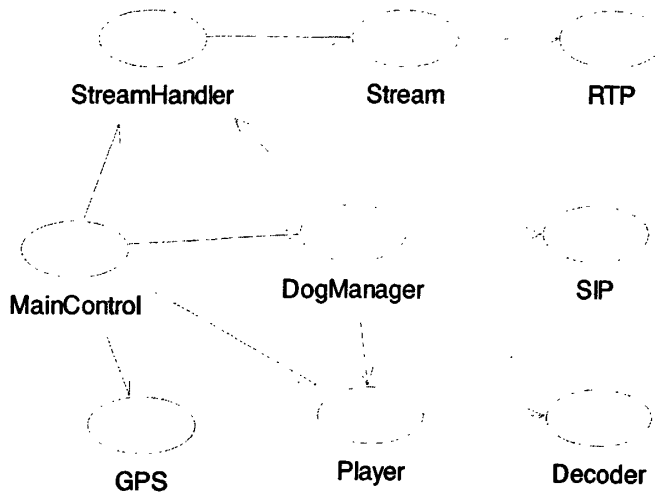


图 4-7 系统功能角色图

表 4-5 功能角色用例说明

角色名称	作用与功能
MainControl	主控模块, 负责各模块的调度
DogManager	监控终端管理器, 对监控终端进行相应的管理
StreamHandler	媒体流管理器, 对媒体数据流进行管理
Player	媒体数据播放器, 对接收到的数据进行播放
SIP	信令模块, 对收到的信令进行分析, 并执行相应回调函数
RTP	数据传输模块, 附属于流管理器
Decoder	解码模块, 用于视频数据的解码。
GPS	GPS 定位模块, 用于地理位置定位。

## 4.2.2 系统功能模块关系

### 4.2.2.1 注册

在显示终端向监控服务器发起注册时,首先由主控模块向终端管理模块发出指令,要求发送注册消息。终端管理模块接收到此消息后,通知信令模块,并将监控服务器的相关信息,如地址等,传递给信令模块,信令模块根据这些信息,向监控服务器发送注册消息(Register),从而通知服务器该显示终端的加入。监控服务器接收到显示终端的注册信息后,从中提取出显示终端的相关信息并记录,在数据库中查找与该显示终端对应的监控终端的信息,如果查找到已经注册的并与该显示终端对应的监控终端,则将这些监控终端的信息嵌入回复给显示终端的200OK 信令中,并发送给显示终端。显示终端的信令模块接收到200OK 并提取出相关监控终端的信息,提交给终端管理模块,终端管理模块记录相关信息,生成相应的终端对象,并将监控终端信息传递给主控模块,以进行界面显示等相关的其他处理。注册过程的功能模块关系如图4-8所示:

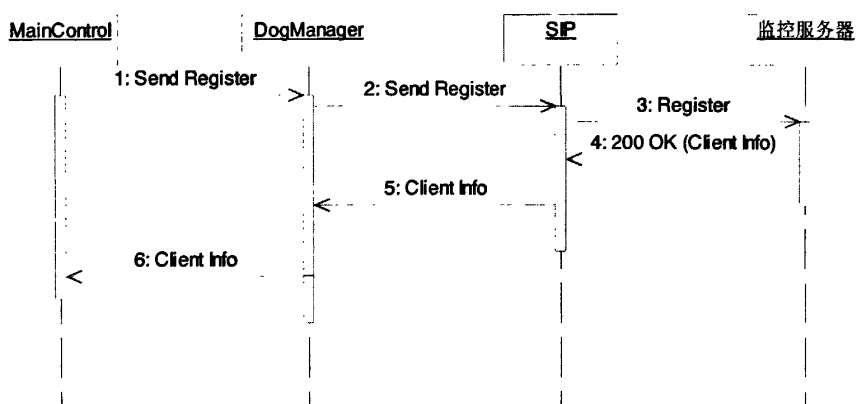


图 4-8 注册过程功能模块关系

### 4.2.2.2 发送监视请求

显示终端是通过 Invite 信令来向监控终端发送监视请求的。首先由主控模块通知终端管理模块,向某一监控终端发送监视请求,终端管理模块之后调用信令模块的接口,向该监控终端发送监视请求,即发送 Invite 信令。再经过 100 trying, 180 ringing 等信令的交互后,如果连接成功,信令模块会收到监控终端发送的200OK 消息,此消息中携带了用于协商媒体参数的 SDP 信息。信令模块将 SDP 消息交给终端管理模块进行处理,终端管理模块分析 SDP 消息,同时通知主控模块接收到了200OK 消息,主控通知流管理模块准备接收数据,并要求终端管

理模块发送 ACK 给监控终端。终端管理模块将 SDP 消息与本地相关媒体信息进行比较修改后，放入 ACK 消息中，通过信令模块将 ACK 消息发送给监控终端。监控终端再对其中的 SDP 进行分析并做出相应的设置。之后便开始进行数据的传输。以上即为发送监视请求时的功能模块关系及消息交互流程，如图 4-9 所示：

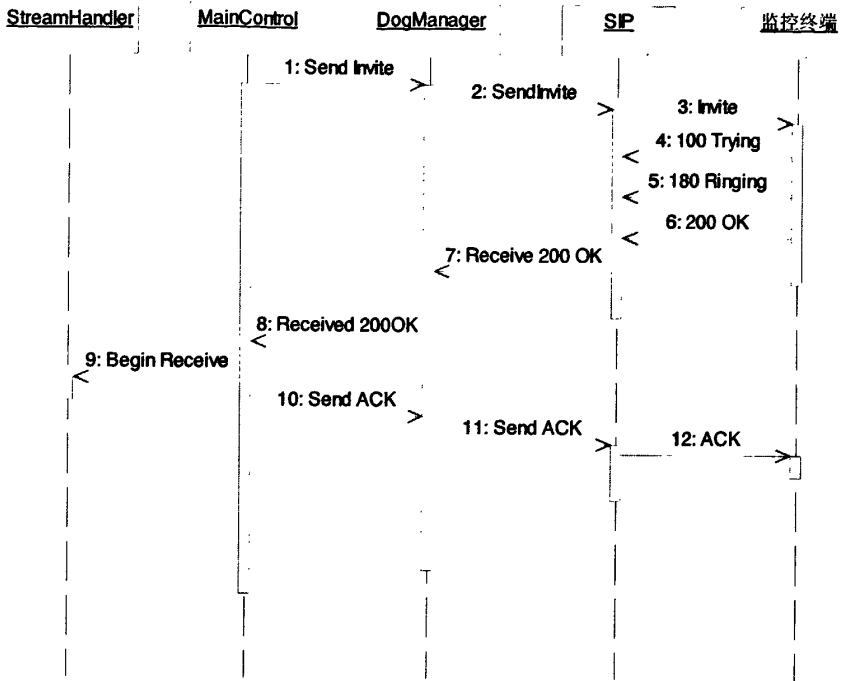


图 4-9 发送监视请求模块关系图

#### 4.2.2.3 视频接收播放

在发送完监视请求，两端参数协商一致之后，监控终端即开始向显示终端发送视频数据。视频数据首先发送到 RTP 所占用的端口，RTP 模块在接收到数据后，会以回调函数的方式，通知播放模块，并将视频数据传送给播放模块。播放模块接收到数据后，调用解码模块的解码功能，对视频数据进行解码，并将解码后的数据回传给播放模块，随后，播放模块将解码后的数据播放出来，呈现给用户。整个过程如图 4-10 所示：

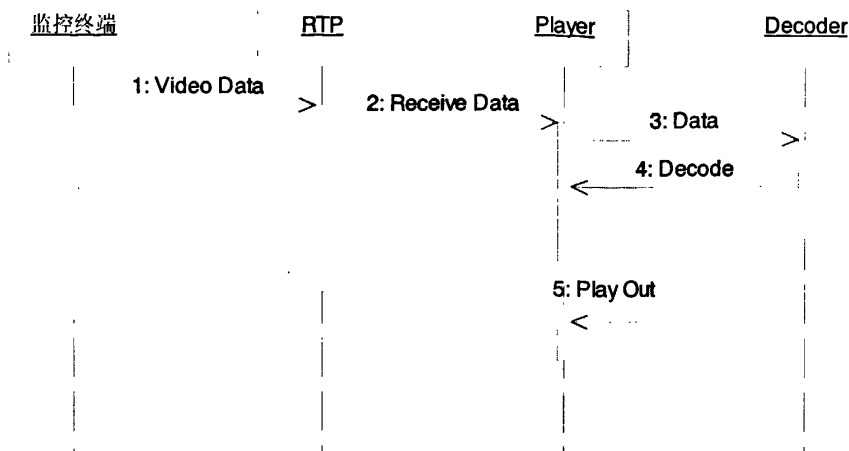


图 4-10 数据接收播放功能模块关系图

#### 4.2.2.4 GPS 定位

GPS 定位信息是通过 SIP 信令 Notify 来传输的。在监控终端获取到 GPS 信息后，封装在 Notify 信息中送到显示终端的信令模块，GPS 信息从信令模块依次上传给终端管理模块和主控模块，主控模块将其写入地图显示文件中，从而显示出来。模块间关系如图 4-11 所示：

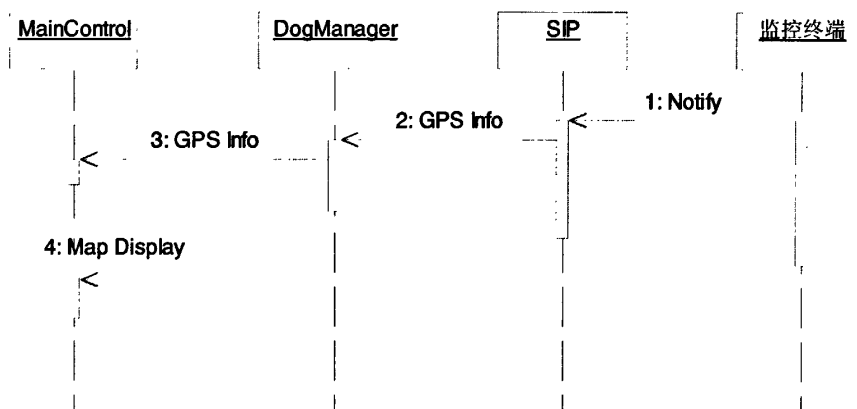


图 4-11 GPS 定位功能模块关系图

#### 4.2.2.5 修改参数

在进行参数修改时，首先由主控模块通知终端管理模块，终端管理模块之后调用信令模块，向监控终端发送对话内 Invite，将要修改的参数放在其携带的 SDP 消息中。监控终端接收到 Invite 信令后，分析其携带的 SDP 消息中的各项参数，并对需要修改的参数进行修改，之后回复 200OK 给显示终端。显示终端的信令

模块接收到 200OK 后，提交给终端管理模块，终端管理模块通知主控模块做出相应处理。最后通过信令模块回复 ACK 消息给监控终端。过程如图 4-12 所示：

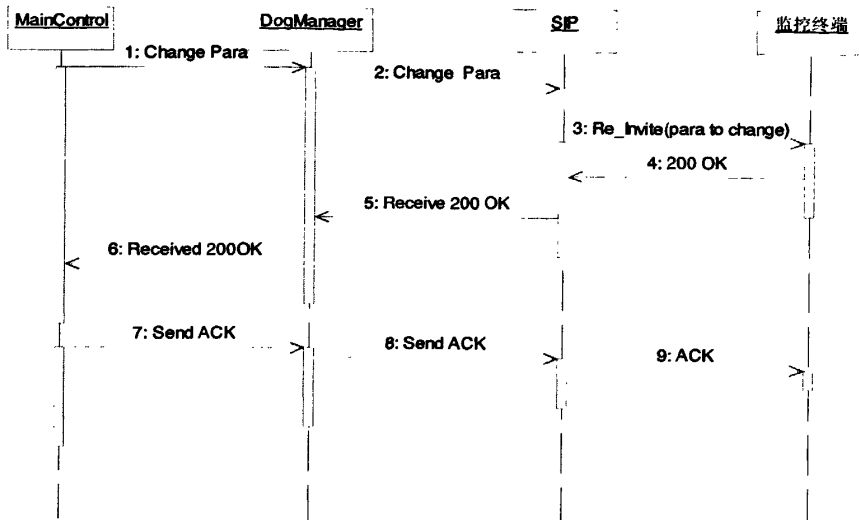


图 4-12 参数修改功能模块关系图

#### 4.2.2.6 停止监视

显示终端要停止监视时，由主控模块向终端管理模块发出停止监视的消息，终端管理模块调用信令模块向监控终端发送 BYE 信令。监控终端接收到 BYE 信令后，停止数据的采集和发送，并回复 200OK 给显示终端。显示终端的信令模块接收到 200OK 后通知终端管理模块，之后终端管理模块会通知主控模块。主控模块收到此消息后，会通知流管理模块停止接收数据，流管理模块接收到命令后，停止接收，并释放相关资源。交互过程如图 4-13 所示：

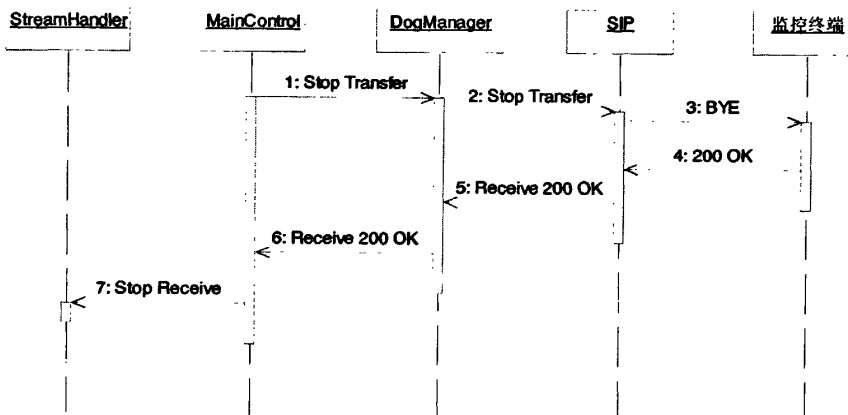


图 4-13 停止监视功能模块关系图

## 第五章 多功能显示终端子系统设计和实现

### 5.1 系统结构设计

系统根据功能可划分用户界面，编解码器，播放器，流管理器，终端管理器，信令模块等。模块间构造如图 5-1 所示：

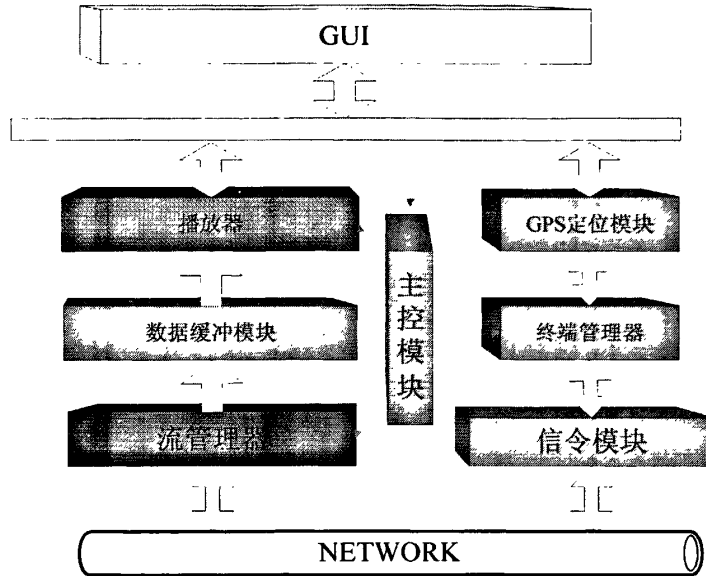


图 5-1 系统模块结构图

表 5-1 各模块说明

模块名	输入	输出	功能
GUI	1. 菜单命令 2. 按钮命令	1. 呈现 2. 发送命令给主控	呈现用户界面，接收用户输入
播放器	多媒体流	视频图像	媒体解码播放。
GPS 定位模块	GPS 位置信息	地图运动轨迹	显示终端运动轨迹
终端管理器	1. 终端信息 2. 终端标示	终端信息	根据终端标示管理终端信息。
数据缓冲模块	原始数据流	平滑数据流	平滑抖动，时延等
流管理器	数据流	分发后的数据流	分发输入的数据流
信令模块	1. 控制命令 2. 对端信令	1. 信令 2. 给其他模块的命令	信令控制与交互
主控模块	用户命令	对各模块的命令	控制其他各模块



## 5.2 GUI 设计

### 5.2.1 固定显示终端界面设计

#### 5.2.1.1 界面功能描述

移动视频监控系统固定显示终端播放界面应具有以下界面功能：

1、对监控终端的控制：

- a.开始播放，暂停播放，停止播放，终端信息查看。
- b.声音设置(播放声音，静音，音量调节)
- c.终端状态显示(已注册，连接中，已连接，正在缓冲，缓冲比例，播放中等等)

2、图像采集参数设置：

- a.亮度，对比度，色调，色度，颜色深度
- b.采集速率(x 帧/s)

3、监控摄像头控制：

焦距设置，方向控制，照明，雨刷器

4、编码方式 (H.263/H.264)，

图像编码大小(QCIF/CIF)，

5、GPS 信息的接入与显示

GPS 地理位置显示，地图显示

### 5.2.1.2 主界面设计



图 5-2 固定显示终端主控界面

作为固定显示终端软件系统与用户的交互界面，主控界面向用户提供了一些基本的接口，尽量达到简单直观的效果。部分按钮的功能如下（每个按钮都有 ToolTip 提示其功能）：

**Register 按钮：**注册

用户点击该按钮后，系统便向服务器发送注册消息，告知服务器自己的相关信息，同时也从服务器的注册回复消息中获取到自己所需要的信息，包括当前监控终端的信息。如注册成功，则终端状态显示子窗口中显示当前活动的监控终端状态。

**Play/Pause 按钮：**监视/暂停

在注册成功并且未进行其他操作时点击此按钮开始发出监视请求，经过一系列的信令交互后，监视开始，接收从监控终端发送过来的实时媒体数据并在主控界面的播放区域内显示实时监视画面。在播放监视画面的状态下，点击此按钮则会经过信令交互使监控变为暂停状态，重复点击此按钮是监视状态在两者之间切换。

**Stop：**停止监视

在监视过程中，用户点击该按钮后，系统发出 Bye 信令，可以停止当前监视。

### 5.2.1.3 图像设置窗口

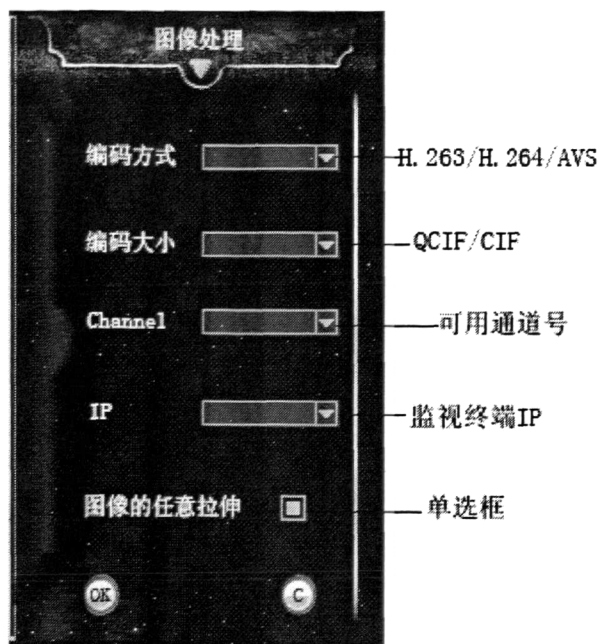


图 5-3 图像信息设置子窗口

此窗口初始化为隐藏形式，当点击主界面上的“图像设置”按钮时，调用此子窗口，对参数进行修改或者查看。

### 5.2.1.4 修改参数窗口

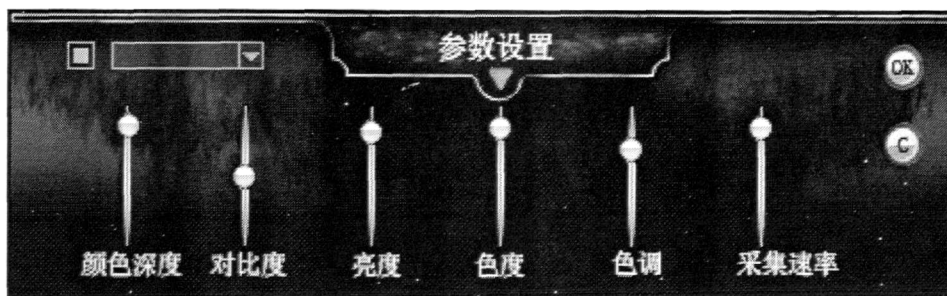


图 5-4 修改参数子窗口

此窗口初始化为隐藏形式，当点击主界面上的“采集参数设置”按钮时，调用此子窗口，用户可以根据需要重新设置参数（主要是媒体数据采集参数的设置），设置完毕后发出修改参数请求。

## 5.2.2 移动显示终端界面设计

### 5.2.2.1 界面功能描述

#### 1 对监控终端的控制:

A 开始播放, 暂停播放, 停止播放, 终端信息查看

B 声音设置(播放声音, 静音, 音量调节)

C 终端状态显示(已注册, 连接中, 已连接, 正在缓冲, 播放中等)

#### 2 图像采集参数设置:

亮度, 对比度, 色调, 色度, 颜色深度

#### 3 监控摄像头控制:

焦距设置, 方向控制, 照明, 雨刷器

#### 4 编码设置

编码方式 (H.263/H.264), 图像编码大小(QCIF/CIF), 采集速率(x 帧/s)

#### 5 图像拉伸(scale)

#### 6 GPS 信息的接入与显示

### 5.2.2.2 主窗口

主窗口是程序开始时显示给用户的窗口, 其余窗口均是围绕主窗口创建的, 移动显示终端的主要功能也将集中在主窗口。整个主窗口分为三个子窗口: 播放窗口、控制窗口和菜单栏。硬件设备 PPC 的分辨率是 240\*320, 为了满足界面需求分析中各窗口占主窗口的比例要求, 设置播放窗口大小为 240\*255, 控制窗口大小为 240\*40, 菜单栏窗口大小为 240\*25。视频显示窗口在播放窗口的中间位置, 正常显示时大小是 176\*144, 拉伸显示时大小是 216\*177。状态显示窗口是控制窗口的子窗口, 位于控制窗口的右边。控制窗口中另外包含了 5 个下按按钮, 分别是播放按钮、停止按钮、增加按钮、减小按钮和静音按钮。由于要求下按按钮有高亮显示, 因此每个按钮至少配置了两幅位图, 一幅用于正常显示另一幅用于高亮显示。5 个按钮一字排开位于控制窗口的左侧, 其中播放按钮的大小是 22\*22, 其余按钮的大小是 16\*16。主窗口的界面如图 5-5 所示

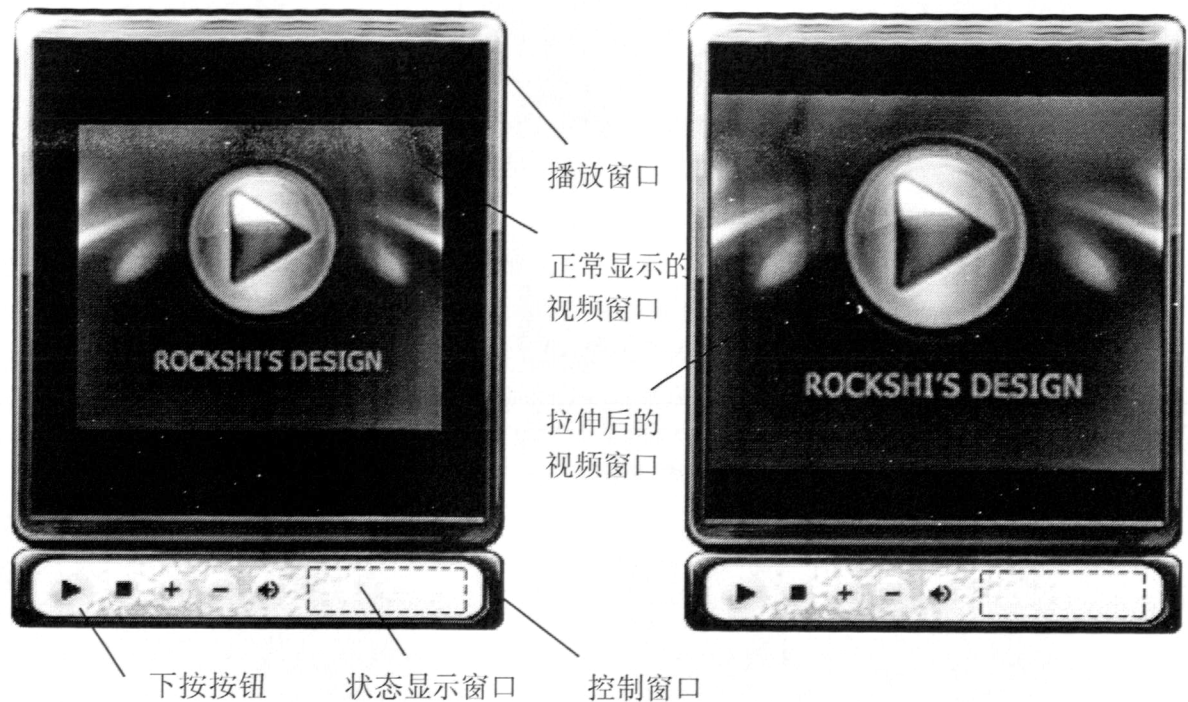


图 5-5 主窗口界面图

#### 5.2.2.4 注册窗口

注册窗口如图 5-6 所示，其采用 MFC 对话框模版，整个模版包含一个编辑框用于输入服务器的 IP 地址，两个组合框一个用于显示所有可用监控终端的 IP 地址，另一个用于每个监控终端所有可用的摄像头通道，一个静态文本框用于显示注册状态和一个下按按钮用于发送注册请求。

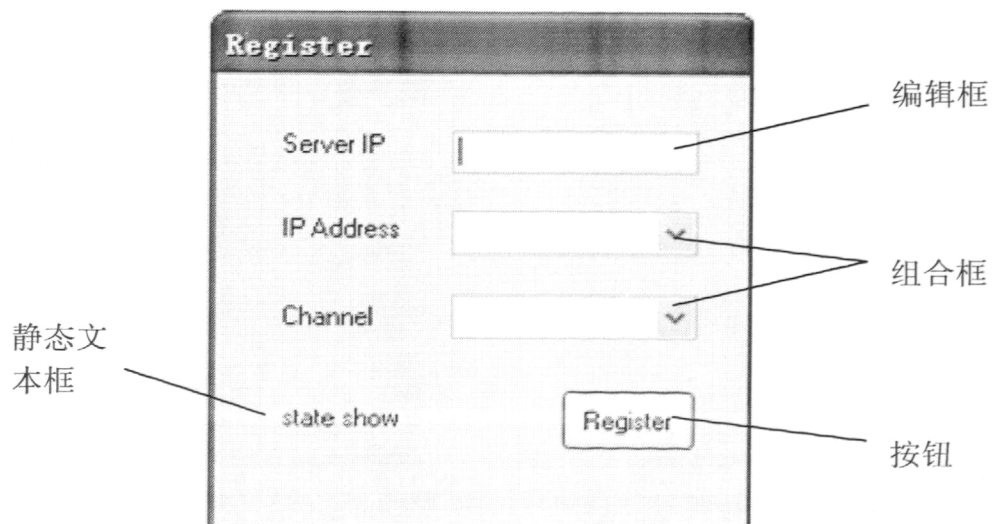


图 5-6 注册窗口界面图

### 5.2.2.5 编码设置窗口

编码设置窗口如图 5-7 所示，依旧采用 MFC 对话框模版，窗口包含两个组框一个用于选择编码格式另一个用于选择编码大小和一个下拉框用于选择采集速率。

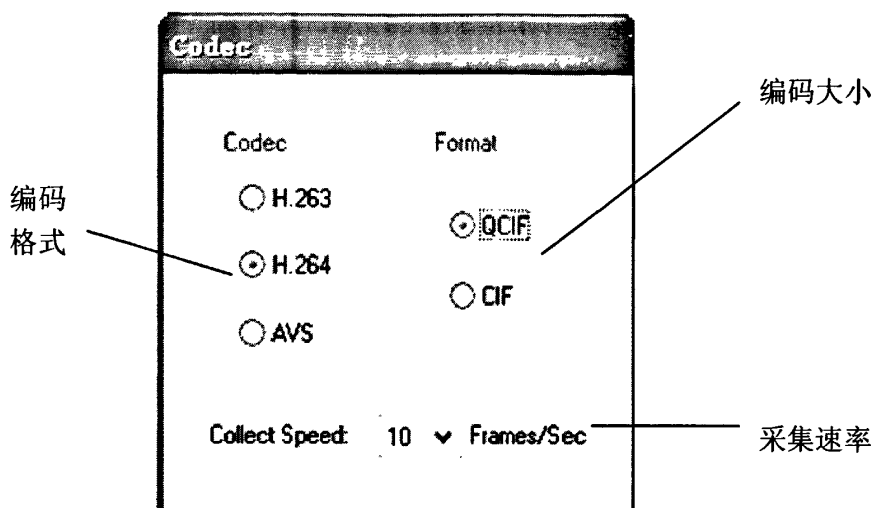


图 5-7 编码设置窗口界面图

### 5.2.2.6 菜单栏

菜单栏是由 MenuBar 实现的，考虑到程序运行后会全屏显示，这样 OK 按钮无法看到了，因此在 MenuBar 的左边设置了退出按钮，右边才是真正的菜单。移动显示终端主要的功能都会集中在菜单中，因此菜单需要包含界面的所有功能，根据各个功能之间的关系，菜单中各菜单项的分布如图 5-8 所示：

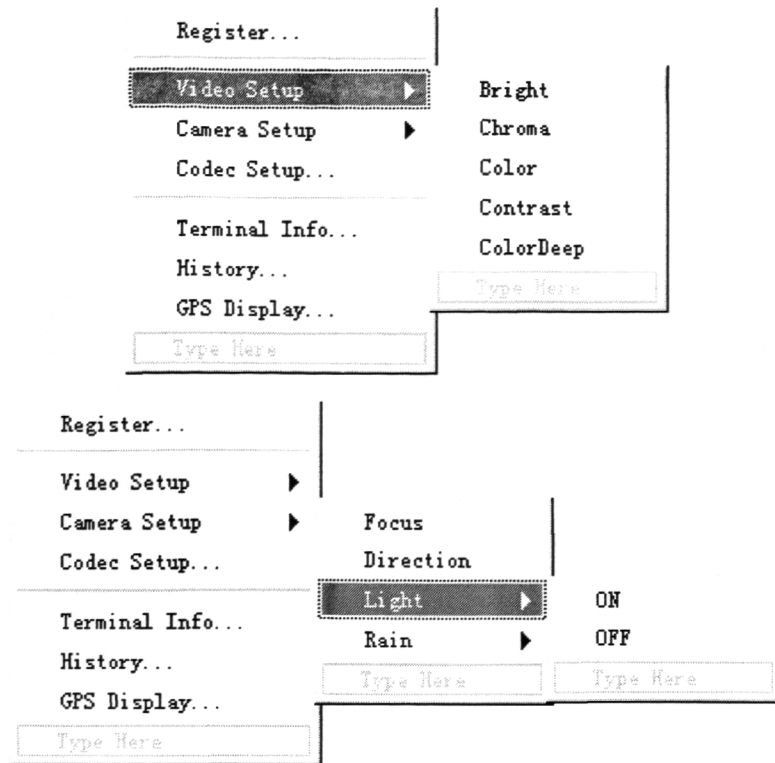


图 5-8 菜单项分布图

## 5.3 主要模块设计

### 5.3.1 终端管理器

终端管理器主要由两个类组成。一个是终端类 `CDog`，另一个是终端管理类 `CDogManager`。`CDog` 类用于描述一个监控终端及其相关参数和操作，`CDogManager` 类用于对多个终端 `CDog` 类的管理，并且系统会声明一个静态的 `CDogManager` 对象，从而保证该对象的唯一性，便于其他类对其的访问。当显示终端向监控服务器注册并收到回复之后，可以确定监控终端的个数，终端管理类会记录相应信息。当显示终端选定某一监控终端并向其发送 `invite` 时，`CDogManager` 类会生成一个 `CDog` 对象，对该监控终端进行管理，包括信令的交互，参数的修改等等。在对话完毕，拆除连接时，`CDogManager` 类会删除当前的 `CDog` 对象，进而销毁其占用的资源。这样，`CDogManager` 类通过对 `CDog` 类的控制，达到对监控终端控制的目的。

终端管理器在设计上需要解决的问题是终端管理类（`CDogManager`）仅需要一个实例，因为在系统运行时若存在多个终端管理类，则会导致各终端数据不统一，不便于管理，程序运行造成混乱。解决这一问题的方法是采用单件模式（`Singleton`）的设计模式，即让终端管理类自身负责保存它的唯一实例。这个类通过截取创建新对象的请求，保证没有其它实例可以被创建，并且还要提供一个访问该实例的方法<sup>[8]</sup>。

终端管理器在设计上需要解决的另一个问题是对于 `SIP` 消息的处理方法。本系统中采用了 `windows` 的消息机制，将每个 `SIP` 消息映射为对应的 `windows` 自定义消息，这种处理方法既简单高效而且也与上层模块（`GUI`）的消息机制取得了统一，方便开发，提高了开发效率。

`CDog` 及 `CDogManager` 类的设计如图 5-9 所示：



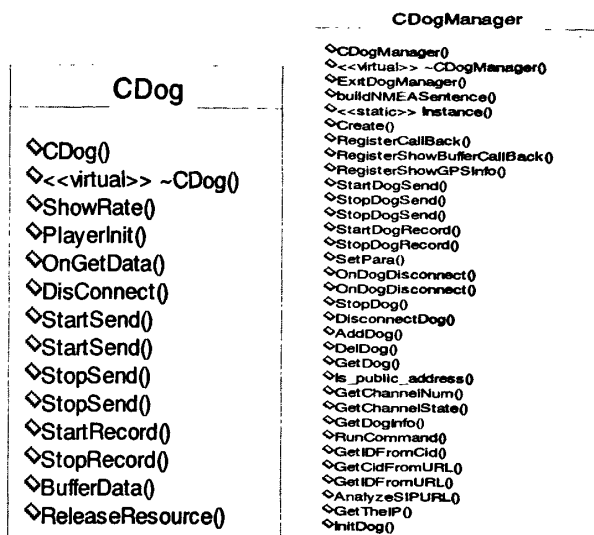


图 5-9 终端及终端管理类设计

表 5-2 CDog 类说明

操作	描述
StartSend()	向终端发送信令，开始数据的发送。
StopSend()	停止当前数据发送。
DisConnect()	断开当前连接。
OnGetData()	当接收到视频数据时的回调函数。
BufferData()	视频数据缓冲函数。
ReleaseResource()	释放占用的资源。

表 5-3 CDogManager 类说明

操作	描述
StartDogSend()	向当前选定监控终端发送开始命令。
StopDogSend()	停止当前终端的数据发送。
DisconnectDog()	断开当前终端的连接。
Create()	创建时的初始化等系列操作。
AddDog()	添加一个 CDog 对象，并设定默认参数。
DelDog()	删除一个已有的 CDog 对象。
InitDog()	初始化当前选中的 CDog 对象。
Instance()	静态函数，声明 CDogManager 的静态实例。

### 5.3.2 信令模块

信令模块负责对 SIP 信令的发送与接收控制，主要包括 Register, Invite 等信令。针对这几个主要信令，信令模块划分为注册管理器(RegistrationManger), Invite 管理器(InviteManger)和对话管理器(CallManger)这几个类。RegistrationManger 主要职责是注册信息的初始化与发起。InviteManger 的职责为 Invite 信息的初始化和发起。最为重要的是 CallManger 类，它负责对大部分信令交互的控制，因为 SIP 信令是以对话为基础的，而 CallManger 正是这些对话的管理器，通过 CallManger 类，系统实现了对信令交互的控制。类设计如图 5-10 所示：

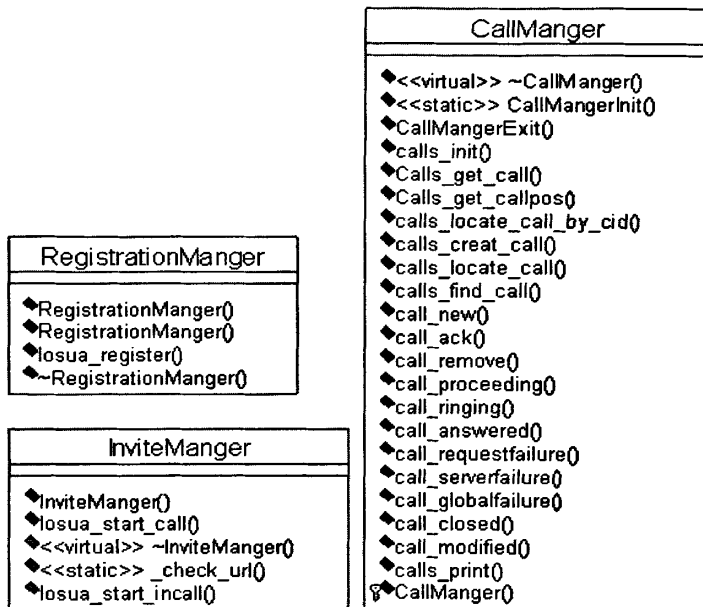


图 5-10 信令控制模块类设计

表 5-4 信令控制模块类说明

操作	描述
losua_register()	构造注册信令并发出。
losua_start_call()	开始一个对话。
losua_start_incall()	开始一个对话内的 invite。
calls_init()	初始化一个对话。
call_new()	添加一个新的对话。
call_closed()	停止一个对话。
call_remove()	移除一个已经存在的对话。
call_ack()	构造 ack 消息并发出。

### 5.3.3 流管理器

流管理器包括数据流类 `Stream` 和流管理类 `StreamHandler`。其中数据流类 `Stream` 是对 RTP 部分的一个封装，主要负责 RTP 的交互。`StreamHandler` 类是负责对多个数据流的管理，即 `StreamHandler` 可以管理多个 `Stream` 对象。`StreamHandler` 使用 `instance` 函数设定为一个静态类，便于其他类对其的访问。类的设计如图 5-11 所示。

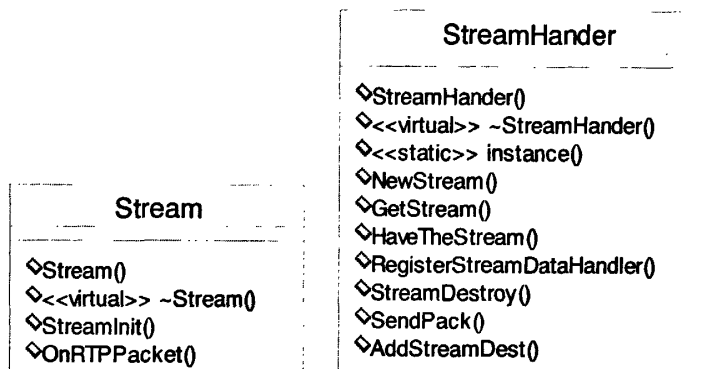


图 5-11 流管理模块类设计

表 5-5 流管理模块类说明

操作	描述
<code>StreamInit()</code>	初始化一个数据流。
<code>OnRTPPacket()</code>	收到 RTP 数据包后的回调。
<code>instance()</code>	声明 <code>StreamHandler</code> 为静态类，通过此函数获取 <code>StreamHandler</code> 实例。
<code>NewStream()</code>	创建一个数据流。
<code>GetStream()</code>	获取当前数据流。
<code>HaveTheStream()</code>	判断数据流是否存在。
<code>StreamDestroy()</code>	销毁一个数据流。
<code>SendPack()</code>	发送 RTP 数据包。

### 5.3.4 播放器

因系统要支持多种视频数据的编解码，所以，要针对每种支持的编解码格式设计一个播放器类，而这些播放器类有很多相同的对外接口，所以，在设计时，使用了继承类的设计方法。首先声明一个父类 `CPlayer`，其他的所有播放器类均由此类继承而来。针对系统要支持的 H.264 和 H.263 视频编解码标准，在设计时，

设计了 CPlayer 的两个子类：CPlayer264Stream，CPlayer263Stream。这两个类分别是 264 和 263 的视频数据解码播放类。在类的函数设计时，使用虚函数方法，从而实现了子类父化情况下播放器的正常播放，即在调用播放器时，可以统一使用父类作为播放器接口，使得函数调用关系更为简洁，类的应用更为方便。类关系图如图 5-12 所示：

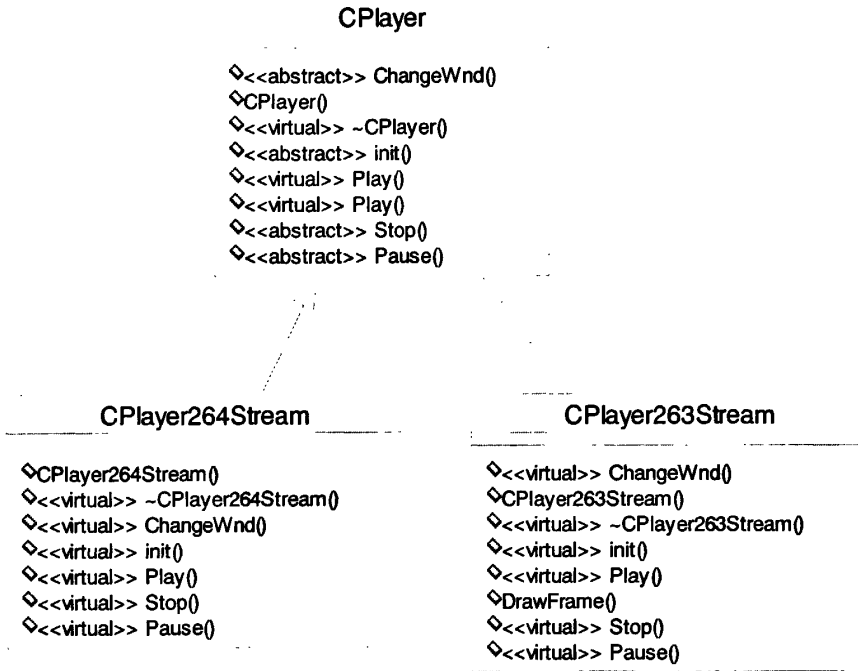


图 5-12 播放器模块类设计

表 5-6 播放器模块类说明

操作	描述
init()	初始化播放器。
Play()	播放视频。
Stop()	停止当前播放。
Pause()	暂停当前播放。
ChangeWnd()	改变播放窗口的句柄。

### 5.3.5 数据缓冲模块

为了消除无线网络的时延，带宽及抖动对视频播放的影响，本系统设计了一种基于反馈的视频流媒体缓冲机制，该机制可以根据当前的网络状况，自适应地调整缓冲区的大小，并利用反馈信息对丢失或发生错误的数据进行判断，然后根

据当前网络状况以及数据的重要性,进行选择性的重传,之后使用差错抑制信息对未能重传的视频帧进行恢复。该机制可以较好的适应各种网络,并对视频流媒体的传输效果有相当大的改善。

该缓冲机制采用延迟约束的重传机制与差错抑制相结合<sup>[9][10]</sup>,针对不同的网络状况,动态调整缓冲区的大小,并根据网络状况及视频数据帧的重要性进行选择性的重传,然后对其他产生错误的视频帧进行重构,从而有效的完成视频数据的恢复。

### 5.3.5.1 缓冲机制概述

本系统中的缓冲机制使用了延迟约束的重传机制与差错抑制相结合的方法,在数据发送端及接收端都设定数据缓冲区,其中,发送端缓冲区存储已经发送的数据包,发送端收到重传请求时,首先分析所请求数据包的重要性,然后根据其重要程度以及当前的网络状况,选择是否对其进行重传。接收端在接收数据时,对数据包进行错误检查,如若产生错误或丢包,则发送重传请求,如果是正确的数据包,则根据其存储的相关数据,判断它在接收缓冲区中所应在的位置,然后放入缓冲区内。最后,根据已传送成功的数据,对其他丢失的数据进行修复。

该机制在数据发送端和接收端均由三个模块构成,即网络监测模块,信息反馈处理模块和数据传输缓冲模块。网络监测模块用于对网络状况的测量;信息反馈处理模块任务在接收端对丢包及误码进行分析统计,并将结果通过信息通道反馈给发送端,在发送端对这些反馈信息进行分析处理;数据传输缓冲模块在发送端功能为对正常数据与重传数据的发送,在接收端则包括数据的缓冲及视频帧的重构。机制构成如图 5-13 所示:

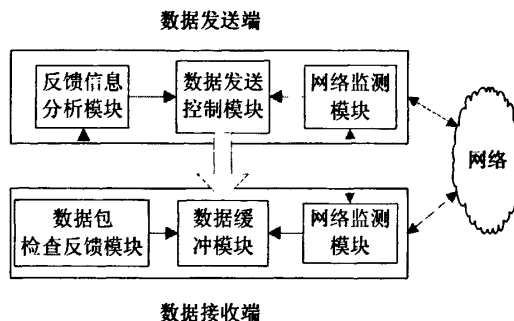


图 5-13 缓冲机制结构图

### 5.3.5.2 发送端机制

根据上述缓冲机制结构,可将数据发送端分为三个功能模块:网络监测模块,反馈信息处理模块,数据发送模块。网络监测模块负责响应接收端发出的监测数据,分析当前的网络状况,并根据当前网络状况,设定重传视频帧的选择原则;反馈信息处理模块负责对反馈信息进行处理,并对丢失数据包的重要性进行判断;数据发送模块负责正常视频数据的发送以及丢失的关键数据的重传。数据发送端结构如图 5-14 所示:

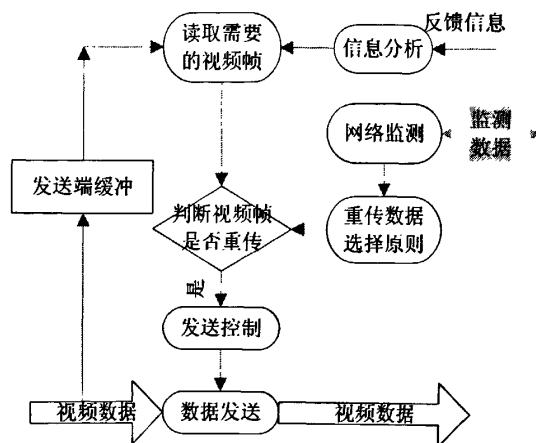


图 5-14 数据发送端处理流程图

### 5.3.5.3 接收端机制

与发送端功能模块划分相似,接收端也可以划分为三个功能模块,包括网络监测模块,数据包检查反馈模块和数据缓冲模块。

首先,由网络监测模块对网络状况进行探测,根据探测的结果,对缓冲区大小进行相应的设定。在之后的过程中,网络监测模块会周期性探测网络状况的变化,并对缓冲区大小做出实时调整<sup>[11]</sup>。

数据包接收过程中,数据包检查反馈模块会对接收到的数据包进行检查,如果有丢包或误码产生,则向发送端发送重传请求。如果是正确的数据包,则检查数据包的序号,确定其是正常顺序包或是重传数据包,然后由数据缓冲模块放入该数据包在缓冲区中所对应的位置。

数据缓冲模块由存储缓冲,视频帧重构和输出队列组成。其中,存储缓冲负责将接收到的正常顺序数据包及丢失后重传的数据包按其自身所带有的序列号排列成正确的顺序,存储在缓冲区内,之后视频帧重构模块根据已接收的数据对传输失败的数据进行恢复重构。经过缓冲重传及重构后,处理完毕的数据包被送入输

出队列，等待播放程序的读取。存储缓冲由两个相同大小的缓冲区组成，两个缓冲区交替作为重传存储缓冲区和数据输出缓冲区。之所以采用此种机制，是因为如果只使用单个缓冲区，则需要对该缓冲区进行循环式利用，而循环式利用在处理重传数据时很容易造成新旧数据的覆盖，致使输出数据时产生错误<sup>[12]</sup>。如果使用多于两个的缓冲区，则会造成资源的浪费。所以，选择两个缓冲区交替使用，在占用资源尽量少的情况下，达到缓冲所需要的效果。接收端模块处理流程如图 5-15 所示：

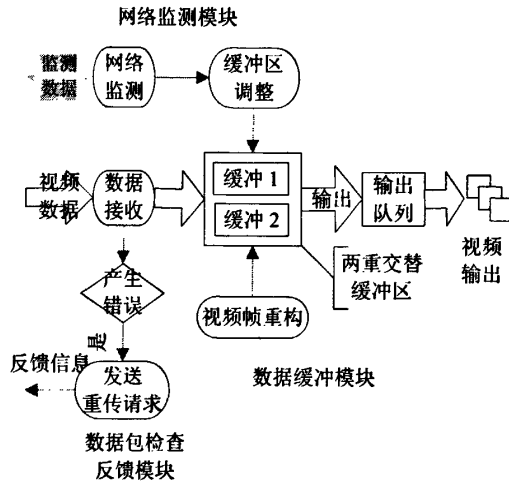


图 5-15 数据接收端模块处理流程

缓冲区大小的设定，既要考虑数据包反馈重传所需要的时间，也要考虑视频数据缓冲后的时延大小。在可以满足反馈重传效果的前提下，尽可能的减小数据缓冲的时延。另外，缓冲区还需要去除网络抖动对数据传输的影响，达到数据输入与输出平均速率的平衡<sup>[13]</sup>。

视频采集播放帧率以  $Fr$  表示，网络传输延迟为  $T_{netdelay}$ ，丢包链表查询间隔时间为  $T_{check}$ ，缓冲区的大小设定为  $S$ ，从重传效率及缓冲时延考虑，为了防止同一数据包的多次重复重传，设定数据包重传次数最高为两次。而经过两次重传以后，大部分情况下，数据包都可以成功到达接收端。也鉴于此，缓冲区的大小设定为

$$S = Fr \times (2 \times T_{netdelay} + 2 \times T_{check}) \quad \text{式 (5-1)}$$

对  $T_{check}$  较为合理的设置是将其设置为网络传输延迟，即在丢包链表查询完成并发送重传请求后，等待重传数据到达，链表更新完毕，然后再对链表进行新一轮的查询。即  $T_{check} = T_{netdelay}$ 。因此，可得出缓冲区大小为

$$S = Fr \times (4 \times T_{netdelay}) \quad \text{式 (5-2)}$$

因为网络传输延迟  $T_{netdelay}$  是在某一范围内波动，会造成缓冲区大小的不断变化，加大系统开销并影响运行效率<sup>[14]</sup>，所以对  $T_{netdelay}$  采取上取整的计算方法，即

$$S = Fr \times \left( 4 \times \left\lceil T_{netdelay} \right\rceil \right) \quad \text{式 (5-3)}$$

### 5.3.5.4 仿真分析

本缓冲机制的一大特点就是可以针对网络状况的变化，对缓冲区的大小进行实时调整；针对此特点，在 Matlab 环境下，对缓冲区大小与网络状况变化之间的关系进行了仿真，因为缓冲区对网络时延的数值变化较为敏感，所以，选择网络传输时延作为网络状况的测量元素。在仿真中，视频帧传送速率设定为 10fps，网络探测与缓冲区调整的时间单位设定为 1 分钟。图 5-16 是对网络状况变化的仿真，其中，在每个网络探测时间单元中，使用一个 1~10 范围内的随机数来表示当前网络时延。而图 5-17 则是在图 5-16 所示的网络状况下，缓冲区大小所产生的相应变化。由图 5-16 和图 5-17 可以看出，本文所描述的缓冲机制可以根据网络状况变化，实时调整缓冲区的大小，达到对视频数据较好的缓冲效果。

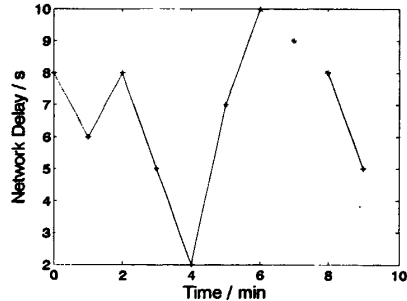


图 5-16 网络状况变化

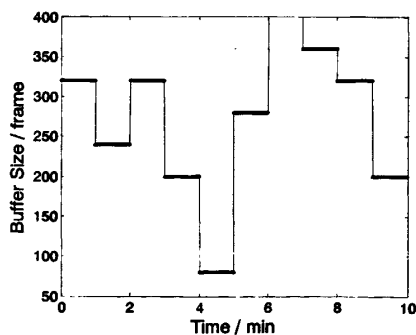


图 5-17 缓冲区大小调整



### 5.3.6 GPS 定位

显示终端通过 SIP 的 NOTIFY 信令获取到监控终端的位置信息, 在该位置信息进行分析之后, 显示终端需将位置信息输出。系统实现了两种常用的 GPS 位置信息输出方式, 一种是将位置信息转换成串口信息, 并将串口信息递交给以串口信息作为输入的地图软件, 之后地图软件即可将监控终端位置在地图上显示出来; 另一种输出方式是将位置信息以 XML 的形式写入文件, 由另一个地图程序不断的从文件中读取位置信息, 并在地图上显示出来。图 5-18 分别说明两种输出方式下的数据结构设计, 其中, CSerial 类是以串口形式输出时的串口操作类, 而 locationInfo 是以文件方式输出时的位置信息结构体。表 5-7 描述了以串口形式输出时的数据结构。

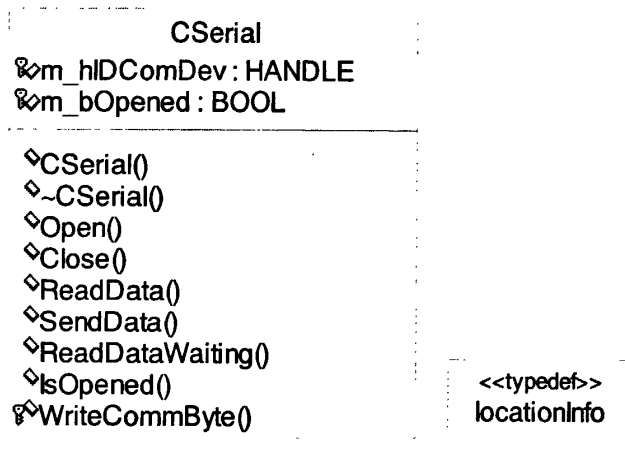


图 5-18 GPS 位置信息数据结构设计

表 5-7 CSerial 类借口说明

操作	描述
Open()	打开串口。
Close()	关闭串口。
ReadData()	读取位置数据。
SendData()	发送位置数据。
ReadDataWaiting()	位置数据读等待。
IsOpened()	判断串口是否打开。

## 5.4 主控模块的工作流程

根据监控系统中各个功能模块之间的关系以及有线监控终端的系统构架，主控模块向服务器注册、发送监视请求、参数修改请求、收到参数修改请求及停止监视的具体工作流程如下：

### 5.4.1 向服务器注册

主控模块启动后进入等待注册命令发出的状态，在接收到注册命令后，从本地配置文件中读取服务器的相关信息，然后向服务器发出注册消息，并等待服务器的响应。在收到服务器的回复消息时，根据消息类型来判断注册是否成功并提示用户。此过程的处理流程如图 5-19 所示：

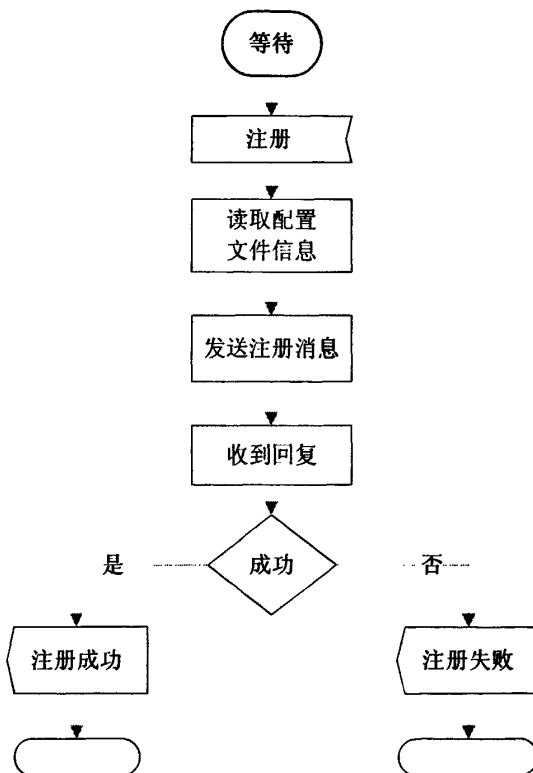


图 5-19 向服务器发送注册请求的处理流程

### 5.4.2 发送监视请求

在主控程序为等待状态时，如果收到开始监视的命令，主控首先检查是否已经注册，如果未注册，则提示用户；如果已经注册，则发送 Invite 消息，之后等待对端的回复，如果收到的消息不是成功消息，则提示请求失败；如果回复消息为成功，主控则会初始化数据传输模块 RTP 和播放器模块，然后发送确认消息 ACK 给对端，之后就进入等待接收数据的状态。此处理流程如图 5-20 所示：

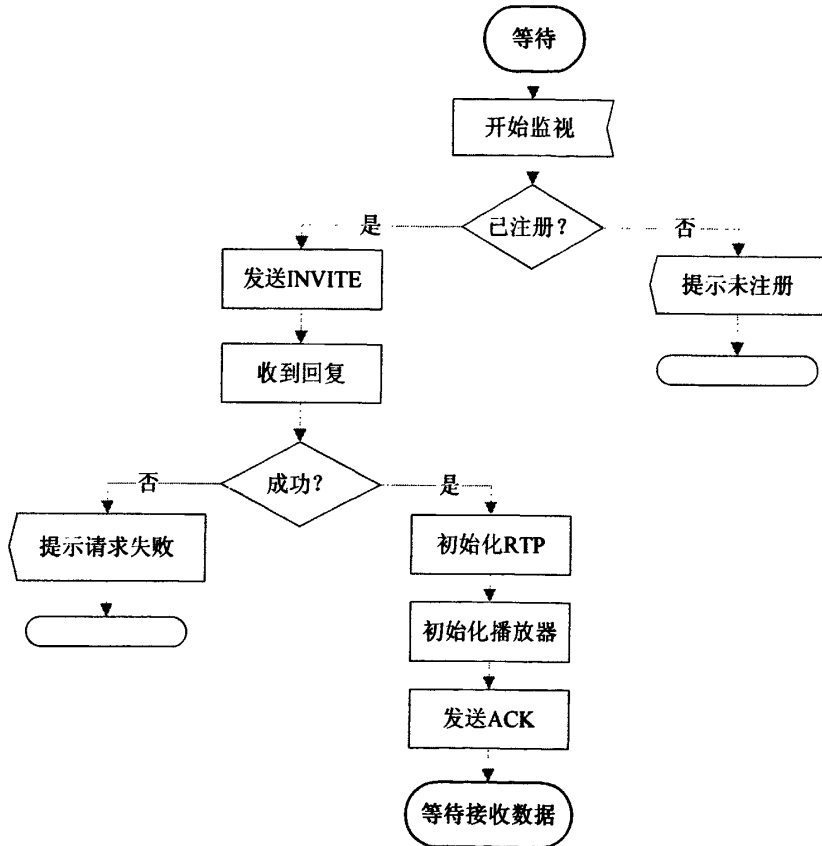


图 5-20 发送监视请求的处理流程

### 5.4.3 参数修改

在数据传输过程中，主控会随时等待修改参数的命令，当接收到修改参数命令时，会向对端发送对话内 Invite 消息，之后接收对端的回复消息。如果回复消息为请求失败，则提示用户参数修改失败；如果回复的是成功消息，主控程序则向对端发送确认消息 ACK，并提示用户修改成功。此处理过程如图 5-21 所示：

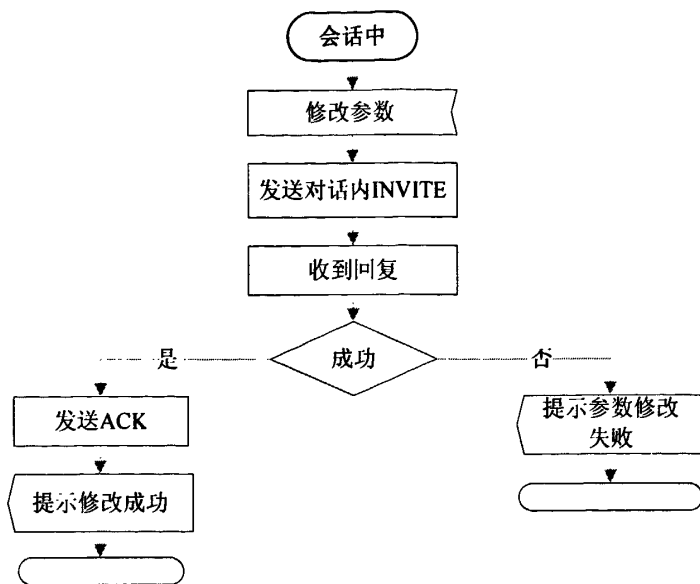


图 5-21 参数修改请求的处理流程

#### 5.4.4 终止会话

在会话进行过程中，如果用户要终止会话，则会由主控程序向监控终端发送 Bye 消息，在收到回复消息后，判断是否成功。如果成功则将会话相关资源释放，并提示用户会话已终止；如果没有成功，则提示用户终止会话失败。此处理流程如图 5-22 所示：

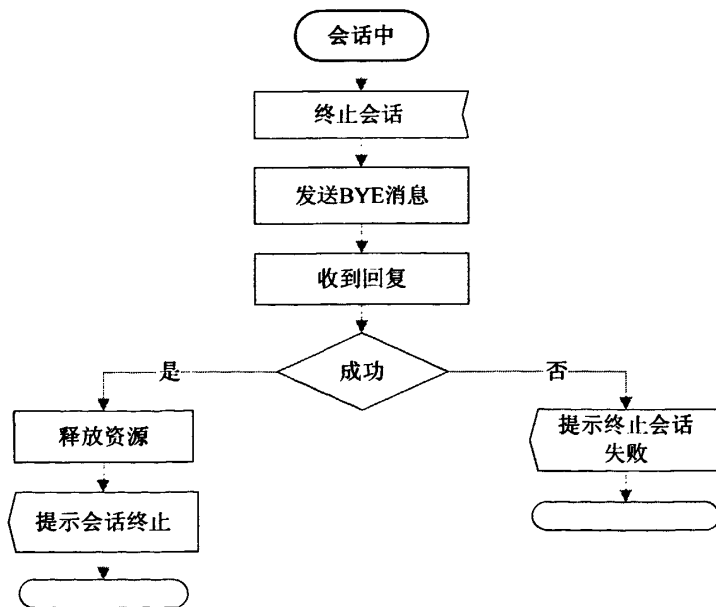


图 5-22 终止会话请求的处理流程

## 第六章 多功能显示终端子系统测试与分析

### 6.1 单元测试

针对第五章中描述的各个功能模块，分别进行了单元测试，包括功能测试和性能测试等，对各模块所应具有的功能以及可能的出错情况进行了实际测试。但鉴于篇幅限制，无法将所有测试结果都呈现出来，在下述的各模块测试结果中，只是对各个模块的正常功能测试用例进行了描述。

#### 6.1.1 终端管理器测试

如表 6-1 所示为终端管理器的测试用例

表 6-1 终端管理器测试用例

模块名称	终端管理器
测试类型	功能、可靠性、边界
用例描述	响应主控的调用。 调用流管理器模块，编解码模块，播放器模块有关功能。
前置条件	流管理器、编解码、播放器、信令模块的实现。
步骤	<ol style="list-style-type: none"> <li>1. 向监控服务器发起注册。</li> <li>2. 调用 StartSend 函数。</li> <li>3. 调用 StopSend 函数。</li> </ol>
输入数据	<ol style="list-style-type: none"> <li>1. 注册消息</li> <li>2. 终端 ID 号</li> </ol>
预计结果	<ol style="list-style-type: none"> <li>1. 通过步骤 1，有关于监控终端的对象生成。</li> <li>2. 通过步骤 2，调用信令模块发送 INVITE 接口，调用流管理器生成新的 RTPSession 会话，调用播放器播放函数。</li> <li>3. 通过步骤 3，调用信令模块发送 BYE 接口，调用播放器停止播放函数。</li> </ol>
实际结果	与预计结果相符
结论	终端管理器实现了预计功能。

### 6.1.2 流管理器测试

如表 6-2 所示为流管理器测试用例。

表 6-2 流管理器测试用例

模块名称	流管理器
测试类型	功能、性能、可靠性、边界、强度
用例描述	生成, 拆除 RTPSession, 通过回调机制调用接收到包的数据处理函数。
前置条件	监控终端的流管理器模块
步骤	<ol style="list-style-type: none"> <li>1. 调用 SreamHandler::AddStream 函数, 指定流的 ID 号。</li> <li>2. 开始从终端传输媒体流。</li> <li>3. 一段时间后, 停止传输媒体流。</li> <li>4. 调用 StreamHandler::DelStream 函数, 指定流的 ID 号。</li> </ol>
输入数据	<ol style="list-style-type: none"> <li>1. RTP 数据流</li> <li>2. RTP 流 ID 号</li> </ol>
预计结果	<ol style="list-style-type: none"> <li>1. 通过步骤 1, 生成新的 RTPSession 对象。</li> <li>2. 通过步骤 2、3, 执行了注册的回调函数。</li> <li>3. 通过步骤 4, 销毁步骤 1 中生成的 RTPSession 对象</li> </ol>
实际结果	与预计结果相符
结论	流管理器实现了预计功能

### 6.1.3 播放器测试

如表 6-3 所示为播放器测试用例。

表 6-3 播放器测试用例

模块名称	播放器
测试类型	功能、性能、可靠性
用例描述	<p>播放 H263、H264 格式多媒体流</p> <p>输入播放窗口和多媒体流, 在窗口上进行播放。</p>
前置条件	<p>编解码管理器的实现与测试</p> <p>263 媒体流的生成</p> <p>264 媒体流的生成</p>

步骤	1. 打开 263 媒体流 2. 调用 Play(IN UINT nStreamID)方法 3. 打开 264 媒体流 4. 调用 Play(IN UINT nStreamID)方法
输入数据	263 多媒体流 264 多媒体流 流标示符
预计结果	通过步骤 1、2 能够在窗口播放多媒体流 通过步骤 3、4 能够在窗口播放多媒体流
实际结果	与预计结果相符
结论	播放器实现了预计功能

#### 6.1.4 信令测试

系统对信令通信模块进行了测试，测试信令通信是否正常。只有显示终端与监控服务器以及监控终端的信令交互正常工作，数据传输才能保证，因此信令交互是保证程序正确运行的关键，此外将信令与数据分开测试在出现问题时有利于分析问题也利于解决问题。

**测试环境搭建：**由于所有的数据流在传输层采用的是 UDP 协议，UDP 是不能穿越内网地址的，而所有的信令在传输层采用的 TCP 协议，TCP 是可以穿越内网地址的。因此本文利用 UDP 和 TCP 的不同，将移动显示终端无线连接到内网地址上，其通过内网网关转发数据。

**测试过程：**测试过程包括正常流程测试、稳定性测试和容错测试。正常流程测试是将移动显示终端的各个功能按照正常流程运行一次，在信令流经过函数的地方输出打印信息，程序运行完毕后查看输出信息判断信令交互是否正常。稳定性测试是将系统不停地按照正常流程操作各个功能，依然通过打印信息的方式查看系统信令交互是否正常、系统内存是否泄漏、系统是否异常退出等。容错测试是测试程序在用户可能不按正常流程操作时是否会提示用户操作错误并告知用户如何正确操作，系统否能够对非用户引起的错误如网络错误有容错能力而保证系统不崩溃等。

**测试结果：**正常流程测试符合预期结果，稳定性测试系统不会出现异常退出等问题，所有信令交互均正常工作。容错测试基本符合预期结果，但是错误处理还不够完善，需要对部分程序进行改进。

如表 6-4 所示为信令模块测试用例：

表 6-4 信令模块测试用例

模块名称	信令模块
测试类型	功能、性能、可靠性、边界
用例描述	实现 REGISTER, INVITE, BYE 消息的发送以及对应 SIP 消息回应，能够执行注册的回调函数。
前置条件	无
步骤	<ol style="list-style-type: none"> <li>1. 调用回调函数注册函数。</li> <li>2. 从另一终端向测试端发送注册消息。</li> <li>3. 调用发送 INVITE 消息函数。</li> <li>4. 调用发送 BYE 消息函数。</li> </ol>
输入数据	回调函数地址，SIP 消息目的地址，SDP 数据
预计结果	<ol style="list-style-type: none"> <li>1. 通过步骤 2，执行了步骤 1 中注册的回调函数。</li> <li>2. 通过步骤 3，目的终端收到 INVITE 消息，测试端收到 200OK 消息后，执行步骤 1 中注册的回调函数，并发送 ACK 消息向目的终端。</li> <li>3. 通过步骤 4，目的终端收到 BYE 消息，测试端收到 200OK 消息后，执行步骤 1 中注册的回调函数。</li> </ol>
实际结果	与预计结果相符
结论	信令模块实现了预计功能

### 6.1.5 数据缓冲模块测试

在数据缓冲模块的测试实验中采用 H.264 作为视频编解码标准，实验环境选取无线局域网，视频帧率固定为 10fps(此帧率可以较好的适应各种无线网络环境下的传输，并且，在此帧率下，视频播放已经可以达到较好的主观效果)，分别在低负载，中等负载及高负载网络条件下进行了实验，以使用缓冲机制与未使用缓冲机制的情况进行了对比。

在低负载条件下(10%负载)，网络状况较好，丢包及误码现象发生的概率较低，在使用了缓冲机制后，对这些发生异常的数据包都可以成功的重传或重构，在实际测量中，基本可以达到 100%的传输成功率。图 6-1 即为在低负载情况下，使用缓冲机制与未使用缓冲机制的对比。



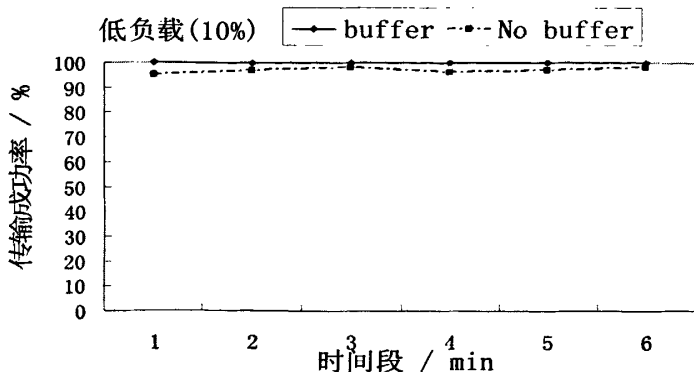


图 6-1 低负载传输成功率对比

在低负载情况下，本文中的缓冲机制的作用并没有得到完全的发挥，但在中等负载或高负载条件下，该缓冲机制的作用就得到的较好的体现。图 6-2 和图 6-3 分别为中等负载条件(50%负载)和高负载条件(90%负载)下的使用缓冲机制与未使用缓冲机制的传输成功率对比。从中可以看出，在这两种条件下，丢包及误码现象较为严重，使用本文中的缓冲机制可以明显提高数据传输的成功率。

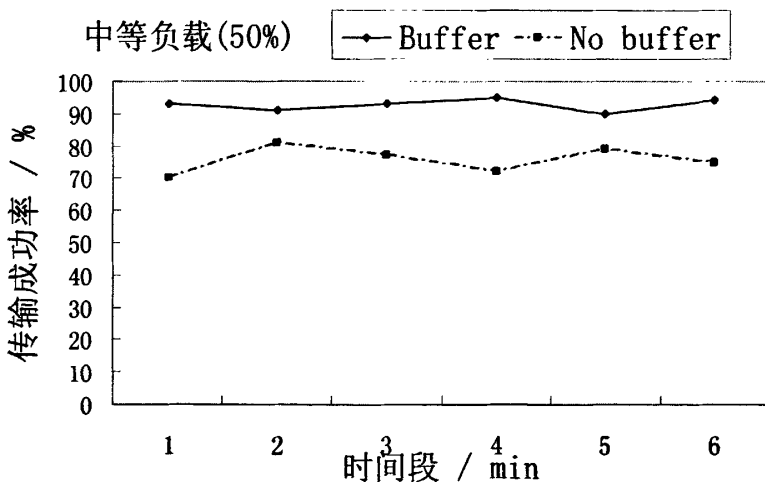


图 6-2 中等负载传输成功率对比

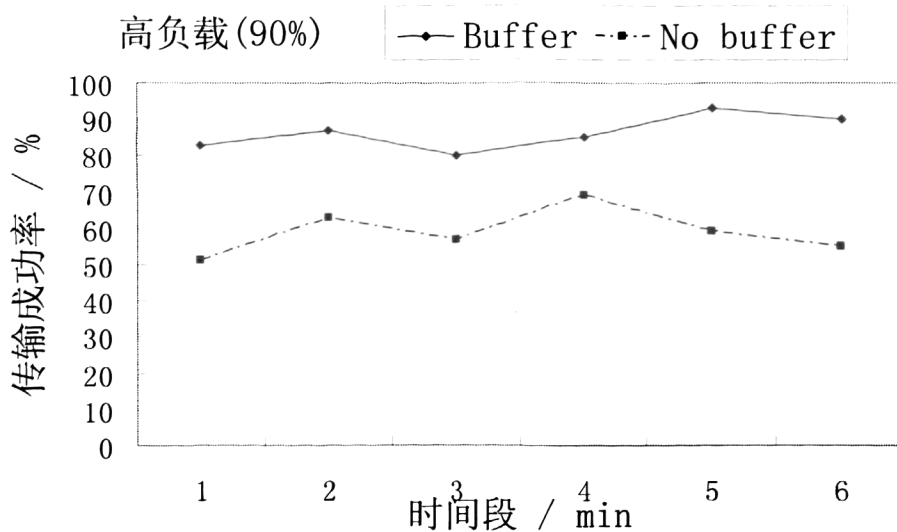


图 6-3 高负载传输成功率对比

针对当前常用的 CDMA1X 无线网络，系统进行了视频传输效果测试。CDMA1X 网络理论传输速率可达 153.6 kbps，但经实际测试该网络环境下平均传输速度大约为 60kbps。测试时以 H.264 作为视频编解码标准，选用一组剧烈运动的视频序列 football 作为测试序列，测试帧率为 10fps。该环境下未使用和使用缓冲机制时的播放效果如图 6-4 所示。由图中可以看出，使用缓冲机制进行传输后，视频播放效果有了明显改善。

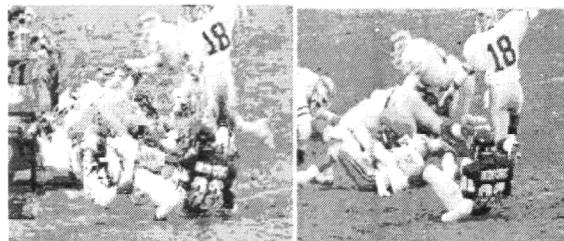


图 6-4 (a)No buffer

(b)With buffer

本缓冲机制可针对不同网络状况，动态调整缓冲区大小，并对丢包及误码数据进行重传与重构处理，可以较好的实现视频流媒体的传输。在无线局域网和 CDMA1x 网络环境下进行的实际测试，本机制均达到了较好的效果。

### 6.1.6 GPS 定位测试

多功能显示终端可以分为移动显示终端和固定显示终端。其中，移动显示终端是部署在移动设备上，因地图软件限制，所以当前方案是将 GPS 信息以文字的形式呈现出来。而固定显示终端则不同，其部署在 PC 上，可以使用地图软件来显示监控终端的位置信息。如图 6-5 所示，即为监控终端在固定显示终端地图上的运行轨迹。经测试，GPS 定位功能运行正常。



图 6-5 GPS 定位轨迹图

## 6.2 系统测试与联调

### 6.2.1 测试场景

目前系统由三部分组成：

1. 视频监控服务器：以固定 IP 地址接入 Internet 的微型计算机（Windows 操作系统）；
2. 监控客户终端：共有三个，均为 Linux 操作系统，包括一台带有摄像头、GPS 接收器和 CDMA 卡的笔记本电脑（CDMA 1X 接入），一台带有摄像头的工控机（以有线方式接入），一台带有摄像头、GPS 接收器和无线网卡（以 WLAN 无线方式接入）
3. 多功能显示终端：共有两个，包括一台微型计算机（有线方式接入，Windows 操作系统）和一部 PDA（WLAN 无线方式接入，Windows Mobile 操作系统）。

系统测试场景如下图所示：

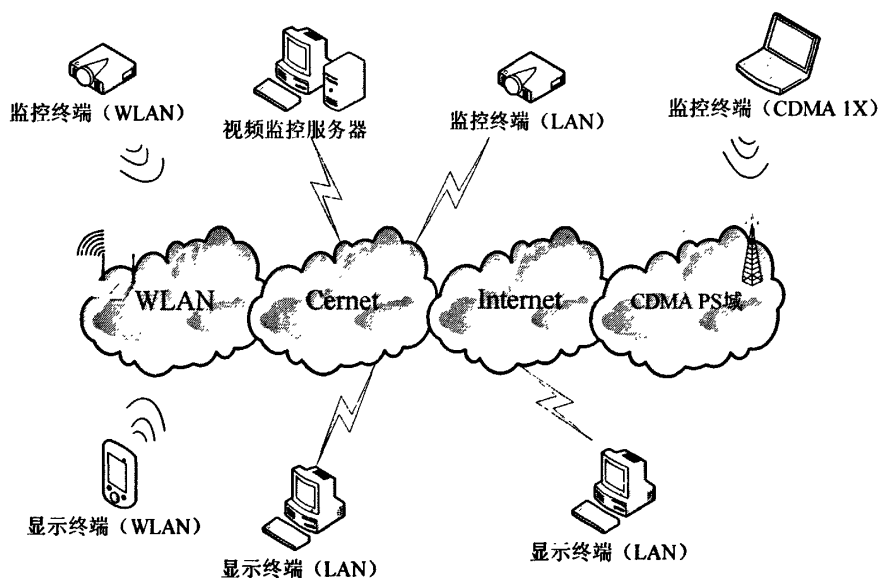


图 6-6 视频监控系统中测试场景图

## 6.2.2 测试功能

### 6.2.2.1 固定显示终端功能测试

需测试功能:

固定显示终端通过有线接入方式可实现与多种类型的监控终端（WLAN、LAN、CDMA 1X）的连接，实现对多种监控终端的定位，能够在 googlemap 中进行移动轨迹显示，并记录监控终端的移动轨迹，可以实现对监控终端的图像相关参数的调整，图像的编码格式可以支持 H.264，图像支持 QCIF、CIF 格式采集，播放画面的大小可以是 QCIF、CIF。

操作步骤:

- A. 启动服务器程序和三个监控终端的程序
- B. 启动固定显示终端程序，向监控服务器进行注册
- C. 选择播放格式为 QCIF，控制某个监控终端开始采集图像，点击图像播放画面，对播放窗口进行缩放，演示 QCIF 格式缩放效果
- D. 控制监控终端停止播放，调整播放格式为 CIF，并令监控终端重新开始采集图像，点击图像播放画面，对播放画面进行缩放，演示 CIF 格式缩放效果
- E. 打开网页显示 googlemap
- F. 调整对比度、色度等采集参数

### 6.2.2.2 移动显示终端（PDA）功能测试

需测试功能:

PDA 通过 WLAN 无线接入方式可实现与多种类型的监控终端（WLAN、LAN、CDMA 1X）的连接，实现对多种监控终端的定位，可以实现对监控终端的图像相关参数的调整，图像的编码格式可以支持 H.264，图像支持 QCIF 格式采集，播放画面的大小可以是 QCIF 或较大的一种自定义形式。

操作步骤:

- A. 启动服务器程序和三个监控终端的程序
- B. 启动移动显示终端程序，向监控服务器进行注册
- C. 控制某个监控终端开始采集图像
- D. 调整采集速度、对比度、色度等采集参数
- E. 点击图像播放画面，对播放画面进行缩放

### 6.2.3 测试结果

✓ 固定显示终端:

通过有线接入方式可实现与多种类型的监控终端(WLAN、LAN、CDMA 1X)的连接,实现对多种监控终端的定位,能够在googlemap中进行移动轨迹显示,并记录监控终端的移动轨迹;可以实现对监控终端的图像相关参数的调整;图像的编码格式可以支持H.264,图像播放画面的大小可以是QCIF、CIF。如图6-7及6-8即为固定显示终端的播放图像和GPS轨迹显示图像。

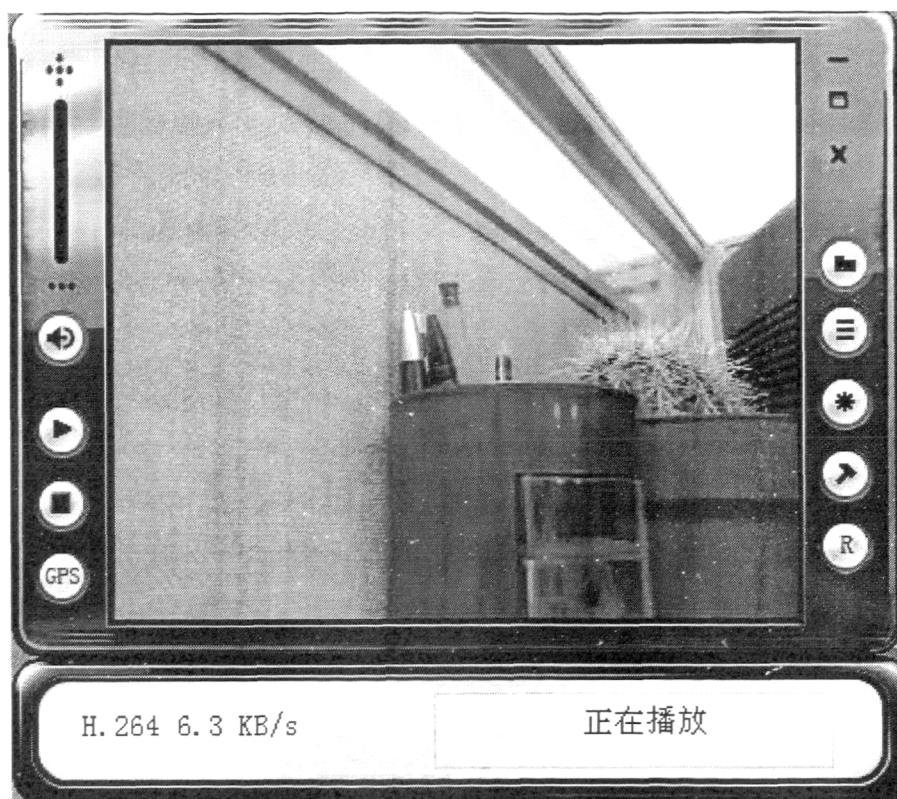


图 6-7 固定显示终端播放图

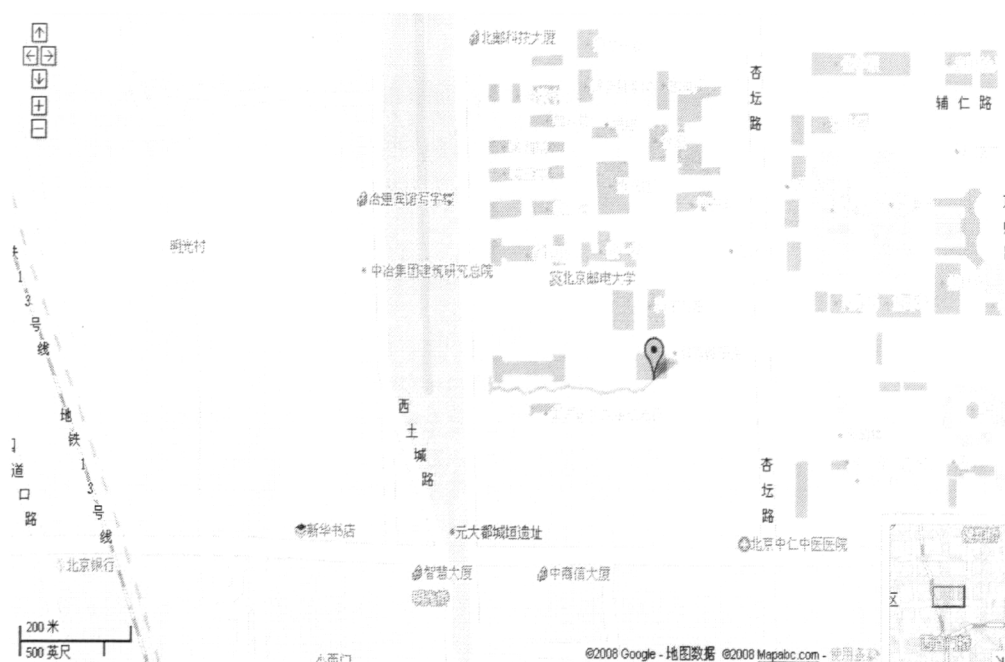


图 6-8 固定显示终端 GPS 轨迹显示

✓ 移动显示终端 (PDA):

通过 WLAN 无线接入方式可实现与多种类型的监控终端 (WLAN、LAN、CDMA 1X) 的连接, 实现对多种监控终端的定位, 可以实现对监控终端的图像相关参数的调整, 图像的编码格式可以支持 H.264, 图像播放画面的大小可以是 QCIF, 或者是屏幕所能呈现的最大尺寸。图 6-9 即为移动显示终端分别以 QCIF 及屏幕最大尺寸作为画面大小时的播放图。

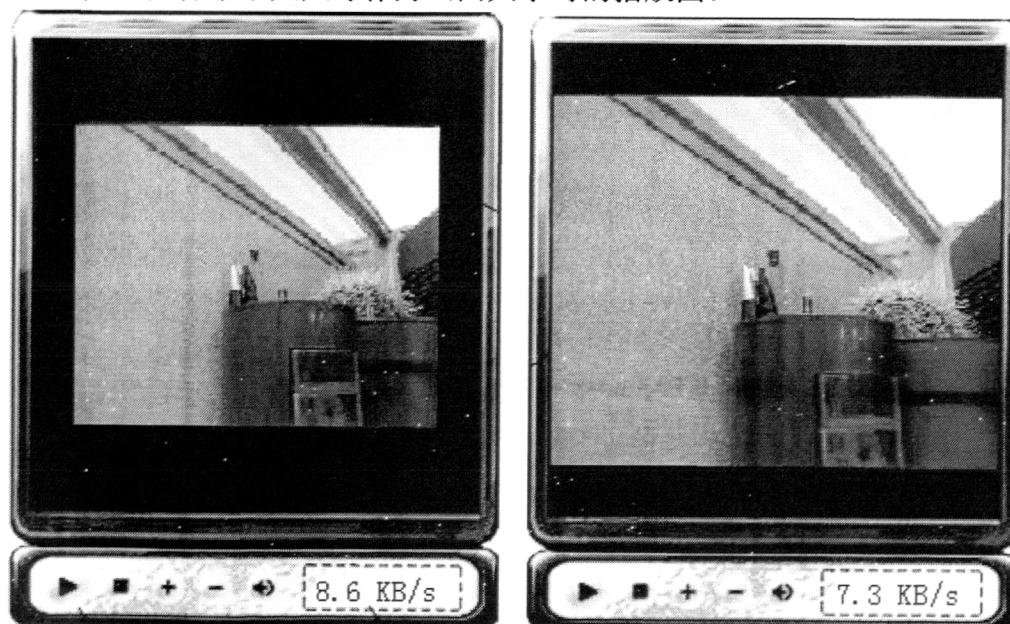


图 6-9 移动显示终端播放图

## 第七章 结束语

### 7.1 全文总结

移动流媒体是当前研究的热点，也是在下一代网络中有着良好发展前景的一项应用。本论文作为“移动视频监控系统的子课题，对其中的多功能显示终端子系统的设计和实现进行了研究，在相关理论研究、分析设计和实现验证上都做了大量的工作。

本文首先对视频监控的相关技术进行了介绍和分析，提出了移动视频监控系统的整体解决方案；然后详细介绍了多功能显示终端子系统，包括需求分析、详细设计和实现等内容；最后通过对系统测试结果的分析，对系统进行了验证，证明了系统的功能和性能达到了设计的要求。

### 7.2 不足和进一步工作

虽然多功能显示终端基本实现了既定的功能和目标，但是在测试过程中仍有一些不尽如人意的地方，比如出错处理还不够完善，虽然覆盖所有的出错点是不可能的，这就如同找到程序中所有的问题一样不可能，但是却可以尽量地完善出错处理，增加系统的容错能力。测试中还有不足的是内存泄露问题还须解决，尽管查找泄露点有一定的难度，但这始终是一个不小的问题。

### 7.3 研究生期间的工作

在硕士研究生期间，本文作者参加了移动视频监控系统的开发。

- 时间：2005.9 至 2007.12，显示终端组组长，主要的研究、设计和开发人员；
- 主持完成多功能显示终端子系统的需求分析和总体设计；
- 多功能显示终端子系统的详细设计和编码工作；
- 模块测试和系统联调；

另外，完成题为《一种基于反馈的视频流媒体自适应缓冲机制》的学术论文一篇，已被《高技术通讯》录用。



## 参考文献

- [1] J.Rosenberg , G. Camarillo , A. Johnston 等 . RFC3261 SIP :Session Initiation Protocol. IETF. 2002
- [2] Gonzalo Camarillo, 《SIP 揭秘》, 人民邮电出版社, 2003 年
- [3] H. Schulzrinne, S. Casner, R. Frederick 等, RFC3550: “RTP: Real-time Transport Protocol” , IETF, 2003
- [4] H. Schulzrinne, S. Casner, RFC3551: “RTCP: Real-time Transport Control Protocol” , IETF 2003
- [5] Gonzalo Camarillo, Miguel A.Garcia-Martin, 《3G IP 多媒体子系统》, 人民邮电出版社, 2006 年
- [6] 袁安存, 《全球定位系统 (GPS) 原理及应用》, 大连海事大学出版社, 1999 年
- [7] M.Handly, V. Jacobson. RFC2327 SDP:Session Description Protocol. IETF. 1998
- [8] 张龙翔, 《UML 与系统分析设计》, 人民邮电出版社, 2001 年
- [9] Qian Zhang, Wenwu Zhu, Ya-Qin Zhang, End-to-End Qos for Video Delivery Over Wireless Internet, IEEE Advances in Video Coding and Delivery, 2005, 93(1):123-133.
- [10]Supavadee Aeamvith, I-Ming Pao, Ming-Ting Sun, A Rate-Control Scheme for Video Transport over Wireless Channels, IEEE Transactions on Circuits and Systems for Video Technology, 2001, 11(5): 569-580.
- [11]Siu-Ping Chan, Ming-Ting Sun, Network Condition Detection for Video Transport over Wireless Internet, ISCAS 2006. IEEE International Symposium on Circuits and Systems, 2006, 3093-3096.
- [12]Hsiao-Chiang Chuang, ChingYao Huang, Tihao Chiang. A Novel Adaptive Video Playout Control for Video Streaming over Mobile Cellular Environment, ISCAS 2005. IEEE International Symposium on Circuits and Systems, 2005, 3267 - 3270 .
- [13]Sang-Ho Lee, Kyu-Young Whang, Yang-Sae Moon. Dynamic Buffer Allocation in Video-on-Demand Systems, IEEE Transactions on Knowledge and Data Engineering, November/December 2003, 15(6): 1535-1551.
- [14] Yih Huang, Genwen Xu, Leijun Huang, Adaptive AIMD Rate Control for Smooth Multimedia Streaming, ICCCN 2003. The 12th International Conference on 20-22 Oct 2003, 171 - 177.

## 致 谢

岁月匆匆，三年的硕士研究生生活即将结束，回首往事，感触颇多。

首先要感谢的是我的导师王文东教授，王老师从我入学开始就一丝不苟地严格要求我，循循善诱的教导我。他严谨的治学方法、渊博的知识、敏锐活跃的学术思想、豁达的处世态度都将使我受益终身。无论是在理论研究，还是动手实践，以至最终的论文整理，都倾注了王老师的大量心血。在此，谨向尊敬的导师王文东教授表示我最诚挚的谢意。

同时也衷心感谢阙喜戎老师和龚向阳老师在研究生期间给予的精心指导。感谢两位老师于我如自己学生般的关怀，为我提供了很多难得的学习机会，良好的实验环境，以及精湛的学术思想。

感谢与我一同奋战的项目组成员王锐，丁郁，郭永磊，郝瑞林，邓杨和王睿思等同学。感谢实验室所有的师兄师姐、师弟师妹们，他们给予我的热情帮助和指导，为我树立了学习和进步的榜样。

最后，要感谢我的父亲和母亲。多年来，他们含辛茹苦，付出了伟大的牺牲，支持着我的学业。亲情是我最大的精神支柱和前进动力，我的每一个进步都离不开他们的鼓励与支持。

再次感谢所有帮助我、关心我、爱护我的老师、同学、朋友和亲人们！