

# 基于 IP 组播的分布式数字音视频监控系统 的设计和实现

## Design and Implement of the Distributed Audio and Video digital Surveillance System Based on IP Multicast

周承芳 汪志兵 唐昆 金鑫 (清华大学 电子工程系 100084)

**摘要:**本文设计并实现了一个基于 IP Multicast 技术的远程数字音视频监控系统,提出了分布式音视频接入和共享的传输方案,能够有效地节省网络带宽。设计了“自适应切换单播通道和组播通道”的解决方案。

**关键词:**监控系统 分布式接入和共享 网络传输 IP 组播 Windows 套接字

### 1 引言

远程视频监控系统是近年来网络 IP 视频应用发展的一个新方向,涉及了计算机领域多种前沿技术。它可以借助网络从而突破距离上的障碍将监控信息从监控中心释放出来,在远程不同地点的分控中心或同一个分控中心可同时调看

### 2 系统结构

以下分别介绍系统的总体结构和软件结构。

#### 2.1 系统总体结构(见图 1)

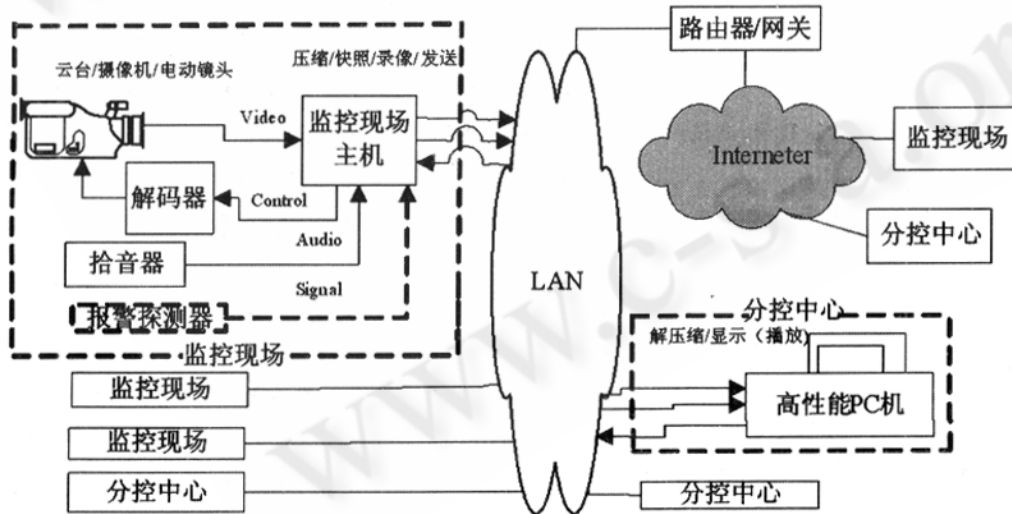


图 1 总体结构

远程监控系统一般包括三个部分:前端监控现场、通信设备和后端分控中心;

(1) 前端监控现场:由监控现场主机及一些外围设备组成;

(2) 通信设备:是指所采用的传输信道和相关设备;通信网络为 LAN 及 WAN;

(3) 后端设备:由若干分控中心计算机组成。

#### 2.2 系统软件

基于客户端/服务器(Client/Server)模式;前端监控主机运行 Server 程序;

后端分控中心运行 Client 程序。开发平台 Windows2000 /Windows XP,开发工具 VC++6.0

某一个或者几个监控现场的音视频数据,从而实现分布式的音视频接入和音视频数据共享。同时,可以与监控现场人员进行对讲;可以对远程监控现场的云台、摄像机等外围设备进行控制。

### 3 主要工作模块的设计与实现

系统的主要工作模块有:(前端)服务器端音(视)频处理

模块;抓拍、录像录音模块;云台摄像机控制模块;(后端)客户端音(视)频处理模块、网络传输模块,包括:音(视)频压缩包发送及控制数据接收模块(运行在 Server 端);音(视)频压缩包接受及控制数据发送模块(运行在 Client 端);

### 3.1 (前端)音视频处理模块

#### (1) 功能及工作流程

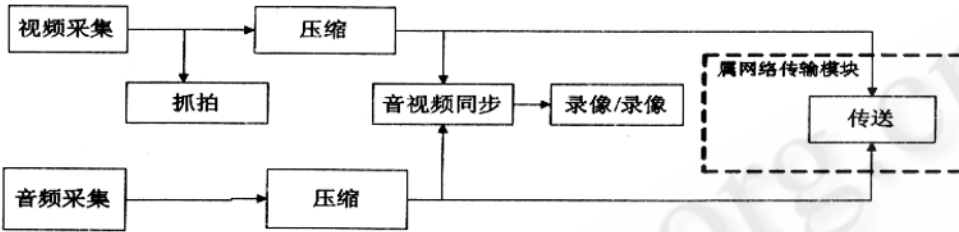


图 2 前端(Server)音(视)频处理模块

(2) 软件实现。音、视频处理模块:采集、压缩和打包;

① 采集、压缩:采用 BT848 芯片组视频采集卡或者 USB

接口的摄像头,完成视频的采集,支持多种采集格式。

② 打包:包格式(如图 3 所示)

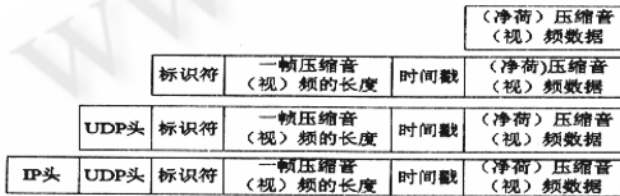


图 3 数据包结构

注:标识符:“FF”为视频包;“00”为音频包;

③ 传输:将打包的压缩音(视)频数据发送到写数线程中,该线程将数据写入发送缓冲区(见网络传输模块);

### 3.2 抓拍、录音录像模块

(1) 抓拍:对采集的原始视频数据可进行“自动”、“手动”单帧抓拍,以 BMP 格式的文件存储;

(2) 录像、录音:对压缩的音视频数据以 AVI 格式存储;其中在写入音频流时,进行了同步处理,程序中采用时间戳技术,对音视频数据进行标志。在通常情况下,音视频的同步问题的处理上,往往以音频数据为准,因为人们对于音频数据的敏感性要超过视频数据的敏感性,这样可以保证听觉效果的连贯性。考虑到视频监控系统的应特点,视频图像信号所包含的信息较之音频来讲更加重要,所以实际中以视频数据为基准,所谓时间戳技术,即是在打包时给每个音视频数据流加入时间标记,在写入音频压缩包到 AVI 文

件之前,先计算音频包与视频包的时间差,对于时间标记相差较大的数据予以舍弃,反之,将该音频包写入 AVI 文件。以保证最后的音视频同步。实践检验这样的处理方法对于音视频的同步改善作用明显。

录像录音的方式:按预设的时间段,进行录像录音;循环录像;动态检测录像;报警启动录像录音。录像录音的方式设置可以在本地/远程监控端进行设置;

### 3.3 (后端)音视频处理模块

#### (1) 功能

(2) 软件实现。封装类 CMonitorDlg(解码及播放管理类),完成对接收到的压缩音视频包进行解压缩,显示视频、播放音频;该类的中解码函数 DecodeOneFrame()与播放(显示)

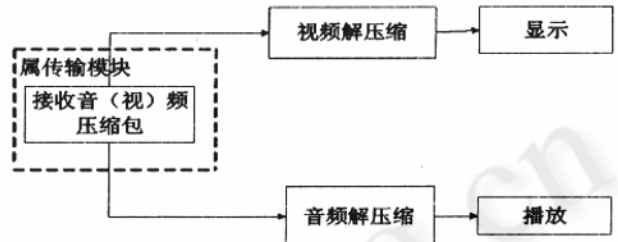


图 4 后端(Client)音(视)频处理模块

函数 DisplayCurrentFrame()调用与网络传输模块之一的接收模块密切相关,故该模块的调用时序放到网络传输模块之一的接收模块中阐述。

### 3.4 云台及摄像机控制模块

(1) 功能描述。运行在前端监控主机上;云台用于实时监控摄像头方便地满足监控者对于监控位置、图像质量的要求。在系统中,前端主机的串口发送控制指令到与云台相连接的解码器,再由解码器控制相应的继电器达到各个电源开关的闭合,从而实现云台的各项控制功能。

(2) 分析与设计。云台镜头控制系统的功能按控制方式可以分为 2 类:动作控制和状态控制(开关)控制;动作控制(云台动作、镜头动作);状态(开关)控制(云台状态)。

本系统中云台运动动作方向有 4 个(上、下、左和右),状态(开关)有一个:自动;镜头控制动作有六个(光圈、焦距和变焦);按照解码器的数据协议将各动作及状态进行数据编码,通过计算机的串口向解码器发送指令,当解码器收到指令后,正确解析并控制继电器的断开/闭合,从而达到最终控制的目的。

(3) 编程: Windows 下写串口有多种方式, 本系统利用 Win API 通信函数

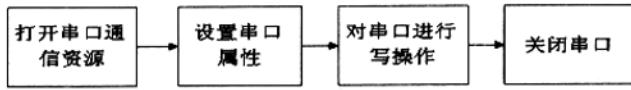


图 5 串口通信的基本步骤

castDataThread)(考虑到网络中路由器等因素, 可能不支持或者没有打开组播功能, 如出现这样的情况就自适应地再开

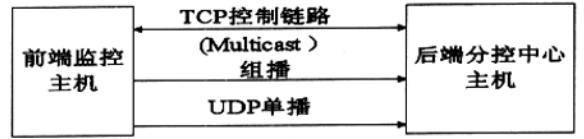


图 6 信道描述

### 3.5 网络传输模块

(1) 系统中传输数据的特点及通信协议的选择。系统中传输的数据有控制数据: 云台电动镜头的控制指令; 录像参数、系统参数; 控制数据要求做到准确无误传输; 音(视)频数据: 是连续媒体, 数据量大, 允许传输中存在一定的数据丢失率, 但实时性要求高, 此外, 监控系统中, 要实现音视频的分布式接入和数据共享, 必须进行音视频的多点传输。

(2) 怎样实现上述目标。首先是通信协议的选择, 采用 TCP/IP 协议; 根据传输数据的不同特点, 采用面向连接的、提供可靠服务的 TCP 协议传输控制数据; 采用无连接的、高效的且支持组播的 UDP 协议传输音(视)频数据。

(3) IP Multicast(组播): UDP 的 IP 传输方式有单播、广播及组播; 比较三者: 单播: 一对一的传输方式; 效率低; 广播: 一对多的传输方式; 增加网络负担; 组播: 一对多的传输方式; 但只传输给加入组的主机; 选用组播方式, 实现音(视)频数据的多点传输。

(4) 实现组播传输, 则组播源和接收者以及两者之间的下层网络都必须支持组播。这包括以下几个方面: 主机的 TCP/IP 实现支持发送和接收 IP 组播; 主机的网络接口支持组播; 有一套用于加入、离开、查询的组管理协议, 即 IGMP(v1, v2); 有一套 IP 地址分配策略, 并能将第三层 IP 组播地址映射到第二层 MAC 地址; 支持组播的应用软件; 所有介于组播源和接收者之间的路由器、集线器、交换机、TCP/IP 协议栈、防火墙均需支持组播。考虑到上述因素, 在网络传输音视频时, 若出现不支持组播的情形自适应地开单播通道。

#### (5) 数据通道

- TCP/IP 信令通道: 用于传输控制指令;
- Multicast/IP 数据通道: 用于多路传输压缩音视频;
- UDP/IP 数据通道: 用于单路传输音视频

(6) 线程安排、缓冲区规划和读写同步。在 Server 端, 当控制中心建立第一对 TCP 连接时打开一个写数线程(CWriteThread)(无论有多少对 TCP 连接, 均只启动一个写数线程), 将编码线程(CEncodeThread)输出的音视频压缩数据(包)写入发送缓冲区; 开组播数据发送线程(CSendMulti-

一个单播线程(CSendUnicastDataThread)), 在组播线程中开组播通道(在单播线程中开单播通道), 发送组播数据(单播数据, 如有多个单播通道, 采用轮询的方式进行单播); 发送缓冲区规划: 开循环缓冲区存放编码器输出的压缩音视频数据包; 封装了类 CCycleBuffer 对循环缓冲区进行管理; 线程同步: 对循环缓冲区进行访问的线程有三个: CWriteThread: 对循环缓冲区进行写(Write)操作; CSendMulticast: 对缓冲区进行读(Read)操作; CSendUnicast: 对缓冲区进行读(Read)操作; 这样就存在一个写线程二个读线程的读写同步问题; 通过控制变量来记录读写位置, WriteIndex 记录写位置; ReadIndex1(ReadIndex2)记录读位置, 变量 ReadIndex1(ReadIndex2)和 WriteIndex 形成了对缓冲区读写的控制; 对这三个变量的操作通过互斥量(Mutex)来保护。

在 Client 端, 接收压缩的音(视)频数据包, 呼叫要进行监控的前端主机, 建立 TCP 连接, 每建立一对 TCP 连接, 就启动一个接收线程(CSocketThread)、建立一个 CMonitorDlg(解码及播放管理类)的实例(一个分控中心, 最大容量是 12 个连接(对应 12 个监控点), 当满设计容量时, 最多 12 个接收线程(CSocketThread), 12 个 CMonitorDlg(解码及播放类)的实例)。

### 3.6 主要工作模块的程序流程

发送端(监控现场主机、服务器端): 在主线程中, 启动音频采集线程、视频采集线程及音频编码线程、视频编码线程, 分别对音频及视频进行采集, 放入音(视)频缓冲区中, 在本地回放视频, 将采集的音(视)频数据发送到编码线程中进行压缩编码, 此时可对压缩音(视)频进行录像及录音; 同时监听分控中心的连接请求, 收到连接请求, 建立 TCP 连接(信令通道); 通过信令通道, 向分控中心发送二组组播地址及端口号(对应音频、视频, 给每个监控点分配一对组播地址及端口号); 启动写数线程(WriteThread), 启动组播送数线程(MulticastDataThread)(自适应地开单播线程 UnicastDataThread), 在 MulticastDataThread 线程中, 调用类 CMulticast, 建立组播通道(在 UnicastDataThread 中, 调用类 CUdpSocket 建立单播通道), 将压缩的音视频数据包组播

(单播)发送到分控中心。

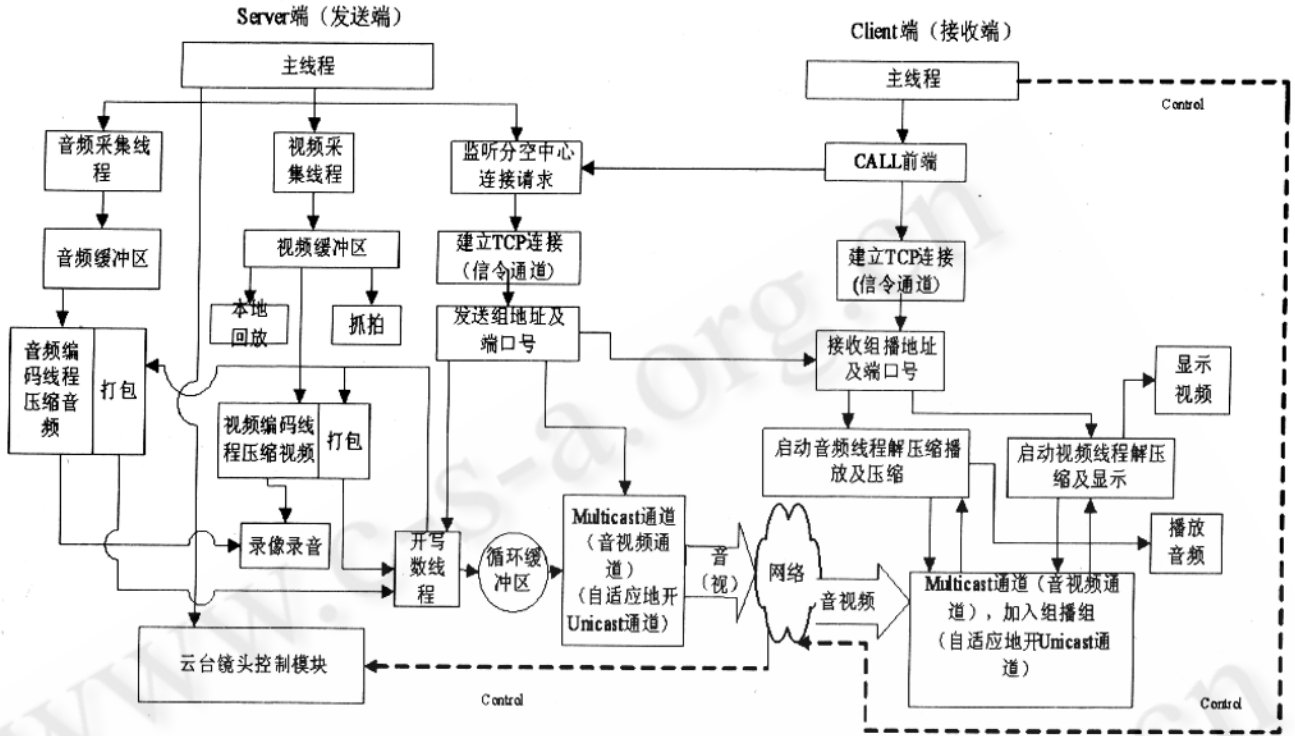


图 7 主要模块工作流程

接收端(分控中心、客户端):分控中心主机运行客户端程序,在主线程中:向前端监控现场主机发出连接请求(CALL),建立TCP连接(信令通道);后端接收到组播地址及端口号后,启动音(视)频接收线程(CSocketThread)完成:调用CMulticast(CUdpSocket),建立组播通道(单播通道);接收音(视)频压缩包,并解码显示(播放)。

一个后端分控中心可同时监控12路视频及音频信号,在设计客户端监控程序时,采用多线程技术,每建立一对前端监控主机与后端分控中心(服务器)的连接,就开一个接收线程(CSocketThread),建立一个CMonitorDlg(编解码及播放管理类)的实例。在接收线程中根据IP地址不同区分来自各个前端的音视频数据包,调用类CMonitorDlg中的函数DecodeOneFrame()及DisplayOneFrame()对压缩的音视频包进行解压缩及显示视频(播放)音频;对云台及摄像机的控制指令通过信令通道传输,设置二级权限,Administrator(主控)只有一个,在发送控制指令时,高一等级权限的用户接管下级用户;同级用户按照时间优先来分配控制权。

### 5 结论

本文给出了基于IP Multicast的分布式音视频接入和共

享的监控系统的模型和主要功能模块的详细实现。该系统具有结构简单、规模扩展性好、资源利用率高等明显优点。在该系统中采用的技术也可应用到相似的系统中去,诸如远程教学、远程会议、VOD。

### 参考文献

- 1 陈龙, 电视监控与安全防范系统, 科学出版社, 1999.
- 2 李现勇 编著, Visual C++ 串口通信技术与工程实践, 人民邮电出版社, 2002.
- 3 徐立新、李宗斌, IP组播技术的API编程实践, 微型机与应用, 2002. 6. P. 32\_34.
- 3 蒋东兴等, Windows Sockets 网络编程设计大全, 清华大学出版社, 1999.
- 4 W Richard Stevens. TCP/IP 详解 卷1: 协议 机械工业出版社.
- 5 刘毓敏编著, 数字音视频技术与应用, 电子工业出版社, 2002.
- 6 Dvid J. Kruglinski 著, 潘爱民、王国印译, Visual C++ 技术内幕 (第四版), 清华大学出版社, 2003年.