

摘要

随着网络技术的发展,数字化、网络化的视频监控系统使人们可以通过网络不受限制地对重要场所实施监控,为安防领域注入新的活力。基于无线局域网的网络监控,常用于小范围的监控,例如生产车间、居民小区。监控系统由支持 IEEE802.11a/b/g 标准的无线设备构成,连接速率可达 11、22、54Mbps 甚至更高。基于无线局域网的监控系统通常能够保证视频传输所需的带宽,监控系统性能可靠、稳定。同有线网络监控系统相比,无线局域网监控系统没有布线的烦恼,不会破坏原有环境设施,施工周期短,性价比高,且便于扩充调整,能够很方便地增加或者减少监测点。

本文首先介绍了移动视频监控系统中所采用的关键技术,然后详细分析了系统所要实现的目标,从而确定了系统实现要采用的技术路线。此系统由视频采集端,服务器端,无线网络,移动设备端组成。系统在视频采集端利用摄像头和采集卡实现对所捕获视频的采集并且进行相应的 A/D 转换;在视频服务器端,视频数据采用 DirectShow 技术经运动检测、MPEG-4 压缩处理后,采用 Windows Media Encoder 技术再压缩成 WMV 格式的流媒体,并封装成 RTP 数据包以广播的方式发送到基于 IEEE802.11g 体系的 WLAN 客户端;WLAN 中的客户端采用 Windows CE 系列操作系统的手持移动设备,移动设备负责接收和解码播放发送来的视频数据,并在接收过程中利用缓冲技术来缓解网络延迟和抖动带来的影响。视频数据发送端和接收端均采用多线程技术实现。

论文所述的系统,需要完成服务器端和客户端的系统开发。其中在服务器端需要利用微软公司的 DirectShow 和 Windows Media Encoder 进行开发,同时借助少量 Socket 编程,着重研究 IEEE802.11g 无线网络广播传输以及 MPEG-4 和 WMV 编码技术。并且采用相邻帧差法对视频监控图像进行运动检测。在客户端需要使用 .net compact frameworks 编程技术实现 Windows CE 平台下的手持设备中视频播放功能。

关键词: IEEE802.11g, WME, Pocket PC 视频监控, DirectShow, 广播

Abstract

With the development of the network technology, digitalized and networking video monitor system enables the surveillance of the important place through the network without any limitation, which brings vitality to the field of the safety and protection. Network surveillance based on the wireless LAN is normal used in small area monitor, e.g. production department, residence community. Composed of wireless facilities supporting IEEE802.11a/b/g standard, the connection rate of the surveillance system can reach as far as 11, 22, 54Mbps or more. The system based on wireless LAN can guarantee the bandwidth for video transmission with reliable and stable performance. Compared with the cable surveillance system, the wireless system saves the work of wiring, brings no destruction to the environment, which cuts down sharply the construction period and improves the cost-benefit. It is convenient to make adjustment to the arrangement of monitoring points.

This thesis firstly introduces the key technology utilized in the mobile video surveillance system, then analyzes in detail the objective of the system, and determines the technology pathway. This system is composed of video collection component, server component, wireless network and mobile facilities. Cameras and video capture cards are adopted to collect the video and perform the A/D transformation. At server side the video data is encoded into stream media using Windows Media Encoder technology after the motion detection and MPEG-4 compression, and encapsulated into RTP data package and broadcasts to the WLAN client side based on the IEEE802.11g architecture. The client components in WLAN use Windows CE based hand-hold mobile equipment. The mobile equipments are responsible for the receiving, decoding and broadcasting of the video data. Double buffering technology is adopted during the receiving process to eliminate the effects of network delay and tremble. Receiving and transmitting of the video data are realized through the multi-threading technology.

Development of the system described in the thesis involves Microsoft

DirectShow and Windows Media Encoder technology, Socket programming, and especially focuses on the IEEE802.11g wireless network broadcasting and MPEG-4 and WMV coding technology. A frame differencing algorithm is accomplished to detect moving objects. Client environment based on Windows CE operation system, the thesis should use .net compact frameworks to accomplish the function of client.

keywords: IEEE802.11g, WME, Pocket PC surveillance, DirectShow , Broadcast

独创性声明

本人声明，所提交的论文是本人在导师指导下进行的研究工作及取得的科研成果。尽我所知，除了文中特别加以标注和致谢的地方外，论文中不包含其他人已经发表或撰写过的研究成果，也不包含为获得武汉理工大学或其它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。

签名：苏焕 日期：2009.5.15

学位论文使用授权书

本人完全了解武汉理工大学有关保留、使用学位论文的规定，即：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电子版，允许论文被查阅和借阅。本人授权武汉理工大学可以将本学位论文的全部内容编入有关数据库进行检索，可以采用影印、缩印或其他复制手段保存或汇编本学位论文。同时授权经武汉理工大学认可的国家有关机构或论文数据库使用或收录本学位论文，并向社会公众提供信息服务。

(保密的论文在解密后应遵守此规定)

研究生(签名)：苏焕 导师(签名)：徐东平 日期：2009.5.15

第1章 绪 论

1.1 课题的研究背景

当今社会是一个高度复杂的社会，越来越多的突发事件和异常事件的发生严重威胁到人们的日常生产和生活的安全。为了增强民众的安全感，一个现代化城市必须具备良好的公共安全性，而稳定的社会治安环境和有序的工作生活环境则是这种安全性的重要体现。

随着网络技术的发展，数字化、网络化的视频监控系统使人们可以通过网络不受限制地对重要场所实施监视和相关控制。它可以通过网络把分散的信息集中起来，实时显示并存储，为监控人员提供实时、直观的视觉材料，从而便于统一管理，也可以在分散的、危险的工作环境中实现无人职守。但是近年来，传统的数字有线监控的市场已经相对饱和，与此同时，无线局域网却获得了广泛的应用与长足的发展。当今国内，教育科研、医疗卫生等领域及制造业、仓储业等各行各业都在广泛应用无线网络。越来越多的公司、企业为提高竞争力，都在争相采用无线网络，以方便移动性的网络接入、提高工作效率、降低长期的投资成本。无线网络是利用电磁波在空气中发送和接受数据，而无需线缆介质连接形式的网络^[1]。它具有传统有线网络无可比拟的优势：(1)灵活性。在有线网络中，网络设备的安放位置受网络位置以及地理位置的限制。而无线局域网只要在无线信号覆盖区域内，任何一个位置都可以接入网络。无线局域网另一个优点在其移动性。连接到无线局域网的用户可以在信号覆盖范围内任意移动且能同时与网络保持连接。(2)安装简易。无线局域网可以免去或最大程度地减少网络布线的工作量，一般只要安装一个或几个接入点设备，就可以建立覆盖整个区域的局域网络。(3)容易维护。有线网络如果出现了由于线路连接不良而造成的网络故障。往往很难检查，而且检修线路需要付出很大的费用。无线网络则容易定位故障。只需要更换设备即可恢复网络连接。(4)易于扩展。无线局域网有多种配置方式。可以很快从只有几个用户的小型局域网扩展到上千用户的大型网络。并且能够提供节点之间诸如“漫游”等有限网络无法实现的特性。

随着无线网络的大发展，个人便携式移动设备的市场也开始繁荣起来，作为无线网络监控系统的终端，拥有高分辨率，大屏幕，强劲性能的 Pocket PC^[2]

(简称 PPC)也在近些年来凸显其价值。通过面向市场的技术开发和大规模集团化的工业生产,PPC 的体积愈发小巧,而性能愈发强劲,售价也从多年前的高不可攀变成了现在可以集中采集的低廉设备。Windows Mobile 的使用,更使得 PPC 真正成为一个可开发的嵌入式设备。Windows Mobile 系统自带 MPEG 视频解码器,依靠 PPC 强劲的硬件基础可以十分的流畅清晰的播放多媒体数字视频流,同时,易操控的用户界面,给用户特别是非技术人员带来了极大的方便。

据上所示,具有数字化、无线化和便携化三种特征相结合的新一代数字视频监控技术已经成为了视频监控系统发展的新趋势,这使得安防人员在移动中进行监控的愿望变为了可能。它正在获得越来越多的关注,将逐步取代传统的视频监控技术,在社会治安监控系统中发挥不可替代的作用。

1.2 课题的研究意义

基于局域网的网络监控,常用于小范围的监控,例如生产车间、居民小区。可以分为有线和无线局域网。前者通常由 10/100M 以太网、同轴电缆等构成,后者由支持 IEEE802.11a/b/g 协议^[3]的无线设备以及无线桥接器构成,通讯速度可达 11、22、54Mbps 或者更高。

传统的有线视频监控系统,虽然拥有传输图像稳定,视频图像质量高等优点,但是其缺点却显而易见:高额的安装费用和较低的灵活,同时安防人员使用有线监控系统时,必须身处监视器旁边,无法实现高效的快速反应能力。同时基于 IEEE802.11 协议的无线网络传输的是数字信号,这对于视频流的存储,回放,放大,甚至对关键区域进行提取都是十分便捷和合适的。

其次,在我国经济社会发展进入新的历史阶段,党和政府明确提出建设节约型社会,就是要在社会生产、建设、流通、消费的各个领域,在经济和社会发展的各个方面,切实保护和合理利用各种资源,提高资源利用效率,以尽可能少的资源消耗获得最大的经济效益和社会效益。这是关系到我国经济社会发展和中华民族兴衰,具有全局性和战略性的重大决策。

相对于传统有线网络的搭建,无线网络明显经济有其价格优势。资料显示:建设同样目标节点的有线网络比无线网络要节约至少 20%的原始采购费用,后期更是节约 40%的能耗和人员开支^[4]。

在终端设备的选择上,相对于传统的显示器,新一代移动终端 PPC 同样由其技术优势和价格优势。2004 年开始,随着 PPC 生产厂商的繁荣,加之技术更新,标准统一,原来定位于高端的掌上设备开始变为普通人群可以接受的大众化移动通讯工具。2008 年底美国调查研究公司 Gartner 发布的调查报告显示:2008 年第二季度全球智能移动终端的总体销售量达到了 3220 万台,相比去年同期增长了 15.7%,在全球手机整体销售成绩中,智能手机和移动终端的份额继续稳定在 11%上。与此同时,智能终端的价格相对于去年平均下降了 37.6%。所以综合考虑,使用无线视频加移动监控端这一系列配置正是我国建设节约型社会所应该大力提倡的。

第三,无线监控系统有着广阔的市场前景,据诺达咨询公司发布的《2007 网络视频监控业务研究报告》显示:2006 年中国网络视频监控市场规模达 18.24 亿元,2007 年中国网络视频监控市场规模将达到 26.36 亿元,未来 5 年内,网络视频监控将保持约 38%的年增长率,预计 2011 年将达 86.61 亿元,而且会有越来越多的网络视频监控系统采用完全数字化技术,特别是在银行、交通、工业、零售业等行业市场。以数字化视频监控为基础的智能视频监控技术将获得长足发展。同时,中华人民共和国公安部 2004 年在全国范围内确定了 22 个城市作为首批科技强警示范城市。2005 年确定中西部地区的 15 个城市和江苏、浙江、山东、广东四省的 23 个城市作为第二批示范城市。2007 至 2009 年,则为第三批示范城市建设的时期,预计到 2008 年,科技强警示范城市将达到 180 个,全国约有 200 万个监控摄像头用于城市监控与报警系统^[5]。这一切突出的反映出视频监控这一关系到国计民生、安居乐业的高科技市场终于在 21 世纪迎来全球目光的关注,随着社会的需要、政府的推动,世界各国工业经济、旅游经济、文化体育事业的互融促使视频监控技术的应用领域越来越多元化,而无线网络视频监控这一高新技术的突破也势必推动全球的繁荣与稳定。

1.3 无线视频监控应用的国内外发展现状

近几年,国内的长城集团、网通电子商务有限公司、北京微电子技术有限责任公司等也自主开发了国产的无线数字视频监控系统。而且,我国有许多高校和科研机构也正在从事与数字视频监控相关的技术研究和系统开发。中国科学院自动化研究所模式识别国家重点实验室视觉监控组处于领先地位,在此研

究基础上已有一批可以实用化的成果。但就目前先进的数字视频监控系统而言，在视频压缩、分析、传输、存储和分级控制等方面仍然有待不断提高和完善，智能化方面更是处于研究探索过程的初级阶段，将是未来监控系统行业发展的趋势所在^[6]。

由于各种原因，我国大陆的安防产业较发达国家晚二三十年。但是近年来，随着我国经济的快速发展、人民物质生活水平的提高和消费观念的改变，安防从过去提倡人防发展为以技防为主、人防为辅，并成为现代管理的重要手段。截至到 2004 年底，安防产业市场规模 400 多亿元，监控约占一半，近年来一直以每年 15%至 30%的速率快速增长，据有关部门预计，2010 年全国监控系统的市场总额将达到 1000 亿元，我国的监控行业正面临着良好的发展机遇。但总的来讲，由于起步晚，技术较落后，使的我们在该行业既有广阔的发展空间，又面临着巨大挑战^[7]。

在国际上发达国家均采用视频捕获的方式来监控公共场所中人们的一举一动。其中以英国为最：英国人口数量仅占全球人口总数的 1%，但监控摄像头数量竟占全球 20%。伦敦居民人均每天受到 300 个摄像头监控。英国如今设有 4 20 万个闭路电视监控摄像头，数量居全球第五位，平均每 1 4 人就有一个。其中通过无线网络传输视频图像的摄像头占到总数的 15%。英国创建的“监控型城市”有效的打击了犯罪，威慑了罪犯，构建了和谐。

随着 3G 技术的逐步应用，我国无线视频监控技术也将逐步升级，从原来的依靠有线网络传输向依靠 3G 网络和无线 WAN 网络技术进化。其中 WCDMA 和 WLAN 成为了无线移动视频传输的成熟网络标准^[8]。依靠 WLAN 网络技术产生的特种监控系统在 2007 年中国国际警用装备博览会上大放异彩，迅速在国际国内产生了较大的影响，引起了越来越多国内外反恐装备研发机构和生产厂商的关注。

同年，上海电信在杨浦区、闵行区就推进“无线城市”建设签署合作框架协议，到年底，上海“无线城市”网络将建成 3000 个无线热点覆盖，在上海主要商务楼、高星级酒店、中小型商务谈判场所、南京路步行街等商业休闲区域实现无线宽带覆盖。今年年底 Wi-Fi(基于 IEEE802.11 标准的无线宽带的品牌)热点覆盖建设完成之后，在热点地区，用户只需一款带 Wi-Fi 无线网卡的终端再加一个无线宽带上网账号，即可实现宽带接入，可享受到 Internet 接入、无线视频监控、无线办公、公众用户无线上网等应用。据 Wi-Fi 联盟预测，2008 年中国 Wi-Fi 应

用市场总值将达 12 亿美元，中国的无线网络市场正处于高速增长的阶段。除上海以外，中国电信还在江苏、浙江、湖北、福建、四川等 7 个省份建设 Wi-Fi 网络，目前中国电信的 Wi-Fi 业务已有近 800 万户，同时在公众场所布置了 1 万多个网点。而香港地区的 Wi-Fi 网络建设则走在了其它城市的前列。香港特区政府去年预算 2.1 亿港元建设 Wi-Fi 网络，今年一二月份，香港新增公众 Wi-Fi 热点超过 1000 个，其总数达到 6400 多个，预计 2009 年年中将增至 8000 个，涉及公共图书馆、体育场馆、文化康乐中心、就业中心、社区会堂、大型公园和政府大楼等地点^[9]。

与此同时，我国政府和谐社会战略决策的实施，“平安城市、平安奥运”等各项安保工作的展开，也给监控安保行业带来了迅猛发展和勃然生机。

1.4 论文的组织结构

本文共六章，下面将分别介绍各章主要内容。

第 1 章绪论。首先阐述了无线网络以及视频监控系统的研究背景和意义；然后分析了所用技术的国内外发展现状；最后介绍了论文的组织结构概况。

第 2 章无线视频监控系统总体设计。首先给出系统所要实现的目标和框架结构；由此基础后完成对系统开发平台的选择、工具的选择、结构的选择等。最后给出了系统实现的技术路线。

第 3 章无线视频监控系统服务器端的设计与实现。首先给出了服务器端的系统构建模型，然后根据各个功能模块分别介绍了模块实现所用技术和程序架构框架。接着详细完成对视频采集模块和视频流媒体编码模块的构建，进而分析了几种多媒体视频编码格式和视频的广播发送原理。

第 4 章无线监控系统的网络通信。本章是将系统的服务器端和客户端之间的网络通信部分独立出来进行介绍。重点介绍了基于 IEEE802.11/g 协议的无线局域网来进行视频的广播发布，进而给出了本系统无线网络的构建模型和实现过程，最后介绍了多接入点无线网络切换技术。

第 5 章移动监控客户端的设计与实现。首先介绍了客户端的工作流程；然后对 Pocket PC 的功能和发展进行详细介绍。着重介绍了基于 WinCE 的 Windows Mobile 5.0 操作系统，以及运用 compact frameworks 2.0 开发移动客户端程序，使用 Windows Encoder 对视频流进行解码等技术。

第 6 章视频监控的智能化技术。在这一章，主要介绍了静止背景情况下的运动检测技术。重点分析并研究了基于相邻帧差法的运动检测技术。

第 7 章总结与展望。对本文所做工作进行总结，同时针对系统的不足之处，提出了一些对今后需要继续完善的研究建议。

第2章 无线视频监控系统总体设计

2.1 系统分析

视频监控系统的产生和发展主要有多个方面的原因：首先是人们的安全防范意识持续加强；其次是视频技术和网络通信技术不断发展，日益成熟。监控系统的发展大致经历了三个阶段^[10]。在九十年代初以前，主要是以模拟设备为主的闭路电视监控系统，称为第一代模拟监控系统。九十年代中期，随着计算机处理能力的提高和视频技术的发展，人们利用计算机的高速数据处理能力进行视频的采集和处理，利用显示器的高分辨率实现图像的多画面显示，从而大大提高了图像质量，这种基于 PC 机的多媒体主控台系统称为第二代数字化本地视频监控系统。九十年代末，随着网络带宽、计算机处理能力和存储容量的快速提高，以及各种实用视频处理技术的出现，视频监控步入了全数字化的网络时代，称为第三代远程视频监控系统。第三代视频监控系统以网络为传输基础，以数字视频的压缩、传输、存储和播放为核心，以智能实用的图像分析为特色，使监控技术的发展达到了一个新的高度，并引发了视频监控行业的技术革命。随着无线网络火热地发展，曾经需要有线电缆承载的视频传输媒介可以转变成成为对终端发射的无线网络信号。相对于有线监控系统，无线监控系统受环境的影响比较小，灵活性很高。

本文所提出的无线监控系统采用 WLAN 作为多媒体流的传输平台，是一种对静止背景下进行安全监控的解决方案，同时，系统使用便携式移动终端，这样就使管理人员可以在巡查中对监控系统的各个摄像头进行监控和查看，方便安全监控人员的工作，从而达到更迅捷地进行事件排查和对突发事件进行快速反应。在设计本系统时，本着架构合理、经济实用的方针，采用先进合理的技术来实现系统，使系统能够平稳运行并具有良好的可扩展性；在满足系统需求的情况下，尽量降低系统成本，提高其性价比。由于本系统的特殊性，有以下三个方面需要重点考虑。

首先，要实现的目标是能在远程端实现对监控现场的实时监控或对历史监控记录的查询播放。因此，系统需要在监控现场连续采集到清晰的监控图像，并能对监控图像数据进行持续存储；然后将这些实时的监控图像或记录通过网

络传输到远程端，并在传输过程中附带简单的网络流量控制功能。这就需要服务器端对摄像头采集而来的视频图像采用适当的多媒体视频算法进行压缩和存储。

其次，使用 WLAN 作为系统的网络承载，其网络带宽和速率受到诸多方面的限制。这便要求服务器更加注重视频的编码效率，采用何种有效的视频编码格式进行网络传输是实时监控系統研发的重点。同时，无线网络传输因其速率和收发质量的不同而产生众多的协议，合理的选择网络传输协议也是系统可以最终架构成功交付使用的关键。

第三，本系统为了实现安全人员在移动中进行监控，监控终端使用了便携式的掌上设备。当前业界所使用的主流掌上设备包括 PocketPC, Palm, SmartPhone 等，其中，PocketPC 依靠其强劲的硬件，小巧的外观，开放的嵌入式操作系统，成为本监控系统的首选终端设备。但是相对于服务器端，PocketPC 的硬件仍然较难处理高码率、高分辨率的视频流^[11]。所以，客户端监控设备需要一个简洁高效的解码器和播放器。这也是系统能否真正应用于实际生活的关键点。

同时，无线信号发射点的选择，不同无线网络间的切换和加密也都是系统在实际架设中所需要考虑的。

2.2 总体设计

2.2.1 系统的软件应用结构

在现代软件工程的发展过程中，计算机软件应用结构的演变经历了三个阶段：集中式或终端式结构、客户机/服务器(C/S)结构和浏览器/服务器(B/S)结构^[12]。

由于本系统是需要有网络承载的通讯系统，所以在软件结构模式的选择上，单层结构不适合网络应用或者分布式应用。从而两层结构的 C/S 模式和三层结构的 B/S 模式应运而生。B/S 采用了逻辑上的三层结构，而在物理上的网络结构仍然是原来的以太网或环形网。这样，第一层与第二层结构之间的通信、第二层与第三层结构之间的通信都需占用同一条网络线路。由于 C/S 在逻辑结构上比 B/S 少一层，降低了网络通信量，对于相同的任务，C/S 完成的速度要比

B/S 快，因此 C/S 更利于处理大量数据。C/S 模式将任务分离，客户机和服务器各司其职，简化了应用，使得系统可靠性好，便于应用程序的移植^[13]。

不同的监控系统所采用的 C/S 系统结构和实现方式均有所不同，系统实现的主要功能基本相同。常用的监控系统都包括监控现场和远程监控中心两部分，两者通过计算机网络进行通信。

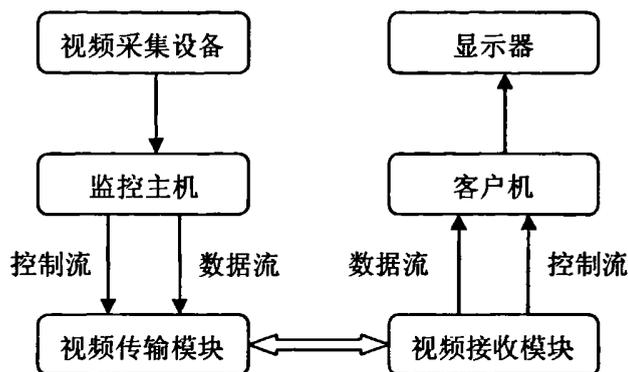


图 2-1 视频监控系统体系结构

如图 2-1，使用典型的 socket 客户机/服务器结构，客户端和服务端之间运用网络协议（IEEE802.11）来进行通信。加之 C/S 模式对数据处理简易迅捷的特点，相信系统的传输复杂度和传输速率都会达到令人满意的程度。

2.2.2 系统的网络拓扑结构

但凡多节点或多终端的计算机网络系统均需要选择恰当的网络拓扑结构。常见的网络拓扑结构包括星形、环形、总线、网状等。由于本系统采用的主机-客户机式的功能设计，加之客户机处理实时数据能力有限，所以系统采用星形网络拓扑结构：它以一台中心处理机（通信设备）为主而构成，其它入网机器仅与该中心处理机之间有直接的物理链路，中心处理机采用分时或轮询的方法为入网机器服务，所有的数据必须经过中心处理机。星型网络几乎是 Ethernet（以太网）网络专用，这种拓扑结构网络的基本特点主要有如下几点：

(1) 容易实现：它所采用的传输介质一般都是采用通用的双绞线或者是本文使用的无线射频信号，这种传输介质相对来说比较便宜。

(2) 节点扩展、移动方便：节点扩展时只需要从集线器或交换机等集中设

备中拉一条线即可，而要移动一个节点只需要把相应节点设备移到新节点即可，而不会像环型网络那样“牵其一而动全局”；

(3) 维护容易：一个节点出现故障不会影响其它节点的连接，可任意拆走故障节点；

(4) 采用广播信息传送方式：任何一个节点发送信息在整个网中的节点都可以收到，虽然这在网络方面存在一定的隐患，但由于局域网中有完整的冲突域机制，这在局域网中使用影响不大。

2.2.3 系统的硬件设备组成

本系统有采集摄像头、服务器端、无线发射设备、移动客户端组成，各组成部分功能概述如下：

摄像头内置 MPEG-4 视频采集压缩卡，每张压缩卡与多路视频输入通道及摄像头的输入端相连，对输入的模拟视频信号首先进行 A/D 转换，然后进行视频压缩，随即传输至服务器端。

视频服务器在本系统中的主要功能是对摄像头所采集来的源视频流进行再压缩或者再编码，并且按照特定的组播号和端口号对采集完成的视频流实时传输给客户端。同时，运用静态视频图像运动检测技术，对实时视频进行分析和处理。视频流数据遵守并使用 UDP 数据传输协议和 TCP 网络控制协议进行网络传输。

无线发送设备由基于 IEEE802.11 协议的无线网络的路由器组成：WLAN 与有线 AP (Access Point) 连接，AP 连接周边的无线网络终端，形成星形网络结构，使整个无线网的终端都能获得实时视频。同样遵守并使用 UDP 数据传输协议和 TCP 网络控制协议。

客户端采用基于 Windows CE 操作系统的手持移动设备。此设备大多具有无线网卡，由此接入到 WLAN 中。系统采用基于 Intel 公司微处理器的 Pocket PC，主频达到了 416Mhz 以上，具有 64M 左右的内存缓冲区，屏幕分辨率可选为 QVGA (320*240) 或 VGA (640*480)

在具体实现时，系统采用普通的数字摄像头，对监控现场的场景进行捕获，捕获到的视频流通过 USB 接口直接进入监控现场服务器端，由运动检测模块对原始视频流进行运动检测，过滤掉没有变化的视频流，将检测到有事件发生的视频图像送入视频压缩模块，经过视频压缩处理后，在本地存储后通过无线网

络以 RTP 数据包的形式广播发到远程端的监控客户端；客户端的接收模块负责接收传送来的视频流，然后交给视频解码模块，将数据解码后在移动设备的 LCD 上播放。期间服务器端与客户端的通信依靠 TCP 协议的控制流。监控系统各个模块间关系构成如图 2-2 所示。

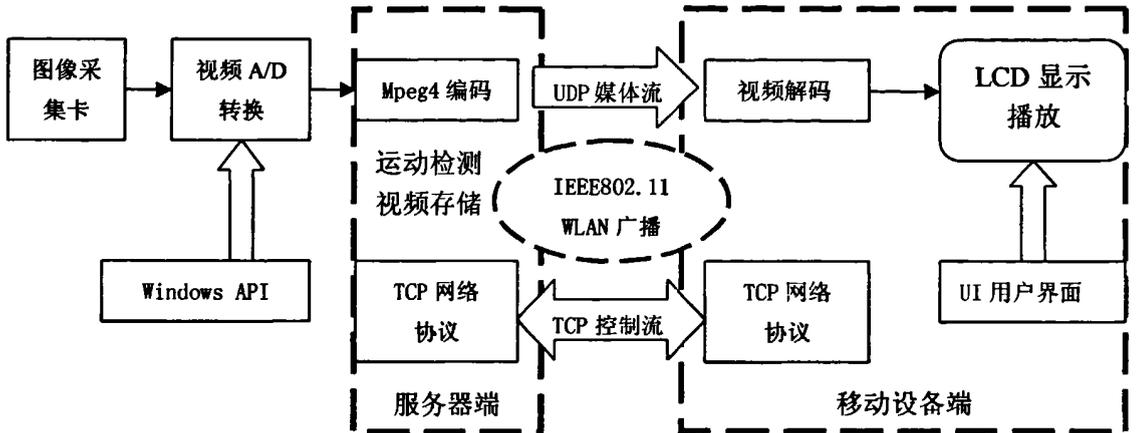


图 2-2 视频监控系统各模块间关系

2.3 系统的支撑理论与技术

2.3.1 视频图像编解码技术

通常未经处理的视频信号具有超大的数据量，不仅不容易保存，而且在信道带宽有限的无线通信中更是一大难点。不过日趋完善的各种数据压缩技术能大大压缩数据量，使视频信号在许多应用中得以可能实现。因此数据压缩在通信，存储和加密等场合成为一项不可缺少的关键技术。在信息论中，对视频信号的数据压缩是通过减少冗余信号来进行的。视频图像中含有两类冗余：统计冗余（也称为空间冗余）和心理视觉冗余。空间冗余是指图像数据中存在某种空间上的规则性。例如大面积的均匀背景就有很大的空间冗余性，因为所有这些点都可以用几种颜色来表示，而不必像原始图像中那样逐点扫描。空间冗余是由各种空间模式发生的概率不同而产生的。心理视觉冗余表示的是主观无法感受到的信息，心理视觉冗余是起源于人眼对某些空间频率感觉迟钝。例如一幅黑白图像，具有 256 级灰度，但是人的视觉通常只能分辨 64(26)级灰度，那么每个

像素就具有两个比特的冗余^[14]。

数据压缩技术主要由变换、量化和编码技术等几部分组成：

变换：把输入的图像数据加上一对一的变换，经过变换后所形成的图像数据比原始图像数据更有利于压缩。典型的变换如线性预测映射，它通过扣除像素密度的可预测部分把像素密度映射到预测误差信号上；单映射如离散余弦变换，把信号的能量压缩成若干个系数；多映射如子带分解和小波变换。

量化：生成一组有限个符号来表示压缩的图像。量化是一个到多个的映射，它是不可逆的。可以用标量量化和矢量量化实现。标量量化指的是数据逐个进行量化，而矢量量化是指一块数据一起量化。

编码：给量化器输出的每个符号指定一个码字，即二进制位流。编码器可以使用定长编码或者变长编码。变长编码(VLC)也称为熵编码，其基本原理是使得用二进制表示的符号的平均长度达到最小；把较短的码字赋给出现概率较多的符号^[15]。这是熵编码的基本原理。对活动图像的压缩，还经常采用预测技术。就是根据前一帧图像预测下一帧图像；或根据前后两帧图像预测中间一帧图像等。预测技术能够使数据得到最大的压缩比。

目前视频监控行业多采用硬件来实现视频压缩，硬件压缩的优点就是高速、高效、占用的系统资源低，故较低配置的 PC 机也可以得到高质量的视频文件，但是压缩硬件价格高，配置繁琐，压缩比不可改变等缺点制约了其在商用领域的发展。相对于硬压缩，软件压缩则需要有较高的 PC 配置，不过由于 PC 机硬件的快速发展，现在基于 Windows 的软压缩也已经很普遍。同时，国外的视频获取和应用也多数是软压缩。因此，本系统采用软件压缩而不是硬件压缩卡来实现视频图像的编解码过程。较之当今视频编解码的发展趋势，一般有两种途径：一种是自行编写编解码器，虽然可以实现很高的压缩比，但是自行开发周期长，技术复杂，开发工程量大，通用性差；另外一种是采用现有成熟的视频编解码标准。考虑到本系统的是一款通用系统，所以本文的实现方案中选择了第二种途径。

根据不同的压缩技术，产生了许多不同的视频信号压缩编码标准，本系统所使用到的压缩标准是 MPEG-4，其特点及优势是：MPEG-4 是作为一个国际化的标准来制定研究的，因此具有很好的兼容性及开放性；它在提供高压缩比的同时，对数据的损失很小，达到以最少的数据获得最佳的图像质量的目的；具有高质量的数字影像，并且允许内容创建者从 MPEG-2 质量一直到极低带宽的

Internet 流式内容全程进行品质和带宽的均衡^[16]。其原理和调用程序将在第三章详细论述。

2.3.2 无线局域网标准 IEEE802.11

无线局域网(WLAN)技术的成长始于 20 世纪 80 年代中期,它是由美国联邦通信委员会(FCC)为工业、科研和医学(ISM)频段的公共应用提供授权而产生的。这项政策使各大公司和终端用户不需要获得 FCC 许可证,就可以应用无线产品,从而促进了 WLAN 技术的发展和应用。与有线局域网通过铜线或光纤等导体传输不同的是,无线局域网使用电磁频谱来传递信息。同无线广播和电视类似,无线局域网使用频道(Airwave)发送信息。它是计算机网络与无线通信技术相结合的产物,它以无线多址信道作为传输媒介,提供传统有线局域网的功能,能够使用户真正实现随时、随地、随意的宽带网络接入。

WLAN 技术使网上的计算机具有可移动性,能快速、方便地解决有线方式不易实现的网络信道的连通问题。WLAN 利用电磁波在空气中发送和接收数据,而无需线缆介质。与有线网络相比,WLAN 具有安装便捷、覆盖范围广、经济节约、易于扩展、传输速率高等优点^[17]。而 IEEE802.11 是 IEEE 最初制定的一个无线局域网标准,主要用于解决办公室局域网和校园网中,用户与用户终端的无线接入,业务主要限于数据存取,速率最高只能达到 2Mbps。由于 802.11 在速率和传输距离上都不能满足人们的需要,因此,IEEE 小组又相继推出了 802.11b 和 802.11a 两个新标准。三者之间技术上的主要差别在于 MAC 子层和物理层^[18]。

802.11a 是 802.11 原始标准的一个修订标准,于 1999 年获得批准。802.11a 标准采用了与原始标准相同的核心协议,工作频率为 5GHz,使用 52 个正交频分多路复用副载波,最大原始数据传输率为 54Mb/s,这达到了现实网络中等吞吐量(20Mb/s)的要求。如果需要的话,数据率可降为 48, 36, 24, 18, 12, 9 或者 6Mb/s。802.11a 拥有 12 条不相互重叠的频道,8 条用于室内,4 条用于点对点传输。它不能与 802.11b 进行互操作,除非使用了对两种标准都采用的设备^[19-20]。

IEEE 802.11b 是无线局域网的一个标准。其载波的频率为 2.4GHz,传送速率为 11Mbit/s。IEEE 802.11b 是所有无线局域网标准中最著名,也是普及最广的标准^[22]。在 2.4-GHz-ISM 频段共有 14 个频宽为 22MHz 的频道可供使用。IEEE

802.11b 的后继标准是 IEEE 802.11g, 其传送速率为 54Mbit/s^[21]。

2003 年 7 月, 通过了第三种调变标准 IEEE 802.11g。其载波的频率为 2.4GHz(跟 802.11b 相同), 原始传送速率为 54Mbit/s, 净传输速率约为 24.7Mbit/s(跟 802.11a 相同)。同时 802.11g 的设备与 802.11b 兼容^[22]。

IEEE 802.11 的多种标准均支持较高数据传输速率 (1~54 Mbps), 采用微蜂窝、微微蜂窝结构, 自主管理的计算机局域网^[23]。其关键技术大致有 3 种, 直序列扩频调制技术(Direct Sequence Spread Spectrum, DSSS)及补码键控(Complementary Code Keying, CCK)技术、包二进制卷积(Packet Binary Convolutional Code, PBCC)和正交频分复用技术(Orthogonal Frequency Division Multiplexing, OFDM)^[24-25]。每种技术皆有其特点, 目前扩频调制是其主流技术, 而 OFDM 技术也由于其优越的传输性能成为 WLAN 新的发展趋势。

2.3.3 IP 实时广播技术

在通信网络中, 数据在网络的传输需要考虑到包含多个主机的共享信道网络如以太网。每个以太网帧包含源主机和目的主机的以太网地址 (48 bit)。通常每个以太网帧仅发往单个目的主机, 目的地址指明单个接收接口, 因而称为单播(unicast)。在这种方式下, 任意两个主机的通信一般不会干扰网内其他主机。然而, 有时一个主机要向网上的所有其他主机发送帧, 这就需要应用广播。广播是指在一个封闭的局域网内, 子网内部的所有用户都将收到来自广播主机封装并发出的数据包, 用户则不论喜欢与否都要无条件的被动地接收^[26]。广播能够有效的实现一对多的网络传输, 并且具有实时和有效的通讯特点。

广播数据不能越过路由器, 只在本本地子网内有效, 从而使位于不同网段的机器不能接收到广播消息, 满足了专用网络安全限制的特点; 作为一种对等通信, 它没有权限设置, 有利于网络终端的增加和扩展, 即任何一个局域网内的用户都可以实时的接收广播数据, 而发送端并不清楚有多少用户。虽然当用户过多时, 数据在整个网段上发送容易引起网络的阻塞, 但是本系统是中型规模并且处于完全封闭的网络监控系统, 最大负荷仍不足以占用超量的网络带宽。

另一方面, 广播运用的协议简单, 它仅应用于 UDP 协议, 所以它对需将报文同时传往多个接收者的应用来说十分重要。TCP 是一个面向连接的协议, 它意味着分别运行于两主机 (由 IP 地址确定) 内的两进程 (由端口号确定) 间存在一条连接。要清楚广播的传播原理和技术, 需要了解主机对由信道传送过来

帧的过滤过程。图 2-3 说明了这一过程。

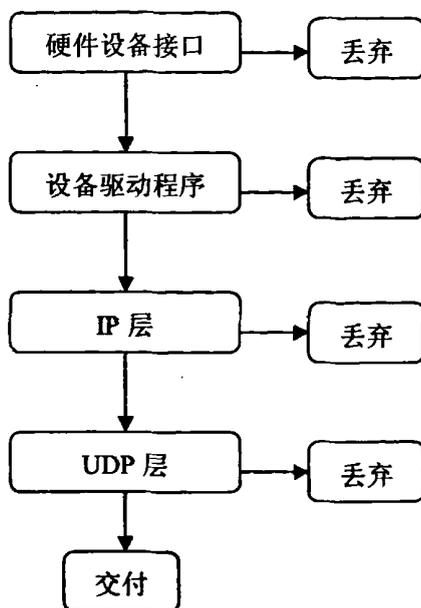


图 2-3 IP 广播帧过滤机制

首先，网卡查看由信道传送过来的帧，确定是否接收该帧，若接收后就将它传往设备驱动程序。通常网卡仅接收那些目的地址为网卡物理地址或广播地址的帧。另外，多数接口均被设置为混合模式，这种模式能接收每个帧的一个复制。如果网卡收到一个帧，这个帧将被传送给设备驱动程序（如果帧检验和错，网卡将丢弃该帧）。设备驱动程序将进行另外的帧过滤。首先，帧类型中必须指定要使用的协议（IP、ARP 等等）。其次，进行多播过滤来检测该主机是否属于多播地址说明的多播组。设备驱动程序随后将数据帧传送给下一层，比如，当帧类型指定为 IP 数据报时，就传往 IP 层。IP 根据 IP 地址中的源地址和目的地址进行更多的过滤检测。如果正常，就将数据报传送给下一层（如 TCP 或 UDP）。每次 UDP 收到由 IP 传送来的数据报，就根据目的端口号，有时还有源端口号进行数据报过滤。如果当前没有进程使用该目的端口号，就丢弃该数据报并产生一个 ICMP 不可达报文（TCP 根据它的端口号作相似的过滤）。如果 UDP 数据报存在检验和错，将被丢弃。

广播是一种应该谨慎使用的功能，不过系统处于封闭的专用网络中，广播的选择反而将有助于终端机与客户机之间更方便的联系。

2.3.4 WINDOWS MOBILE 移动设备操作系统

Windows Mobile 是 Microsoft 用于 Pocket PC 和 Smartphone 的软件平台。Windows Mobile 将熟悉的 Windows 桌面扩展到了个人设备中。使用 Windows Mobile 操作系统的设备主要有手机、PDA、随身音乐播放器等^[27]。经过了近十年的发展, Windows Mobile 也有了很不同的历史版本: 20 世纪末该操作系统称为: Windows CE 2.11, Palmsize-PC, 后开发出 Pocket PC 2002 (Windows CE 3.0), Windows Mobile 2003 (Windows CE 4.2) 和 Windows Mobile 5.0(Windows CE 5.01), SE 版支持 480x640 的 VGA 显示屏, 并可水平旋转显示画面。两种版本都加入对 WiFi 无线网络 (IEEE802.11b) 的支持。目前最新的版本是 Windows Mobile 6.1, 6.5 版本已经公布

Windows Mobile 操作系统迅速的发展得益于其显而易见的优点: 首先, 其界面类似于台式机的 Windows, 便于熟悉电脑的人操作。其次, 系统预装软件丰富, 内置 Office Word, Excel, Power Point, 可浏览甚至编辑, 内置 Internet Explorer, Media Player。第三, 移动设备与电脑同步非常便捷, 完全兼容 Outlook, Office Word, Excel 等。第四, 多媒体功能特别是对视频播放的处理十分强大, 借助第三方软件可播放几乎任何主流格式的音视频文件。最后, 系统拥有良好的设计工具和开发环境, 这使得其第三方软件数量丰富, 质量优秀^[28]。

2.4 开发环境

本系统的开发将分为服务器端开发和客户端开发两部分。二者均属于 Windows 环境下的软件开发, 其中服务器端软件将运行在 Windows XP 系统环境下, 客户端软件将运行在 Windows Mobile 5.0 系统环境下。

服务器端的开发环境使用微软公司的 Visual Studio 2005 (VS2005)。这是微软公司与 2005 年发布的一款集成开发环境, 它使得专业开发人员能够使用改进后的可视化设计工具、编程语言和代码编辑器, 享受高效率的开发环境 在统一的开发环境中, 开发并调试多层次的服务器应用程序。同时, VS 环境中安装了 DirectX9.0SDK 以及 Windows Media Encode SDK 插件, 用于视频的捕获和编码压缩。其中, DirectX SDK 是微软公司提供的一套在 Windows 操作平台上进行多媒体处理的开发包, 为媒体流的捕捉和回放提供了强有力的支持^[29]。而

Windows Media Encode 用于实现开发人员对视频的编码和发布, 使用 Windows Media Encoder SDK 开发编码应用可以将设备或者文件中的流媒体采集到并按照微软的媒体格式进行编码, 并输出, 这就是编码器的作用。Windows Media Encoder 的源可以是设备也可以是文件, 设备可以是屏幕或者摄像头或者其他视频、音频设备, 文件可以是视频文件、音频文件。Windows Media Encoder 的输出可以将内容编码到文件, 也可以对内容进行实况广播^[30]。如果要进行广播, 既可以通过推传递(在客户端没有请求数据的情况下传递)将流传输到运行 Windows Media Services 的服务器上, 也可以允许 Windows Media 服务器和播放机通过拉传递(仅在客户端发出请求时将数据传递给客户)直接从编码器接收流。

DirectX9.0SDK 和 Windows Media Encode SDK 二者同时使用可以让应用程序开发人员从复杂的数据传输、硬件差异等工作中解脱出来, 极大地降低了多媒体应用开发的复杂度, 提高了开发效率。

客户端的开发环境使用 Visual Studio 2005.net compact framework。它是完整的桌面版 .NET Framework 的一个“轻型”版本。它包括完整 .NET Framework 基类库的一个兼容子集, 并且只含有很少的、专为移动设备所设计的新类。 .NET compact Framework 也包含公共语言运行库的一个新的实现, 它是最新的, 可以有效地运行在内存容量和 CPU 能力都都有限的小型设备上。 .NET compact Framework 能为程序设计人员提供了一个完美的平台, 在其上可以方便的创建针对移动设备的新的软件产品。更完美的解决许多在实现移动解决方案时所面临的问题^[31]。

2.5 本章小结

本章根据无线监控系统的特殊性分析了系统需求, 完整给出了系统结构及实现的技术路线。确立了选用 Windows XP 作为系统开发平台、Visual Studio2005 为系统服务器端的开发工具, 并选用 C/S 系统结构模式实现; 并在该领域所用各技术的优缺点分析基础上, 确定用 .net 下的 compact framework 框架进行客户端程序设计、用 MPEG-4 进行视频图像的压缩、采用 TCP 协议传输控制指令并用 RTP/RTCP 协议对视频进行广播发送。

第3章 系统服务器端实现

3.1 功能分析和结构规划

服务器端的功能分为采集，处理，广播发布三部分。其工作原理是：当系统运行后，由视频采集模块实现对监控场景的图像采集，然后服务器对采集而来的视频流进行实时存储，同时由运动检测模块对采集到的视频图像序列进行自动判断是否有运动目标入侵，在检测到有运动事件发生后，交由视频图像压缩模块对视频数据流进行视频压缩，压缩后的视频流通过视频发送模块以广播的形式发送给客户端。如图 3-1 所示。

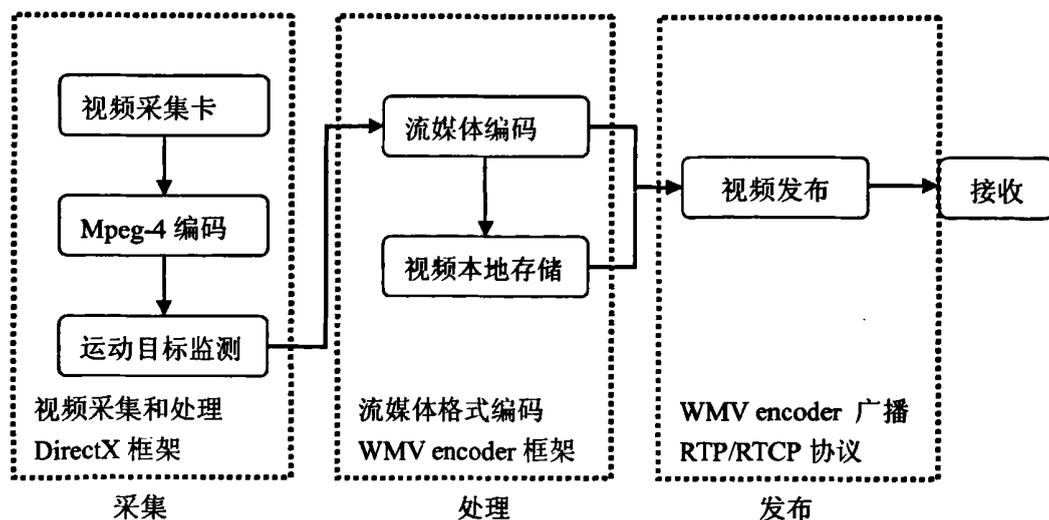


图 3-1 服务器端功能模块及流程图

3.2 编程框架和核心技术

如图 3-1 所示，各个功能模块由具体的设计框架支撑，所以在服务器设计中要首先考虑搭建程序设计的基本环境。本系统使用的设计环境是 Visual Studio 2005，其中需要用到 Direct X 9.0 SDK 以及 Windows Media Encoder 9.0 SDK。前者将用于实现底层视频采集程序的开发，后者用于实现视频流的实时编码以

及存储，并且实现对其编码后的广播发布。同时，在视频流的广播中，系统需要用到高于传输层的 RTP/RTCP 协议，将编码完成的视频流封装成 RTP 数据包。下面首先简要概述下所用的这些核心技术。

3.2.1 RTP/RTCP 协议

RTP 包括 RTP (Real-time Transport Protocol) 和 RTCP (Real-time Transport Control Protocol) 两个组成部分，是为支持实时多媒体通信而设计的传输层协议。RTP 被定义为在一对一或一对多的传输情况下工作，其目的是提供时间信息和实现流同步。提供的服务包括时间重建、丢包侦测、安全控制、内容识别等^[32]。

计算机网络的 TCP/IP 协议可以提供可靠的数据传输服务，这对非实时性数据的传输没有问题，但对实时视频等要求时延很小的其他数据时，传输质量就不能很好地保证了，这时候通常采用 RTP/RTCP 协议来传输。在多媒体数据传输中，有一个问题是数据到达时间的不可知。理想的情况下，数据按照回放所需的要求到达，但在通常的情况下往往无法做到这一点。RTP 协议是利用时间戳、系列号以及其它的结构来控制实时数据的传输质量的。在流媒体的概念中“时间戳”是最重要的信息。按照发送数据的理想状况，发送端依照即时的采样在数据包里设置了时间戳，接收端接收到数据包以后，按照时间戳的前后恢复数据。相对于实时性来说，由于监控系统对传输的图像质量要求可稍低一些，视频图像在传输的过程中有少量的丢帧现象是可以忽略的。UDP 协议因取消了重发校验等机制，故通信效率很高。因此对于实时视频监控来说，UDP 比 TCP 更适合实时视频数据的传输。然而，由于 TCP 拥有面向连接的网络安全体系，监控系统可采用如图 3-2 所示 TCP 传输控制命令 UDP 传输视频流的通信方法来实现。

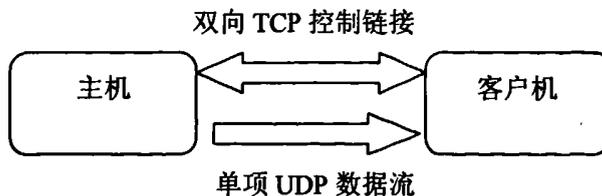


图 3-2 服务器与客户端之间的通信图

但是 RTP 协议不提供然后机制来保证实时地传输数据。也就是说 RTP 协议只管发送接收数据，而怎样提供发送接收数据的可靠性则由 RTCP 协议来完成。RTP 本身不负责同步、丢包侦测、拥塞控制，而是由上层的应用层实现；也不提供任何确保实时性传输的机制，而是由下层协议实现。同时 RTP 本身并不具有独立传输数据的能力，它通常要和 UDP 协议一起使用才能完成数据传输服务，也可以运行在 TCP 之上。RTCP 协议是一个控制协议，它是 RTP 的配套协议。主要负责监视延时和带宽以及网上传输的信息，并将其通知给发送端。即 RTP 负责数据流的传送，而 RTCP 负责传送状态的控制。RTCP 协议的功能是通过不同的 RTCP 数据报来实现的，主要包括 SR(发送端报告)，RR(接收端报告)，SDES(源描述)，BYE(通知离开)，APP(由应用程序自己定义)^[33]。由于在一个 RTP 会话过程中，发送端和接收端有间隔地发送 RTCP 包（主要是 SR 包和 RR 包），以报告传输信息和反馈传输质量。如果 RTP 可获得的带宽减少，RTCP 会立即把带宽变窄的消息通知发送端，发送端将据此信息及时调整发送速率或发送质量。RTP 和 RTCP 配合使用，它们能以有效的反馈和最小的开销使传输效率最佳化，因而特别适合实时数据的传输^[34]。

RTP 在 UDP 数据报的前端加上时间戳和序号后发出，接收端给以可控缓存，那么接收端就可根据时间戳和序号对信息和数据包进行复原，同时记录失序包并同步视频、音频和其他数据。这样一来，可以保证接收方以正确的顺序和速率播放视频流。大多数情况下，在 IP 网络中，先发送的数据包有会比后发送的包更早到达，因此接收方无法依据包到达的时间顺序来播放媒体流，接收方需要利用 RTP 的时间戳来达到这个目的。接收方将收到的 RTP 包一次放入缓冲区并依照时间戳的顺序开始播放。如果某个即将播放的媒体数据包尚未到达，接收方可以利用抽空技术填补空隙（如持续播放最后的音频包或者视频包），直到该数据包到达接收方^[35]。如果该数据包长时间未到达或者到达时发生破损和错误而不能满足播放要求的，将被丢弃。

3.2.2 视频处理的 DirectX 技术

为了对视频的特定帧进行处理（如运动目标监测，特殊区域提取），我们需要将采集来的模拟视频转换成 MPEG-4 格式的视频流放入服务器的缓存中。这就需要用到 DirectX 技术。

DirectX 软件开发包是微软公司提供的一套在 Windows 操作平台上开发高性

能图形、声音、输入、输出和网络游戏的编程接口。微软将其定义为“硬件设备无关性”，即使用 DirectX 可以用于设备无关的方法提供设备相关的高性能

其中，本系统需要用到 DirectX 9.0 家族中的一个重要成员——DirectShow，它是为在 Windows 平台上处理各种格式的媒体文件的回放、音视频采集等高性能要求的多媒体应用所提供的完整的解决方案。它可以方便快捷的让程序开发者处理各种由于不同的多媒体源和多媒体格式而产生的问题^[36]。这也是本系统所着重学要解决的。

为了高效地处理音视频数据，DirectShow 向应用提供了直接访问系统底层功能的接口，可以使应用直接控制从数据采集到数据播放的各个中间环节。DirectShow 是基于组件对象模型的，编写 DirectShow 应用程序应该掌握一定的 COM 编程知识。对大部分应用程序，开发者并不必自己写 COM 组件，DirectShow 可以提供部分所需要的组件。如果需要定制自己所需求的 DirectShow 组件，则必须将该组件以 COM 组件的形式加以实现^[37]。

在 DirectShow 框架下，使用 Visual Studio.net 环境开发视频获取程序将会使用到 DirectShow 框架特有的 source filter，它可以支持多种接口的视频采集卡，在本系统中，我们使用以 USB 接口的外挂是视频采集摄像头，其内置了硬驱动（免安装 Windows 软件驱动程序）的采集卡。

3.2.3 Windows Media 9 流媒体开发技术

由于本系统的网络通信使用的是无线网络，对带宽和编码方式有很强的要求，所以在广播发布的过程中，需要将缓存中的 MPEG-4 的视频文件转换成流媒体，这样一来，依靠流媒体文件的极强的实时性和编码效率，高效率的广播发布将可以得到满足。构建流媒体编码和广播的平台需要用到 Windows Media 9 Series SDK。它包括以下多个软件开发包：

- Windows Media Player SDK：用于媒体文件或媒体流的播放。
- Windows Media Format SDK：用于开发 Windows Media 编码格式
- Windows Media Device Manager SDK：用于开发多媒体设备的管理功能
- Windows Media Encoder SDK：用于进行音视频流的编码保存或广播，
- Windows Media Rights Manager SDK：用于有关数字版权管理方面的技术
- Windows Media 9 Series SDK 用于支撑 Windows Media 技术的应用和扩展的软件开发包，开发者可以使用这些 SDK 扩展 Windows Media 系列各部分的功能，

或者根据具体的应用要求定制不同的系统。本文的无线视频监控系统是基于 Windows Media 9 Series SDK 平台构建,使用的是其中的 WMFS(Windows Media Format SDK)和 WMES (Windows Media Encoder SDK) 来进行开发的。

WMFS 将 Windows Media Audio 和 Video 编解码器与其应用程序高度集成,开发人员不必从头开始开发编解码器、编辑和确定访问格式,实现了效率和灵活性的最大化。WMFS 是组件群中唯一与 Windows Media Audio 和 Video 编解码器联系的软件开发包,因此是本文无线视频监控系统中媒体内容编码发布能够实现重要途径。

在配置环境前,首先需要在 Visual Studio.net 中集成两个安装包,一个是: Windows Media Encoder,另一个是: Windows Media 编码器 SDK。通过 Windows Media 编码器 SDK 开发的应用程序在分发时候必须与 Windows Media 编码器一起发布。安装结束后,会在开始菜单中生成一个名为 Windows Media 的程序组,里面包含了 Windows Media Encoder 应用工具和 SDK 文档。Windows Media 编码器 SDK 所支持的开发语言有 VBScript VB , JavaScript, C, C++, C#。本系统所选择的开发语言是 C#。与 Direct X SDK 相同, Windows Media Encoder SDK 也是以 COM 的形式提供各种功能的使用。

3.3 服务器端框架具体实现

首先,摄像头将模拟信号的图像采集来,使用底层的 DirectX 技术将模拟信号转换成数字信号 (MPEG-4) 实时压入缓存。以 MPEG-4 数字格式可以方便对视频关键帧的访问和控制;然后,使用 media encoder 技术将缓存中的数字信号转换成流媒体格式 (WMV)。第二次编码压缩可以将数字视频格式转变为流媒体格式方便服务器端对视频进行广播发送,同时可以更方便的控制发送视频流的配置文件 (分辨率、缩放比);最后,服务器端主机将视频流封装在 RTP 包中,并定时发送和接受 RTCP 包进行反馈和控制。与此同时,采集后的视频流经过压缩后,压入一个已经设置的缓冲区,当缓冲区被压满时,弹出数据,封装成 RTP 包,利用 Winsock 技术以 UDP 协议的广播形式发送出去。其中包括了视频采集模块,视频编码模块,广播服务器模块三部分。

功能流程图如图 3-3:

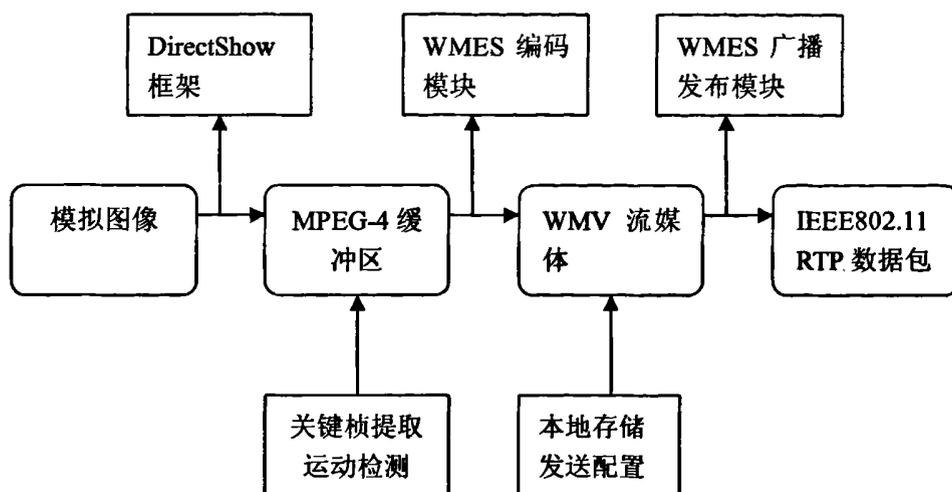


图 3-3 功能流程图

3.3.1 视频采集模块的开发

本系统的硬件采集设备选用的是 USB2.0 数字摄像头，通过它可以实现对监控场景图像的采集预览，由于 Windows 操作系统对硬件操作进行了隔离，应用程序不能直接访问硬件。所以使用 DirectShow 时，必须为这些硬件设计包装 Filter（可以认为是 DirectX 的 COM 接口）；同时这种 Filter 能够工作在用户模式下，外观、控制方法跟普通 Filter 一样，而在包装 Filter 内部完成了与硬件驱动程序的交互。本系统使用的摄像头采用 WDM（Windows Driver Model）驱动模型，这类硬件都使用 Ksproxy.ax 这个包装 Filter。该 Filter 上定义了统一的接口，而接口的实现因具体的硬件驱动程序而异，但是这个 Filter 不会预先知道要代表哪种类型的设备，所以当此 Filter 上的接口方法被应用程序或其他 Filter 调用时，它会将调用方法以及参数传递给驱动程序，由驱动程序最终完成指定功能，然后动态配置 Filter 上应该实现的接口^[38]。KsProxy.ax 能够代表各种 WDM 的音视频设备，当采集设备正确安装时，DirectShow 就能把它包装成一个 Filter，并且注册于 WDM Streaming 开头的目录下。

下面是视频图像采集过程的部分 C++代码：

- 1、首先要初始化 Filter Graph。通过以下步骤实现
- ```
// 创建 Filter Graph Manager
```

```

hr = CoCreateInstance (CLSID_FilterGraph, NULL, CLSCTX_INPROC,
 IID_IGraphBuilder, (void **) &g_pGraph);
// 创建 Capture Graph Builder
hr = CoCreateInstance (CLSID_CaptureGraphBuilder2, NULL,
CLSCTX_INPROC, IID_ICaptureGraphBuilder2, (void **) &g_pCapture);
// 得到相关接口
hr = g_pGraph->QueryInterface(IID_IMediaControl, (LPVOID *) &g_pMC);
hr = g_pGraph->QueryInterface(IID_IVideoWindow, (LPVOID *) &g_pVW);
hr = g_pGraph->QueryInterface(IID_IMediaEvent, (LPVOID *) &g_pME);
//将 Filter Graph Manager 指针设置给 Capture Graph Builder
hr = g_pCapture->SetFiltergraph(g_pGraph);

```

2、要实现视频采集，首先要枚举到当前系统存在的视频设备。首先创建一个 ICreateDevEnum 接口；再调用 ICreateDevEnum::CreateClassEnumerator()方法生成类型枚举器，返回一个 IEnumMoniker 接口；调用 IEnumMoniker::Next()方法来枚举每个视频设备。

```

//创建系统设备枚举器
hr = CoCreateInstance(CLSID_SystemDeviceEnum, NULL,
CLSCTX_INPROC_SERVER, IID_ICreateDevEnum,
(void**)&pCreateDevEnum);
//为视频捕获 Filter 目录创建一个枚举器
hr = pCreateDevEnum -> CreateClassEnumerator
(CLSID_VideoInputDeviceCategory, &pEnum, 0);
//枚举每个视频设备
while(hr = pEnum -> Next(1, &pMoniker, &cFetched), hr == S_OK)
{
Moniker -> BindToStorage(0, 0, IID_IPropertyBag, (void**) & m_pBag);
//略去得到 Friendly Name 并动态添加的过程
}

```

3、生成 Capture Filter。Capture Filter 的创建不象其他 Filter 一样使用 CoCreateInstance()就可以，它先要创建一个系统设备枚举器，从中选择代表视频捕获设备的 Capture Filter；然后与添加别的 Filter 一样调用

`IFilterGraph::AddFilter()`就可以将该 Capture Filter 加入 Filter Graph。当枚举到所选的设备后（在系统实现时，是默认选择第一个视频设备的，其枚举过程与步骤 2 相似），即可调用 `IMoniker::BindToObject()`将设备标识生成一个 `DirectShow Filter`。

```
hr = pMoniker -> BindToObject(0, 0, IID_IBaseFilter, (void**) &m_pVCap);
```

```
//如果上步成功，则将该 Filter 添加到 Filter Graph 中
```

```
hr = g_pGraph ->AddFilter(m_pVCap, L"Capture Filter");
```

4、进行视频的捕获的方法。

```
hr = g_pCapture->RenderStream (&PIN_CATEGORY_PREVIEW,
&MEDIATYPE_Video,m_pVCap, NULL,NULL);
```

如果要将所捕获的视频流存储成视频文件，可调用 `ICaptureGraphBuilder2::RenderStream()`方法，指定相应的输出文件路径等参数即可。这里我们将使用一个缓冲区然后把缓冲区指定为 Windows Media Encoder 中 `IWMEncVideoSource` 对象（将在下文介绍）的指向目标。

### 3.3.2 视频流媒体编码模块的开发

通过摄像头的视频数据是以 MPEG-2 或 MPEG-4 格式（视 `DirectShow` 捕获文件格式而定）传送到视频编码模块的。而本系统广播发送端是通过 WMV 格式对视频流进行发送，所以要针对服务器缓冲区中的视频数据进行 WMV 格式编码。

上文介绍了 Windows Media Encoder SDK，现在需要完成格式编码程序。首先，在系统工程中需要添加对“Microsoft Media Encoder”的引用，如图 3-4 所示。

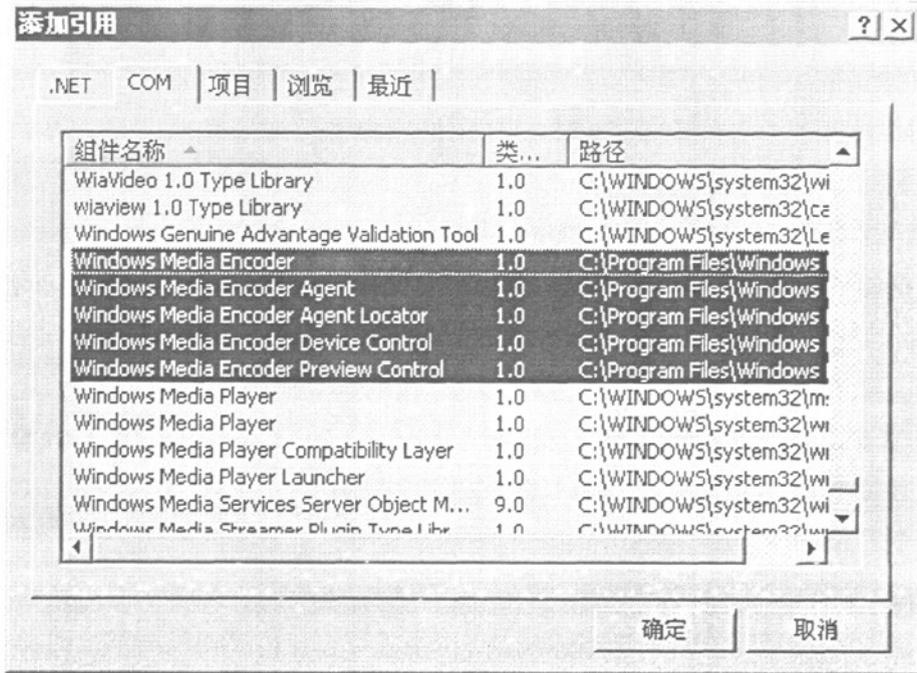


图 3-4 WME SDK 工程引用示意图

然后在代码中添加 WMEncoderLib 命名空间。SDK 中有四个对象是必须使用的，它们分别是：

- WMEncoder 对象：用于控制视频流。
- IWMEncSourceGroup 对象：用于得到来源视频媒体文件。
- IWMEncVideoSource 对象：用于得到来源音频媒体文件。
- IWMEncProfileCollection 对象：用于储存编码后的 WMV 文件。

程序中，为了实时保存 WMV 视频，需要先创建一个 WMEncoder 对象 IWMEncSourceGroup 对象，IWMEncSourceGroup 里面保存的是需要进行编码的原始视频和音频来源。通过 IWMEncSourceGroup 创建视频源对象 IWMEncVideoSource 和音频源对象 IWMEncSource。为源对象设置输入源即查找到采集端的 MPEG 缓存，在本章里 MPEG 文件位于摄像头缓冲区中。将其作为输入源后，通过 IWMEncFile 对象设定输出目的即服务器的本地储存地址，在本章内使用文件作为输出目的，除了文件以外还可以向流媒体服务器输出或者直接广播到网络上。遍历 IWMEncProfileCollection 对象选择一个配置文件，配置文件将决定输出的格式。

下面是系统转换文件格式的部分 C#代码，它将缓存区中的 MPEG 文件转换

为一个 WMV 文件。

```
using WMEncoderLib;
```

1、创建 WMEncoder 对象，创建 IWMEncSourceGroupCollection 对象，IWMEncSourceGroupCollection 用来存放 IWMEncSourceGroup 对象。

```
WMEncoder Encoder = new WMEncoder();
```

```
IWMEncSourceGroupCollection SrcGrpColl = Encoder.SourceGroupCollection;
```

2、创建 IWMEncSourceGroup 对象。

```
IWMEncSourceGroup SrcGrp = SrcGrpColl.Add("SG_1");
```

3、设定源，将视频来源和音频来源都设定为 C:\\WMSDK \\capture.mpg（视频来源和音频来源可以来自不同文件）。

```
IWMEncAudioSource SrcAud =
(IWMEncAudioSource)SrcGrp.AddSource(WMENC_SOURCE_TYPE.WMENC_AUDIO);
```

```
SrcAud.SetInput("C:\\WMSDK \\capture.mpg", "", "");
```

```
IWMEncVideoSource SrcVid =
(IWMEncVideoSource)SrcGrp.AddSource(WMENC_SOURCE_TYPE.WMENC_VIDEO);
```

```
SrcVid.SetInput("C:\\WMSDK \\capture.mpg ", "", "");
```

4、设定输出目标。

```
IWMEncFile File = Encoder.File;
```

```
File.LocalFileName = "C:\\WMSDK \\EncodeFile_dest.wmv";
```

5、下面的代码用于列举所有配置文件，并把当前编码的配置设定为适合客户端分辨率和码率的 Windows Media Video 8 for Local Area Network (384 Kbps)格式。

```
IWMEncProfileCollection ProColl = Encoder.ProfileCollection;
```

```
IWMEncProfile Pro;
```

```
for (int i = 0; i < ProColl.Count; i++)
```

```
{
```

```
Pro = ProColl.Item(i);
if (Pro.Name == "Windows Media Video 8 for Local Area Network (384 Kbps)")
{
SrcGrp.set_Profile(Pro);
break;
}
}
```

### 3.3.3 服务器端视频编码预览功能的实现

在监控系统的服务器端，通常需要一个视频预览功能，方便服务器端用户可以直观的进行监视和控制。本系统仍然使用 WME 技术实现对已经编码完成的 WMV 文件进行实时的预览。

具体功能的实现大体分为下面几个步骤：首先创建 WMEncDataView 对象。然后 IWMEncSource 对象获得 IWMEncDataViewCollection 对象。然后将 WMEncDataView 对象添加到 IWMEncDataViewCollection 集合中。开始编码。将 WMEncDataView 对象和一个窗口句柄进行绑定。最后开始预览。

实现编码预览需要使用一个数据视窗的对象，这个对象用 WMEncDataView 来实现。同时数据视窗对象需要被添加到数据视窗集合对象 IWMEncDataViewCollection 中。编码前和编码后的流数据在和视窗对象绑定之后才可以在视窗对象中显示出来。各对象详细用法如下：

IWMEncDataViewCollection 对象通过 IWMEncSource 对象中的下列属性得到预览效果：

- IWMEncDataViewCollection = IWMEncSource.PostviewCollection  
得到编码后视频实际效果。
- IWMEncDataViewCollection = IWMEncSource.PreviewCollection  
得到编码前视频预览。

IWMEncDataViewCollection 对象包含以下方法：

- int IWMEncDataViewCollection.Add( object pDataView);  
功能是增加一个 WMEncDataView 对象到集合中，返回值是表示预览流对象的 ID 值，在删除集合中对象时必须使用这个值。
- void IWMEncDataViewCollection.Remove( int lStreamCookie);

功能是通过 ID 值删除包含在集合中的指定的预览流对象。

- `void IWMEncDataViewCollection.RemoveAll();`

功能是删除集合所包含的所有预览视窗对象。

WMEncDataView 对象包含以下方法：

- `void WMEncDataView.SetViewProperties( int lStreamCookie, int hWnd);`

功能是将预览流对象 ID 值和一个窗口句柄进行绑定。这个 ID 值就是前面提到的 `int IWMEncDataViewCollection.Add( object pDataView);` 方法调用时的返回值。

- `void WMEncDataView.StartView(int lStreamCookie);`

功能是开始指定流上的预览。

- `void WMEncDataView.StopView(int lStreamCookie);`

功能是结束指定流上的预览。

下面是实现视频预览的部分 C#代码：

1、首先创建两个 WMEncDataView 对象和两个 IWMEncDataViewCollection 对象。

```
WMEncDataView Preview = new WMEncDataView();
```

```
WMEncDataView Postview = new WMEncDataView();
```

```
IWMEncDataViewCollection DVColl_preview = SrcVid.PreviewCollection;
```

```
IWMEncDataViewCollection DVColl_postview = SrcVid.PostviewCollection;
```

2、将 WMEncDataView 对象增加到 IWMEncDataViewCollection 对象中。

```
int lpreviewStream = DVColl_preview.Add(Preview);
```

```
int lpostviewStream = DVColl_postview.Add(Postview);
```

3、开始编码。

```
Encoder.PrepareToEncode(true);
```

```
Encoder.Start();
```

4、将预览和窗口对象绑定，这必须在开始编码之后进行，其中 Panel\_Preview 和 Panel\_Postview 是两个 Panel 窗口对象。

```
Preview.SetViewProperties(lpreviewStream, (int)Panel_Preview.Handle);
Preview.StartView(lpreviewStream);
Postview.SetViewProperties(lpostviewStream, (int)Panel_Postview.Handle);
Postview.StartView(lpostviewStream);
```

### 3.3.4 视频广播发送模块

流媒体编码完成后，WMV 文件除了保存在本地服务器中，还需要将其以 RTP 数据包的形式广播发送给客户端。本模块开发需要对已经编码完成的流媒体文件封装成 RTP 数据包，广播发送协议需要用到位于网络层的 IEEE802.11 标准，同时程序的实现需要结合 Windows Media Encoder SDK 广播发送技术。将在下一章：数字视频监控系统的网络通信中完成。

## 3.4 本章小结

本章首先分析了系统服务器端的功能结构，确立了视频监控服务器端的三大功能模块，同时对各个模块所需要的程序构建框架和程序设计技术做了详细的介绍和阐述。然后依次完成了对视频采集模块和视频流媒体编码模块的开发。最后简略介绍了视频广播发送模块开发的技术路线。

## 第4章 无线监控系统的网络通信

在本系统中，监控现场要向客户端，以广播方式发送视频流。网络负载使用的是工作在 2.4GHz 频段上的 WLAN，采用的技术标准是 IEEE802.11/g。在上一章的介绍中，已经完成了服务器端的视频采集和编码。下文的工作将介绍如何利用 WLAN 作为网络负载，将编码完成的 WMV 格式视频流封装为 RTP 数据包并向网内接收端进行广播。

### 4.1 IEEE802.11 无线网络的设计与规划

无线局域网有其一套特殊技术规范。系统需要设计并建设性能良好并符合 IEEE802.11/g 标准的无线网络，这样不仅可以满足目前的需要，而且日后网络还可以平滑升级，可以有效地保护投资。目前 IEEE802.11g 对大信息流量及多工作组同时工作、流量控制及更安全的保密编码、安全认证等技术问题有了完整的标准。所以，本系统在无线网络的规划和设计中需要符合其标准并遵守其规范。

#### 4.1.1 系统的网络结构

采用无线网络信号进行数据传输，通常需要设计合理高效的网络结构，在这里就需要给出一种名为基本服务集 BSS(the Basic Service Set)的网络结构概念

BSS 系统网络拓扑结构如图 4-1 所示。它是 WLAN 的基本构造模块。在该拓扑结构中，要求一个无线站点充当中心站，所有站点对网络的访问均由其控制。它至少需要包含一个 AP，AP 是 BSS 的中心控制站，网中的其他客户端站点在该中心站的控制下与其他 BSS 中的客户端站点进行通信。这样，当网络业务量增大时网络吞吐时延性能的恶化并不剧烈。由于每个站点只需在中心的覆盖范围之内就可与其它站点通信，故网络中心站点布局受环境限制也小<sup>[39]</sup>。

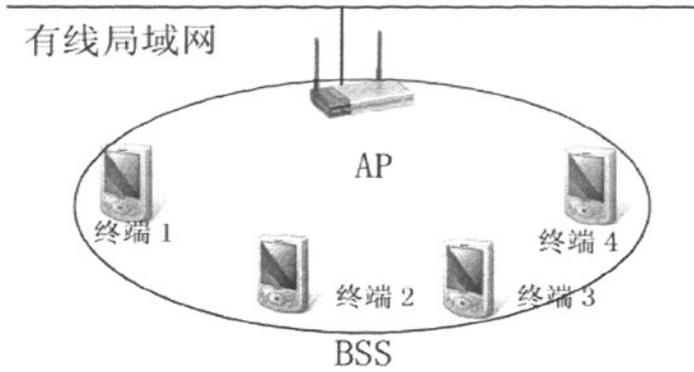


图 4-1 BSS 网络拓扑

在实际应用中无线局域网往往与有线主干网络结合起来使用。当用分布式系统将一系列重叠的 BSS 系统连接到一起,就形成了所谓的扩展服务集 ESS(the Extended Service Set)系统(见图)。ESS 实际上是一种由多个 BSS 组成的多区网,其中每个 BSS 都被分配了一个标识号 (BSSID) 用以区分不同的 BSS,以便完成 BSS 间的通信和切换。

本系统的网络结构采用单一 ESS 下多 BSS 网络结构,具体网络硬件构成如图 4-2:

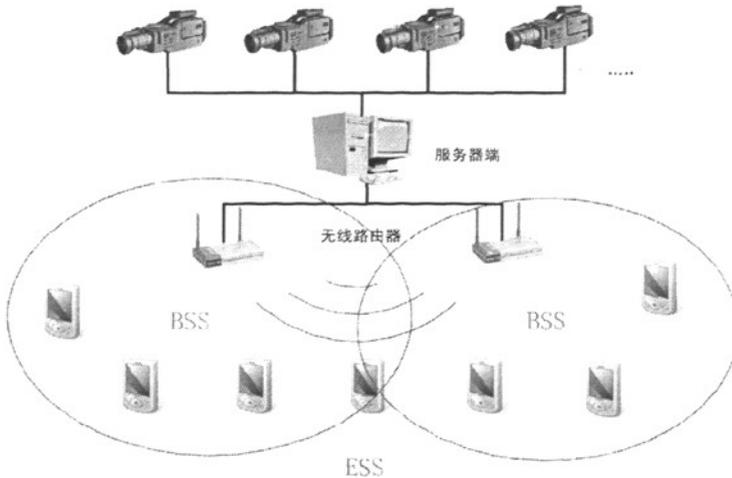


图 4-2 系统的网络拓扑结构

图 4-2 中的无线路由器充当的是 AP 的角色,摄像头与服务器以星型连接的

网络结构组成了有线局域网接入点，最后连接 AP 后构成完整的 ESS 网络拓扑。

#### 4.1.2 IEEE802.11 的接入点切换

一般来说，WLAN 采用两种网络结构，一种是基础结构模式，移动终端之间进行通信需要经过 AP 的转接；另一种是自组网结构模式，无需 AP 点，移动终端之间可以直接通信<sup>[40]</sup>。本系统使用的是第一种建网模式，这就要求在系统运行中当监控客户端来回移动时，网络试图维持连接，而当用户移出使用范围时，首先降低连接的速率。如果信号继续降低，终端设备会选择另一个信号较强的网络。这就是所谓的终端移动问题。

在 IEEE 802.11 的技术指标中，有三类终端移动：

- 非转移：终端在同一个 AP 中移动
- BSS 转移：终端在同一无线基站 BSS 多个 AP 之间移动。
- ESS 转移：终端在不同无线基站（跨 ESS）多个 AP 之间移动。

本系统需要解决第二种终端转移，即 BSS 转移。BSS 转移涉及到 WLAN 中网络切换的问题。所谓“切换”是指 STA 在移动到两个 BSS 覆盖范围的临界处时终端与新 AP 重新关联并与原 AP 断开关联的过程。IEEE802.11 无线网络标准允许无线网络用户可以在不同的 AP 范围中使用相同的信道，或在不同的信道之间互相漫游，大多数无线 AP 每隔 100ms 发射一个烽火信号，烽火信号包括同步时钟、网络传输拓扑结构图、传输速度指示及其他参数值，跨网络终端利用该 AP 烽火信号来衡量网络信道信号质量，如果质量不好，该终端会自动试图连接到其他新的网络接入点。当无线终端改变其 AP 时，无线终端用户的数据信息必须改变其物理路径。这种改变不可能瞬间完成，而且很可能触发各种错误，包括信息丢失、误插入、重复、破坏数据包的顺序以及增加等待时间。本系统网络使用 IAPP（Inter Access Point Protocol）协议进行越区切换。过程如图 4-3：

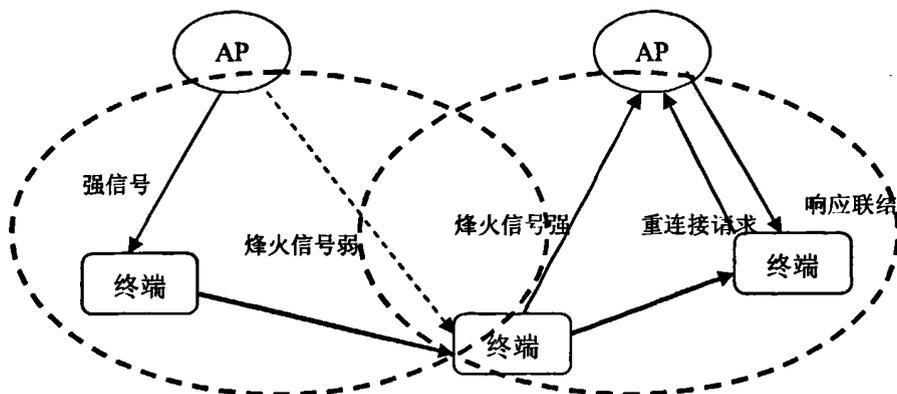


图 4-3 IEEE802.11 越区切换过程

终端首先跟踪它所联结的 AP 的烽火信号的强度，若发现强度变弱时便从相邻的 AP 开始搜索更强的烽火信号。一般来说无线网卡具有搜索信道的功能，终端将网卡搜索到的烽火信号最强的 AP 作为新的 AP 并发送关联请求帧。关联请求帧包含该终端和以前联结的 AP 的相关信息<sup>[41]</sup>。关联帧消息内容如下：

- 消息类型：Management
- 消息子类型：Association Request
- 消息内容：发起关联的终端 MAC 地址；将接收的 AP 的地址；BSSID
- 消息传输方向：从终端到 AP

然后新的 AP 会发送一个响应联结帧，响应帧消息内容如下：

- 消息类型：Management
- 消息子类型：Association Response
- 消息内容：关联请求结果识别码； BSSID
- 消息传输方向：从 AP 到终端

到这时，终端才发送取消当前关联消息帧并与旧 AP 断开，并切换到新 AP 信道上。取消关联帧信息内容如下：

- 消息类型：Management
- 消息子类型：Disassociation
- 消息内容：将取消关联的 AP 的地址； BSSID

● 消息传输方向：从终端到 AP

802.11 指标指出有可能在 ESS 转移时服务中断，而 BSS 转移时有可能不中断服务<sup>[17]</sup>。然而资料中并没有提供实现无线终端重新连接或 AP 断开功能的任何具体步骤。既然没有实现这些功能的标准方法，每个设备商可能提供其独特的测试方法。当终端正在移出一个 AP 范围时，它能够不断地在正在传送数据的低速率和未传送数据的高速率之间进行速率调整。另外，如果无线终端处于两个 AP 的范围内，它也许会不断地在两个 AP 之间切换。这种切换对性能会产生很大影响。所以需要介绍一种越区切换的算法。

越区切换是终端的移动动作，所以越区切换管理首先是终端发起的。所以终端是越区切换的主动发出者，而 AP 却是被动的，AP 只是在切换完成时对 AP 内部的动态表进行维护。而终端在某个 AP 上联结成功后，每收到一个烽火信号帧，便把当前的 AP 的信号强度储存为一个数组。然后开始做如下工作：

1. 由信号强度的数组计算处当前 AP 信号强度的平均值记为  $u$ ，方差记为  $v$ 。
2.  $P_{min}$  为终端接收要求的最低信号强度，如果  $u > P_{min} + v$ ，说明 AP 强度合适，无需切换。
3. 在 BSS 上搜索并只搜索另外两个 AP，保存在网络信息的结构表中（部分属性如表 4-4）。
4. 从网络信心的结构表中找出信号强度的最大值，记为  $AP_{max}$ 。
5. 如果  $AP_{max} > u + v$ ，就开始越区切换。

| 属性符号          | 符号说明               | 数据类型         |
|---------------|--------------------|--------------|
| BSSID         | BSS 网络区域 ID        | tBSSID       |
| ESSID         | ESS 网络区域 ID        | tESSID       |
| RSSI          | 最后一次联结成功时 AP 的信号强度 | tUInt        |
| NetAddress    | 最后一次联结成功时 AP 的物理地址 | tIEEEAddress |
| LastUpdate    | 最后一次联结成功时的时间       | tSnit        |
| TimeStamp     | 最后一次联结成功是的时间戳      | tTimeStamp   |
| BeaconPerid   | 烽火信号帧的发送频率         | tUnit        |
| Index         | 本 AP 的收发工作频率       | tUnit        |
| Capability    | AP 的保密能力和网络形式      | tUnit        |
| SupRates      | AP 支持的收发速率         | SupRates     |
| ConnectStatus | 当前的联结状态            | Char         |

表 4-4 系统中 AP 信息结构表

不过,当终端联结请求失败或者出错时,需要向原来的 AP 申请重联结。AP 对重连接请求的处理完全与连接请求的处理相同。

当然,在越区切换过程中一定会出去短暂的数据传输中止,在实时视频监控中会产生短暂延迟。不过,因为服务器是以流媒体形式对终端进行广播,终端在短暂的联结中断时会丢弃为接收到的流媒体信息。这样一样就不能妨碍到视频传输的质量。此外,IEEE 标准组织正在同 802.11 工作组一起致力于 BSS 间的无缝切换,早已开始并几近完成了 IEEE802.11r 标准<sup>[42]</sup>,它通过制定研究新的机制实现快速切换。IEEE802.11r 将可以实现 STA 与 AP 之间的快速切换,从而尽量减少切换带来的连接中断对实时业务的影响,实现宽带无线局域网对 VoIP 这类实时业务更好的支持。

## 4.2 在 WLAN 网络负载下的数据广播发送

IEEE802.11 定义了无线局域网中站点是从数据链路层接入网络。这种接口方式并不沿用有线局域网的 MAC 协议,而采用更适合无线传输环境的 MAC 协议。在实现时,MAC 层及其下层对上层是透明的,通过配置相应的驱动程序来完成与上层的接口,这样可以保证有线局域网上的操作系统或应用软件能在无线局域网上正常运行。

### 4.2.1 视频数据封装成 RTP 数据包

在服务器端向客户端进行广播发送视的过程中,服务器端将数据打包成 RTP 数据包时,要根据数据大小决定是否要先进行数据分割,发送的时候是分批以数据包的形式发送到客户端,就是说发送一个数据包需要几次 packet 发送来完成,发送成功以后再发送下一个数据包,这是因为如果数据包太大,网络传输过程中很容易造成丢包。RTP 分组长度由所经过的网络 MTU (Maximum Transmission Unit) 决定。互连网络中的各个网络可以指定不同的最大数据包大小<sup>[43]</sup>。本系统中压缩编码部分采用上文介绍的 WME 编码器,它在低码流下提供高质量的视频传输。编码完成的 WMV 格式流媒体实际上是基于 MPEG-4 的编码算法,由于在 MPEG-4 中是对每个 VOP (MPEG-4 中的压缩区域) 单独进行编解码的,因此以 VOP 为单位进行打包既提高了效率又充分利用了 MPEG-4

的编码特性。另外为了不造成 IP 碎片，包长必须不超过该网络路径的 MTU。

在 WLAN 上，通常将 RTP 分组的包长设置为 1k 字节以上。考虑到传输的高效性，取包长为当前 VOP 大小与路径 MTU 值的较小值，采用如下组包机制：若当前 VOP 能放入单个 RTP 包，就把此 VOP 单独放入一个 RTP 包中；若放不下，则把 VOP 分段，放入多个 RTP 包，此时须把 VOP 头部信息复制到每个 RTP 包，以去除包间的相关性，为减小包数，降低开销，一包中能放入多少宏块就尽可能多地放入多少宏块，但即使最后一个包中仍有剩余空间，也不能把另一 VOP 中的宏块放入此包中。

本系统中将使每段的数据不超过 1024 个字节，对于数据长度大于 1400 字节的图像帧，将其拆成不超过 1024 个字节的段，然后再分别对每段加上 RTP 协议头封装成 RTP 数据包，发送出去。在拆帧过程中，对于同一帧数据，采用相同的时间戳来标记。数据分包发送的过程如图 4-5 所示。

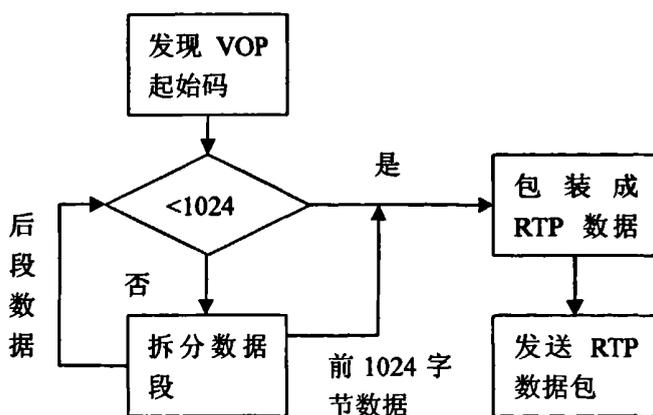


图 4-5 RTP 数据包封装发送过程

RTP 包封装 WMV 数据的算法描述如下：

```

While(发现下一个 VOP 起始码 WMV 数据流结束前) {
If(当前分段长度≤1024)
{把此段数据打入 RTP 包; }
else
{把剩余宏块打入 RTP 包}
发送 RTP 包并设置帧发送完成标记}

```

## 4.2.2 视频数据广播发送的实现

在第三章介绍了使用 Windows Media Encoder (WME) SDK 对采集来的 MPEG-4 文件进行再编码为 WMV 格式的流媒体文件。同时, Windows Media Encoder 仍然提供了 RTP 视频广播发布的 API。广播发送的 API 基于 Windows Media Format (WMF) , Format SDK 是 Encoder 的底层。广播发送的实现可以依靠 WMF 包含的接口从文件或缓冲读取媒体数据然后修改 RTP 协议依托 WLAN 网络负载进行发送;也可以使用 WME 扩展以上功能以便建立更高层次的编程解决方案。二者都将原始多媒体内容 (raw multimedia content) 转换为 windows media 文件或流,但用 Format SDK 必须在处理时每一步都管理数据流。

RTP 协议通常为一个具体的应用来提供服务,通过一个具体的应用进程实现,而不作为 OSI 体系结构中单独的一层来实现,RTP 只提供协议框架,开发者可以根据应用的具体要求对协议进行充分的扩展<sup>[44]</sup>。而流媒体传输的特点要求传输协议能够处理传输过程中的延迟和抖动现象,能够处理突发的流媒体传送,并且能够根据接收端的接收质量判断网络状况,做出相应的反馈。这就需要扩展 RTP 协议。本系统使用 WME 实现视频广播发送,并未修改和扩展 RTP 协议。

在 WME SDK 中,定义了 IWMEncBroadcast 对象作为服务器站点的接口完成流媒体的广播发送功能。当开始进行 WMV 编码时,这个接口控制并管理着局域网(WLAN 或 LAN)中终端机的广播端口号和传输协议,同时可以自动调用满足特定传输协议的媒体配置文件 (media profile)。它拥有以下两个方法:

get\_PortNumber 和 set\_PortNumber

前者用于返回广播编码内容的端口号。其中包含了储存在 WME 配置文件中的传输的协议,后者用于指定一个广播端口号。两个方法的结构和定义如下:

```
HRESULT get_PortNumber(
```

```
 WMENC_BROADCAST_PROTOCOL enumProtocol,
```

```
 short* piPort
```

```
);
```

```
HRESULT set_PortNumber(
```

```
 WMENC_BROADCAST_PROTOCOL enumProtocol,
```

```
short iPort);
```

其中 `enumProtocol` 用于列举满足广播发送时端口号的传输协议；`iPort` 是广播发送时用户指定的端口号。

下面是实现广播发送模块的部分 C# 代码：

#### 1、指定广播源

```
IWMEncSourceGroupCollection SrcGrpColl = Encoder.SourceGroupCollection;
IWMEncSourceGroup SrcGrp = SrcGrpColl.Add("SG_1");
IWMEncSource SrcAud = SrcGrp.AddSource(WMENC_SOURCE_TYPE.WMENC_AUDIO);
IWMEncVideoSource SrcVid = SrcGrp.AddSource(WMENC_SOURCE_TYPE.WMENC_VIDEO);
SrcAud.SetInput("E:\\demo.wmv", "", "");
SrcVid.SetInput("E:\\demo.wmv", "", "");
```

#### 2、指定 RTP 配置文件

```
IWMEncProfile Pro;
IWMEncProfileCollection ProColl = Encoder.ProfileCollection;
ProColl.ProfileDirectory = "C:\\WMSDK\\DEMO\\";
```

#### 3、查找并在广播中更新配置

```
ProColl.Refresh();
for (int i = 0; i < ProColl.Count; i++)
{
 Pro = ProColl.Item(i);
 if (Pro.Name == "ProfileSample")
 {
 SrcGrp.set_Profile(Pro);
 break;
 }
}
```

#### 4、建立广播对象并设置广播端口

```
IWMEncBroadcast BrdCst;
```

BrdCst= Encoder.Broadcast;

BrdCst.set\_PortNumber

(WMENC\_BROADCAST\_PROTOCOL.WMENC\_PROTOCOL\_HTTP,4367);

当然，在广播中，我们可以使用 EnumPorts 函数查找无线网络终端可用的数据交换端口，这里便不在陈述。演示系统中指定了 4367 作为通信端口。

值得注意的是，本系统的广播模块并不是将缓冲区中的 WMV 文件进行 RTP 封装，而是在 encoder 对象触发了 start 函数(在服务器端编码模块中触发)时便将编码完成的 WMV 文件实时封装为 RTP 数据包进行广播。RTP 数据包发送出去以后才将 WMV 文件写入缓冲区。这样减少了延迟并保持了视频监控的实时性。

### 4.3 系统服务器端运行结果分析

演示系统有两个监控摄像头，预览窗口分辨率为 QVGA (320\*240 并依照 WMV 配置文件自适应)。服务器端预览画面清晰无抖动，并开始在内网发送封装完毕的 RTP 数据包。当监控摄像头增多时，可以进行选择性监控以满足系统的扩展的需要。由于是对采集而来的 MPEG-4 文件再编码，所以预览画面与实时环境有近 2 秒的延迟。服务器端程序运行状态如图 4-6:

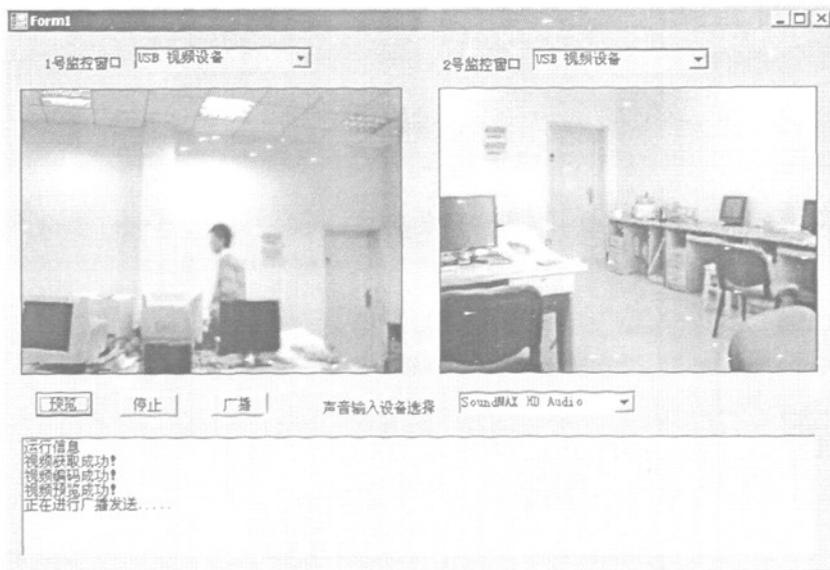


图 4-6 服务器端程序运行结果

## 4.4 实时视频QoS的控制

实时视频 QoS 服务质量是根据网络状况自适应地控制视频质量，主要目的是保证在视频传输过程中改善视频质量。由于无线局域网的特殊性，虽然无线网络发展迅猛，但目前的无线网络还处于过渡阶段。迄今为止，还没有任何 WLAN 可以保证传输语音和视频流的带宽和服务质量。WLAN 只能提供“尽力而为”的服务，在负载较重特别是有其他类别信息同时传输的情况下，系统的实时性必将受到一些影响。为此 IEEE 正在制订 802.11e 标准，以解决这一问题。

802.11e 的 QoS 标准目前仍在制订过程中，它为网络站点创建分配带宽的机制，并为物理介质中可能发生的意外情况提供时间保证。标准提供了两种机制，既可单独使用，也可结合使用。这种功能称为无线多媒体扩展 (Wireless Media Extensive)，它可实现不同数据流的优先权分配。为此，每个服务器和 AP 可以实现最多 4 种类别 (背景、尽力而为、视频、音频)，根据 802.11e 区分优先级的数据包将被映射。为此每个服务器应该具有相应的 4 个等候队列<sup>[41-45]</sup>，并对数据链路层的以太网帧头加入优先级字段，同时对发送到终端终端的数据包进行分类。本系统中视频的优先级应被定为最高级，当无线网络的传输负荷受到影响时，首先应该满足视频的实时性。

然而，WLAN 只能简单的区分业务的优先级，并不能精确定义详细的参数指标。所以需要主动对 WLAN 的 QoS 进行测量，即用测量的手段获取网络的性能和 QoS 的参数。当前基于 IEEE802.11 的 WLAN 通常使用主动测量的手段，通过向网络中发送实际的业务量进行测量，利用对这些业务量的测量来反映网络用户服务的参数，从而进一步获得吞吐量、丢包率和延时抖动等指标。

## 4.5 本章小结

本章介绍了系统中视频广播的实现。深入分析了基于 IEEE802.11 的 WLAN。研究了网络规划和设计要求，系统使用跨 BSS 转移模式的模式来进行无线网络的 AP 切换。继续利用 WME 技术实现了 WMV 流媒体文件的广播发送，并依据 IEEE802.11e 这一新标准对无线网络的 QoS 进行了研究和控制。

## 第5章 移动监控客户端的设计与实现

本系统使用的监控客户端是内置了基于 Windows CE 嵌入式操作系统的 Pocket PC（简称 PPC），其硬件性能属性如表 5-1：

|        |                                              |
|--------|----------------------------------------------|
| CPU    | Intel XScale PXA263 芯片，频率 400MHz             |
| 内存     | 128M（可扩展 SD/MMC 卡槽）                          |
| 闪存 RAM | 32M                                          |
| 显示屏幕   | 240×320 像素 TFT 屏，颜色 65536 色                  |
| 无线网卡   | Intel 集成 wireless netcard 支持 IEEE802.11a/b/g |
| 操作系统   | Windows Mobile 5.0（内置 Media Player 9）        |
| 电池     | 3600mAh 锂电池                                  |

表 5-1 PPC 硬件性能属性

需要介绍的是 Intel 的 XScale 处理器，它主要用于掌上电脑等便携设备，它是 Intel 公司基于 ARM 处理器发展的产品，在架构扩展的基础上同时也保留了对于以往产品的向下兼容，因此获得了广泛的应用。相比于 ARM 处理器，XScale 功耗更低，系统伸缩性更好，同时核心频率也得到提高，达到了 400Mhz 甚至更高。这种处理器还支持高效通讯指令，可以和同样架构处理器之间达到高速传输。其中一个主要的扩展就是无线 MMX，这是一种 64 位的 SIMD 指令集，在新款的 Xscale 处理器中集成有 SIMD 协处理器。这些指令集可以有效的加快视频、3D 图像、音频以及其他 SIMD 传统元素处理。

### 5.1 客户端功能分析

客户端的功能是：系统启动后，由视频接收模块通过移动设备的无线网卡接收从服务器端广播发送过来的 WMV 视频数据，直接进行实时解码播放，由于 PPC 特定的硬件属性，不适合在其内存区中储存接收到的视频文件。故其不同于以 PC 为载体的监控客户端，没有视频存储，视频回放功能。其主要工作流程如图 5-2 所示。

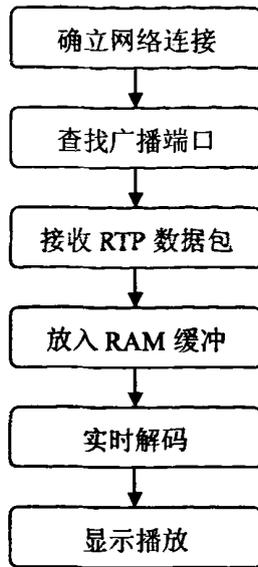


图 5-2 客户端工作流程

监控客户端首先确认网络连接是否成功，如果成功就扫描广播端口，随后找到广播 IP；然后开始接收无线路由器发送到的 RTP 数据包，去掉报头和协议字段后放入固定大小的 RAM 中；与此同时，客户端的解码器对 RAM 中的 WMA 视频流进行解码；最后，播放器对解码后的视频在 Windows Mobile 5.0 操作系统层上进行播放并在 LCD 屏幕上显示出来。

在第四章，系统完成了对 RTP 数据包的广播发送，PPC 客户端的与无线发射端处于同一个广播域中，其上的无线网卡将以上面介绍的方式接收到数据包。然后本章中需要完成的客户端功能重点在于如何读取接收到的数据包并将其解码显示出来。客户端功能架构模块如图 5-3 所示。

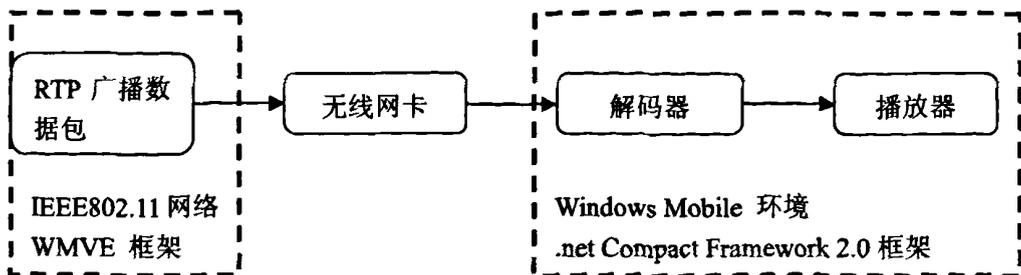


图 5-3 客户端各功能模块

基于 Windows CE 的 Windows Mobile 5.0 嵌入式操作系统在第二章曾经详细介绍过，其中随系统安装了 Windows Media Player 9（捆绑了 WMV 的编码器）。从理论上来说，Windows Media Player 可以很好的播放广域网上的单播或者广播视频流，但是由于内部拥有过多的协议匹配，从而无法直接将 RTP 数据包进行解码，所以导致其不能方便有效地播放无安全检测的特定广播地址的视频流；并且 Windows Media Player 作为一种可以对多种媒体格式进行解码的播放器，它的实时广播延迟和电力续航能力必然会影响。从本系统的必须进行延迟很小的实时监控这一要求出发，客户端的必须设计一种简单有效的自定义的解码播放器。这就涉及到 Windows CE 系统下的嵌入式软件开发问题。

## 5.2 客户端开发环境分析和架构

在第二章选定的技术路线中，本系统客户端视频播放软件使用 Microsoft .NET Compact Framework 2.0 在 Visual Studio 2005 托管环境下进行开发。为了方便方便调试，开发工具在 PC 环境上安装了 Windows Mobile 5.0 for Pocket PC 模拟器（使用 Microsoft WM5 Emulator SDK）。架构完成的虚拟机环境如图 5-4：

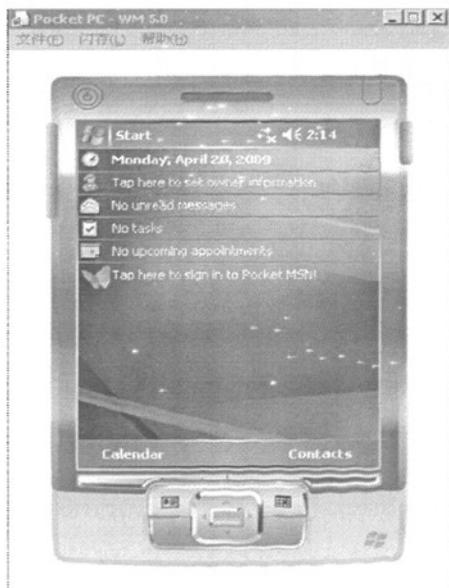


图 5-4 客户端虚拟机环境

在许多情况下，将标准媒体播放器作为单独的进程启动（如前面服务器端的开发所示）可能会满足业务需求，但如果需要更多地控制视频播放，则可以使用媒体播放器控件。媒体播放器控件包括在 Windows Mobile 5.0 软件中，它使开发人员能够将媒体播放器用作自己应用程序的一个自定义控件，但是由于在 Visual Studio 2005 环境下 .NET Compact Framework 2.0 没有对 ActiveX 控件的内置支持，使用托管代码（.net 语言）来使用 ActiveX 控件将要使用 .NET Compact Framework 2.0 中包括的 COM 互操作性来包装 ActiveX 控件<sup>[46]</sup>。

### 5.3 客户端各功能模块的实现

存储文件只需将数据从接收端缓冲区中拷贝到本地文件，需要时进行解码和播放，本章重点介绍在 Windows Mobile 模拟器下使用 DirectShow 技术和 Windows Media Encoder（WME）技术设计并完成视频解码和播放两大模块的开发。

#### 5.3.1 接收与解码模块的实现

客户端对视频的解码流程是：首先通过无线网卡得到数据信号输入，得到的多媒体数据已经去掉了报头和 RTP 协议字段；然后将多媒体数据进行转换，转换过程使用视频解码芯片完成；最后交给上层应用程序处理，处理后播出图像和声音。

对于不同架构的移动设备来说，芯片的架构有所不同，本系统客户端使用 Intel XScale PXA263 芯片组，这款芯片组包括了独立的视频、音频和输入输出控制芯片。同时，它依靠自身的 16 位 SIMD 指令来对视频、音频进行处理，支持 MPG、MPEG、WMV、AVI、DAT、VOB 等文件的解码<sup>[47]</sup>。

客户端系统需要使用高级语言，获取和控制由 PXA263 芯片组解码完成的 WMV 视频数据，并交由播放模块进行播放。在 Windows CE 3.0 以上版本的操作系统中，解码过程由底层的机器指令完成，可以无需程序设计者处理。如果希望在高级语言中调用解码器，则需要调用第三章提到 Microsoft Windows Media 9 Series 提供 WME SDK，并使用 DirectShow 来编写一个 Source filter。实际上 WME 是一组底层 API，可以将它封装成一个 DirectShow 下的 Receive filter。本

系统客户端的实现就采用了 WME 封装。与此同时，在本系统的服务器端中，使用了“推”模式，所以客户端实际上是被动的接收到视频文件，所以客户端是无法控制视频流的播放。下面介绍本系统使用 DirectShow 技术，通过 C++ 语言在 Visual Studio 2005 Windows Mobile Emulator 环境下将 WME 封装成 filter 的实现过程。

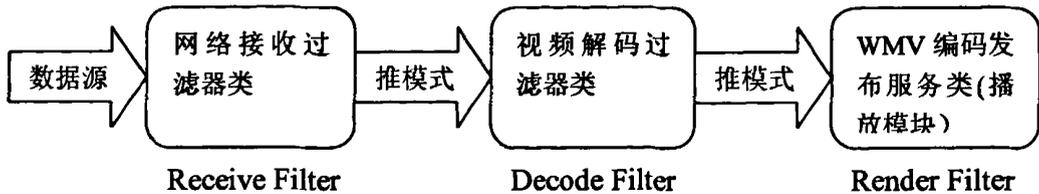


图 5-5 客户端 WME 封装 filter

采用推模式实现视频流的网络接收解码播放的过滤器类的连接如图 5-5 所示。其中网络接收和视频解码这两个过滤器类是需要重写基类设计实现的，Render Filter 则可以通过智能连接来完成。在推模式下，必须在数据源过滤器类的下游 Receive Filter 类的输入 Pin 上实现一个网络接收过滤器类的输入 Pin 类接口，数据通过数据源过滤器类调用这个接口的 Receive 方法进行传输；然后将接收到的数据“推”给视频解码过滤器类；经过解码，最后将数据“推”给 WMV 编码发布服务类。

下面使用 DirectShow 封装成 filter 模块：

网络接收过滤器输入 Pin 类 CStreamingInputPin，继承自 CRenderedInputPin 类，用于实现 CFilterWMVStreaming 上的输入的 pin，Filter 通过该输入 pin 接收网络接收端，也就是无线网卡中得到的视频数据源。这个类实现以下功能：

- 重写方法 EndOfStream。
- 实现 IPin::BeginFlush 和 IPin::EndFlush 两个函数。
- 重写方法 CBasePin::Receive()，接收视频数据源并发送。

类结构如下：

```

Class CStreamingInputPin : public CRenderedInputPin
{
 virtual HRESULT Receive(IMediaSample * pSample);
}

```

```
HRESULT CompleteConnect(IPin *pReceivePin);
HRESULT CheckMediaType(const CMediaType*);
}
```

Receive 方法接收到流中的媒体文件，定义如下：

```
HRESULT Receive(IMediaSample *pSample);
```

其中 pSample 为指向文件源的指针。本函数有多个 HRESULT 结果的返回值如下表：

| Value                  | Description |
|------------------------|-------------|
| S_OK                   | 文件源接收成功     |
| S_FALSE                | 文件源被拒绝      |
| E_POINTER              | 空指针错误（无文件源） |
| VFW_E_INVALIDMEDIATYPE | 可用的文件类型     |
| VFW_E_RUNTIME_ERROR    | 超时错误        |
| VFW_E_WRONG_STATE      | 上游 PIN 被终止  |

视频编码过滤器类 CFilterWMVStreaming 类继承自 DirectShow 的 CBaseFilter 类和 WME 的，这个类实现以下功能：

- 将类中定义 CFilterWMVStreaming 上的 Pin 的实例 m\_videopin。
- 实现继承自 CBaseFilter::GetPin 用于返回 Filter 上各个 Pin 的对象指针。
- 实现继承自 CBaseFilter::GetPin 用于返回 Filter 上各个 Pin 的数量。
- 将 CStreamingInputPin “推”来的数据委托给类 IWMEncInputCollection 的解码器对象进行处理。

WMV 编码发布服务类：CWMVStreamer，继承于 WMVE 的 IWMStatusCallback 类，这个类主要实现以下两大功能：

- 完成发布服务的初始化工作
- 接收到数据进行编码发布服务

类结构如下：

```
Class CWMVStreamer : Public IWMStatusCallback
{
CFilterWMVStreaming *m_pFilter;
HRESULT ConfigVideoInput(GUID guidSubType , VIDEOINFOHEADER *
```

```

inVideo);
//开始发布服务
HRESULT StartStreaming(void);
//停止发布服务
HRESULT StopStreaming(void);
//接收到数据进行编码发布服务
HRESULT ReceiveVideo(IMediaSample * pSample);
HRESULT ConnectionStreamType (WCHAR * inConnectionName, GUID *
outType);
}

```

客户端与服务端之间的交互端口包括命令端口和视频端口，当要求读取数据时，数据的读取由 `CWMVStreamer::StartStreaming()` 完成。

### 5.3.2 播放模块的实现

通过上文介绍的 .NET Compact Framework 2.0 中包括的 COM 互操作性对 ActiveX 组件的包装，这样就可以方便的使用 Window Media Player 组件。它不同于 PPC 操作系统中自带的 Window Media Player 程序，而是通过自定义重新设计的简单高效的播放组件。它省去了多项格式勘测和播放控制的功能，仅保留了对 WMV 文件的播放能力。其专用性和高效性完全满足视频监控系统各项要求。

下面是监控端播放器的部分 C++ 代码，它将缓存区中的 WMV 文件播放出来：

```

CAxWindow m_wndView;
CComPtr<IWMPPayer> m_spWMPPayer;
CComPtr<IConnectionPoint> m_spConnectionPoint;
DWORD m_dwAdviseCookie;
1、创建主应用程序窗口
RECT rcClient;
GetClientRect(&rcClient);
m_wndView.Create(m_hWnd, rcClient, NULL,
 WS_CHILD | WS_VISIBLE | WS_CLIPCHILDREN | WS_CLIPSIBLINGS);

```

```
if(NULL == m_wndView.m_hWnd)
 goto FAILURE;

CComPtr<IAxWinHostWindow> spHost;
HRESULT hr = m_wndView.QueryHost(&spHost);
if(FAILMSG(hr))
 goto FAILURE;

hr = spHost->CreateControl(CComBSTR(
 _T("{6BF52A52-394A-11d3-B153-00C04F79FAA6}")), m_wndView, 0);
if(FAILMSG(hr))
 goto FAILURE;

2、得到 GUID，创建封装完成的 Media Player 控件(m_spWMPPlayer)
hr = m_wndView.QueryControl(&m_spWMPPlayer);
CComWMPEventDispatch *pEventListener = NULL;
hr = CComWMPEventDispatch::CreateInstance(&pEventListener);
CComPtr<IWMPEvents> spEventListener = pEventListener;
CComPtr<IConnectionPointContainer> spConnectionContainer;
hr = m_spWMPPlayer->QueryInterface(__uuidof(IConnectionPointContainer),
 (void**)&spConnectionContainer);
hr = spConnectionContainer->FindConnectionPoint(__uuidof(IWMPEvents),
 &m_spConnectionPoint);
hr = m_spConnectionPoint->Advise(spEventListener, &m_dwAdviseCookie);
```

全局声明（前四行）之后，控件主机窗口（m\_wndView）作为主应用程序窗口（m\_hWnd）的子窗口（并且具有和客户端区域相同的大小）进行创建。然后，检索主机窗口的接口引用（spHost）并用它来创建媒体播放器控件。GUID（对应于与版本无关的 ProgID WMPPlayer.OCX）用于创建该控件，从该主机窗口检索该控件的实际接口引用（m\_spWMPPlayer）。最后，使用侦听所有类型事件的事件调度程序（pEventHandler）建立事件处理。当 ActiveX 控件的包装准备就绪时，可以像处理任何托管控件那样处理该控件。该媒体播放器控件可以按以下

方式使用展示视频加载:

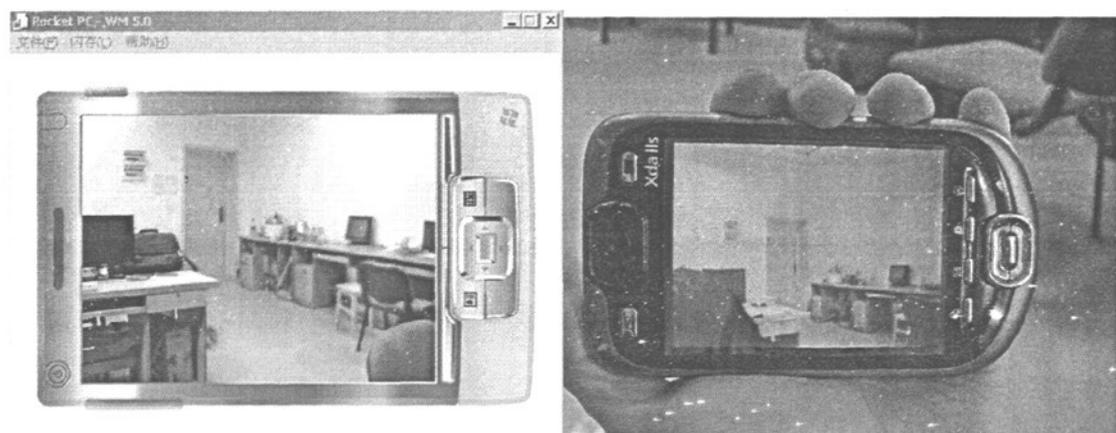
`windowsMediaPlayer.URL = fileName();`

URL 属性可以是指向流式源或远程文件的 URL, 本系统中, 它指向视频数据的广播地址然后调用解码模块对数据进行解码后播放。本系统中, 使用 WLAN 客户机可用端口号进行访问。

正如在关于从本机代码使用多媒体播放器控件的讨论中提到的一样, 该控件有许多功能还没有挖掘。在独立媒体播放器中可以进行的任何操作在该控件中几乎都可以实现。示例包括使用播放列表、连接、远程媒体、播放器状态以及很多事件 (如播放和结束流)。这可以满足客户端软件的以后的升级需求。

客户端运行结果分析

服务器启动后, 在客户端 URL 输入广播地址: `http://192.168.2.112:4367` 后客户端接收到的视频流清晰流畅。结果如图 5-6。



(1) 虚拟机环境下演示图

(2) 真机环境下演示图

图 5-6 客户端演示结果图

## 5.4 系统客户端结果分析

由于视频图像由采集到发送经历了一系列的数据处理, 因此接收端在接收到视频数据时要比发送端晚, 就是延时问题。不仅仅如此, 另外还由于系统在进行网络传输时, 对于接收到的数据要再次填充到 Pocket PC 里面的缓冲区, 即进行一次数据的拷贝, 这些都会造成实时播放的时延, 系统调试过程中, 从接

收端开始接收到视频图像出现的延时大约在 5 秒左右。接收端的视频播放在此延时过后可以流畅的进行。使用真机测试，Pocket PC 可接收的到 AP 无线数据的有效半径为 150 米（有物体阻隔）和 200 米（空旷地带）。综上所述，下一步可以自己实现 RTP/RTCP 协议下的网络传输部分，使得接收到的数据直接拷贝到应用层，而不用先拷贝到 Socket 的缓存里，另外还可以研究采用更高效的接收端缓存技术及解码技术来减小延时。

## 5.5 本章小结

本章详细介绍了无线视频监控系统客户端的设计和编码。服务端的实现主要是完成了在 PPC 模拟器环境上对接收解码和实时播放两大功能模块的设计和编码。其中，接收和解码使用 DirectShow 技术将 windows Media Encoder 封装成 RTP 数据接收 Filter 和 Windows Media 解码流媒体服务 Filter。播放模块在模拟器环境下使用 Windows Media Mobile 组件，实现了视频实时播放功能。

## 第6章 视频监控的智能化技术

在一般情况下，视频监控人员并非是一个可以完全信赖的观察者，他们在观察实时的视频流或观察录像回放的时候，由于监控人员个体条件的不同以及自身生理上的弱点，经常无法察觉到安全威胁，从而可能导致漏报现象的发生。随着视频分析技术、多媒体数据库、人工智能技术的发展，智能化视频监控逐步走进了安防应用市场，智能化技术能够及时、自动地从原始视频信息中提取大量有用信息，用来完成视频的传输保存和检索，也可以驱动其他数据、触发其他行为，轻而易举地完成人力很难完成的任务。这样可以实现监控系统智能化，弥补传统监控系统固有的不足。

### 6.1 监控系统的运动检测技术

本文提出的“运动检测”功能需要在系统中实现一个运动检测算法来判断监控图像的数据信息中是否包括运动目标，并根据判断来控制存储、发送模块的动作，从而有效地节省存储数据的空间，减少不必要的数据传输。另外还可以在对图像序列自动进行运动检测的基础上，增加报警接口，减轻监视人员的视觉负担。

#### 6.1.1 运动检测的理论

在当前的运动检测技术发展过程中，比较成熟的实现手段是采用“相邻帧差法”<sup>[48-50]</sup>来提取运动目标。所谓相邻帧差法是在连续的图像序列中两个或多个相邻帧间采用基于像素的时间差分并且阈值化来提取出图像中的运动区域。其方法简单、运算量小，易于实现，对光线等场景变化不太敏感，对于动态环境具有较强的自适应性，鲁棒性较好。然后，由于本系统应用与静止背景下的监控环境，监控视频或多或少会受到光照和噪点的影响，故研究界提出了一种累计图像差分法和对称图像差分法，可以在提高检测概率的同时降低虚报概率<sup>[51]</sup>，故本文采用相邻三帧差分的方法来提取运动目标，它是累计图像差分法的一种简单形式，可以有效地去除摄像头镜头本身的颗粒噪声和光照的影响，并

且能够快速有效地从背景中检测处静止背景下的运动目标。

帧差法数学表达式公式 6-1 所示。

$$Max|f_1(x,y)-f_2(x,y)| \geq T + \lambda \frac{1}{N_A} \sum_{(x,y) \in A} |f_1(x,y)-f_2(x,y)|$$

其中  $f_i(x,y)$  为灰度图像序列在  $t$  时刻点  $(x,y)$  处的分别为第  $i$  帧和  $i-1$  帧的平均灰度值。上式也为识别运动目标报警的判决条件， $T$  为某个灰度阈值。当目标区域出现移目标时，通过帧差上式判决条件成立，向用户报警端发送信息。该式  $T$  后的一项为添加项，目标区域光照发生明显变化时，添加项会大于不等号左边，导致判决条件不成立。这样可以有效的抑制了光照带来的影响<sup>[52]</sup>。判定过程如图 6-1 所示。

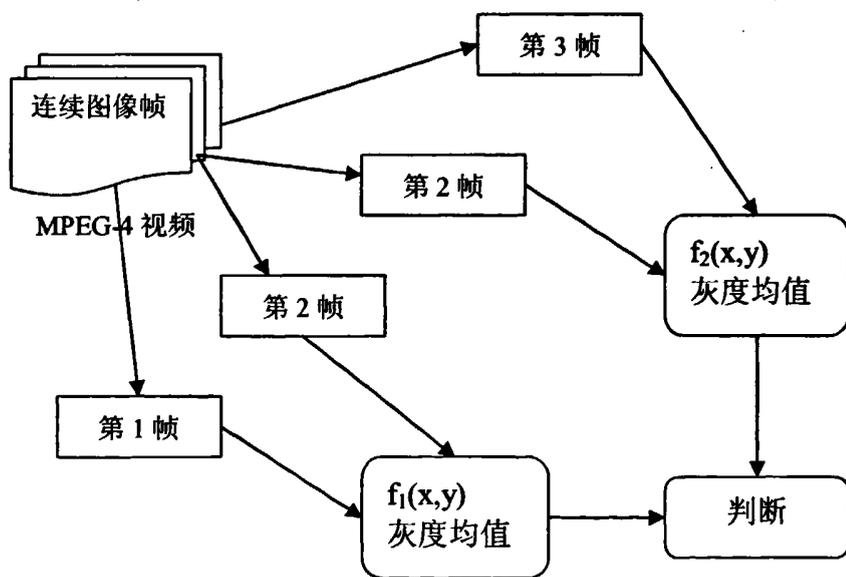


图 6-1 某时刻运动检测算法过程图

采样相邻三帧判决并不影响获取 BMP 图片的时间间隔，这是因为在判决时总是对当前一帧进行判决，如果目标区域出现移动物体，则符合判决条件。

在此算法中，阈值  $T$  的选择非常重要。若  $T$  过大，则会造成漏判；过小则会造成误判。当前业界通常采用自适应的阈值作为判定标准，所以必须建立长期的阈值样本库。而本系统由于缺少相应的阈值样本，故选择固定可调的阈值  $T$ 。

由于运动目标的出现占有一定的时间，因此不必连续检测图像序列中的每个相邻帧，当没有检测到运动物体时，系统选取时间间隔为 2s 从视频图像序列中提取相邻两帧图像进行检测，当检测到有运动物体时，则连续 10s 不进行检测而是直接将此时间段内的数据压缩后进行存储或发送。

### 6.1.2 静止背景下的运动检测模块

在第三章的介绍中，系统使用了 DirectShow 技术对视频进行采集并以 MPEG-4 格式编码，然后再通过 WME 技术对 MPEG-4 再编码为 WMV 流媒体格式随后进行广播发送。进行两次视频压缩编码的原因主要在于 WMV 流媒体文件只需占用很少的无线网络带宽，方便广播传输。但是由于流媒体的特殊性，WMV 文件无法准确的对视频中特殊帧进行处理，这样就妨碍了对实时视频进行运动检测。所以，在这里需要继续使用 DirectShow 技术对采集来的 MPEG-4 视频进行特定帧的处理。

模块实现的思路如下：首先写一个 DirectShow 的 Videoprint Filter 用于以 2 秒的间隔实时获取图像；然后后在用 DirectShow 连接在摄像头的 capture output pin 上面摄像头采集的每一帧图像都会经过你的 filter。最后储存视频时用上该 Filter，这样就可以实时处理视频的图片。

Videoprint Filter 是从 CTransInPlaceFilter 类派生而来，并实现一个自定义接口，为了便于运算中对数据的缓存和处理，在该派生类中开辟了三个帧存分别用于放置所提取的相邻三帧的图像数据。具体实现过程使用 DirectShow 技术并以 Sample Grabber Filter ( Sample Grabber Filter 在 SDK 目录中 Samples\C++\DirectShow\Filters\Grabber 提供了源代码)为基类，将其改写成为一个 Videoprint Filter。Sample Grabber 可以抓取任何类型的数据。编程过程如下：首先要创建 Sample Grabber，并将之加入到 Filter Graph 中。然后给 SampleGrabber 设置 Pin 上连接用的媒体类型。最后构建并运行 FilterGraph 并选择其特有的缓冲工作模式，即将输入的 Sample 进行缓存后，再往下传送。在缓冲模式下，我们可以调用 IsampleGrabber: GetCurrentBuffer 来获取 Sample 数据。

提取出来的图像将进入运动检测 filter。由前面对运动检测算法的分析，在实现运动检测的过程中，首先要从 Sample 中提取相邻的三帧图像进行预处理使其转化为灰度图，程序中用三个  $320 \times 240$  的数组 A、B、C 来存放灰度值信息；然后将第一二帧、二三帧灰度数组分别求矩阵均值后放入 D、E 两个数组；最后

将 D、E 矩阵相减并取绝对值来判断是否存在运动。判断的依据是将取绝对值后的数组 A 中的各项与给定阈值 T 比较，并统计所有大于阈值 T 的个数；根据统计个数是否大于给定阈值 N 来最终判定是否有运动物体入侵。如果判断有运动物体出现则报警；反之，则删去缓存中三帧图像，对下一个时间段的图像进行提取分析。其具体 Filter 结构如图 6-2 所示。

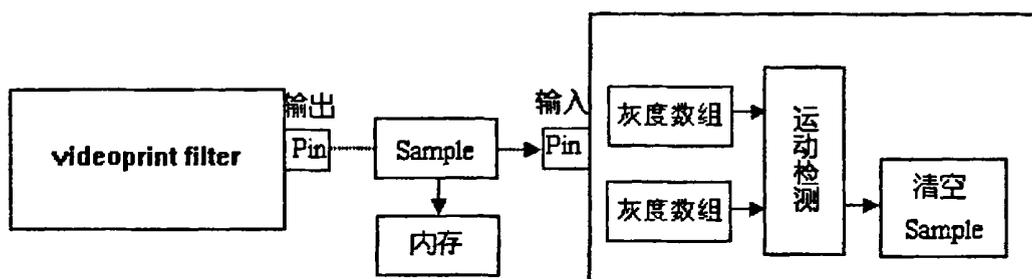


图 6-2 运动检测 Filter 算法实现结构图

服务器和客户端的报警通信可以采用多种方式。在 WLAN 环境中，通常使用 socket 嵌套字技术进行通信。但是，在第五章我们介绍了 Pocket PC 的发展现状，当前其硬件都内置了 GSM/GPRS 网络支持模块，利用稳定且覆盖面广的 GSM 网络进行通信将可以有效的克服 WLAN 作用域有限等固有缺陷。由于 GSM 报警功能可以利用多种第三方软件完成，在此，本文便不作更多阐述。

## 6.2 本章小结

这一章主要介绍监控系统的智能化理论技术。着重介绍了基于相邻帧差法的运动检测技术，采用提取相邻三帧求灰度均值的算法作为理论基础，并利用 DirectShow SDK 实现了运动检测的 Filter。最后提出一种以 GSM 网络为负载的系统报警机制。

## 第7章 总结与展望

无线视频监控是一项新兴的安防监控系统。经过数月的理论研究与技术试验,本人深刻体会到建立一个完整系统的复杂性,然而通过论文的撰写和系统的开发,自己的理论知识得到进一步丰富,眼界得以开阔,技术能力获得显著提升。同时也认识到无线视频监控系统无论是在市场潜力还是科学研究方面都有巨大的发展空间。当然,本系统只是一个用于试验的演示程序,并没有完成其完备的系统功能。因此,在本次研究基础上可以使之更加完善。本章所进行的总结与展望可以使之呈现出清晰的思路。

### 7.1 工作总结

本文首先介绍了数字视频监控系统的各项相关技术及系统组成,包括视频编编码解码技术、无线网络传输技术、嵌入式系统开发技术等众多前沿信息技术。本文就主要针对 IEEE802.11 无线网络下的监控问题讨论了多种理论和技术的选择,在之前的众多研究者所做的贡献之上,设计并实现了一套自己的解决方案。服务器端是在 Visual Studio 2005 开发平台在 WindowsXP 操作系统环境下实现的,客户端是在 Visual Studio 2005 开发平台在 Windows Mobile 虚拟机环境下实现的。所作的主要工作如下。

1、使用基于 IEEE802.11 无线网络标准的 WLAN 作为监控系统的网络负载,分析和研究了其技术原理和应用特点。利用无线接入点将客户端和服务端连接起来,深入思考并研究了多接入点的网络切换问题。

2、对服务器端的功能进行有效的分析,着重研究视频图像编码标准。最终使用 DirectShow 技术对摄像头的拍摄的视频进行采集,并编码成 MPEG-4 文件进行本地的储存;使用 Windows Media Encode 框架对 MPEG-4 文件以 WMV 格式进行实时再编码,并以视频广播的形式发送给客户端。

3、由于使用 Pocket PC 作为监控的客户端,为了更好的实现对监控视频的播放,本文深入研究了 Pocket PC 接收 RTP 数据包的功能。同时,针对本系统特殊需求,在 Windows Mobile 嵌入式系统环境下开发专用解码播放模块。

4、研究网络通信技术,包括 RTP/RTCP 协议以及广播技术。在不同平台之

间多媒体数据传输协议采用 RTP/RTCP 协议。同时对 WLAN 的跨区切换问题做了详细的研究。

## 7.2 改进与展望

由于时间相对紧张，系统构成复杂，本文所做工作也仅处于研究阶段，距离实用化还有一定距离。本系统的网络化、数字化、智能化程度还有待不断加强；同时，对不同平台间的整合，跨平台的设计，编码算法的优化还需要深入讨论。故后续的研究工作主要包括：

1、本系统服务器端首先将视频信号以 MPEG-4 压缩编码，这样可以方便对视频的特定帧进行处理，但本文对特定帧的处理仅仅停留在对运动目标检测的研究和讨论上，下一步还可以在此基础上，研究并实现运动目标的分割、定位、目标跟踪和特征提取。从而更进一步实现目标识别、行为理解、威胁估计和决策推理，实现监控系统真正的智能化。

2、在视频编解码方案上，本文采用了 MPEG-4 来实现视频图像的压缩，采用基于 MPEG-4 编码的 WMV 视频格式来实现视频的广播发布。可以考虑采用 H.264 来实现对视频数据的压缩。H.264 具有码率低、容错能力强、网络适应性强和应用覆盖面广等技术优势，是代表最新技术水平的一项国际标准。

3、本系统使用 Pocket PC 作为监控客户端，如能将监控平台扩展到 Symbian、Palm 移动操作系统环境下，同时向 TD-SCDMA 和 WCDMA 无线网络扩展。那么系统的应用空间将得到大幅度扩展。这涉及到软件复用、多环境互联、异构网络切换等问题，就需要在今后的研究中实现数据跨平台跨网络通信的问题。

## 参考文献

- [1] 陈锦山.无线局域网的现状与前景展望[J].商品储运与养护.2007, 29(5):54-55.
- [2] Microsoft Corporation.What's New for Developers in WindowsMobile 2003-based Pocket PC. 2003.6. <http://msdn.microsoft.com/library/default.asp>
- [3] 李阳, 王应雪. 基于网络的监控系统的建设[J]. 微电脑世界, 2003, (01).
- [4] 张明, 瞿朝成. 搭建无线网络监控系统[J]. 电脑知识与技术, 2008, (36).
- [5] 中国网. 平安城市建设带动视频监控市场迅速增长 [OL]. [http://www.china.com.cn/tech/txt/2007-05/01/content\\_8201229.htm](http://www.china.com.cn/tech/txt/2007-05/01/content_8201229.htm), 2007.5.1.
- [6] 汤宁. 主动监视技术研究[硕士学位论文]. 西安: 西北工业大学, 2007.
- [7] 李波. 数字视频监控系统的多一双眼睛看世界 [OL]. <http://tech.sina.com.cn/oi/2001-10-30/89889.shtml>, 2001.10.30
- [8] 张海涛, 张绍伟, 鄢仙玲. WCDMA 和 WLAN 的联合建设[J]. 电信工程技术与标准化, 2003, (10)
- [9] 牟晓隆. 通信世界网: 中电信加速布局 Wi-Fi 与移动宽带形成竞争.[OL]<http://it.sohu.com/20080521/n256986648.shtml>
- [10] 韩阳.多媒体数字视频监控系统的设计与实现[硕士学位论文], 成都: 电子科技大学, 2003.
- [11] Bruce E. Krell[美]. Pocket PC 开发指南[M]. 北京: 清华大学出版社. 2003, 3.
- [12] 王文庆, 毛文林, 张利飞. 基于 MPEG-4 的视频监控系统模型及实现[J]. 计算机工程. 2006, 32(11):237-239.
- [13] 高焕兵. C/S 模式视频监控系统的设计与实现[J]. 成都信息工程学院学报, 2003. 12, 386-389.
- [14] 洪小达. 多媒体计算机与数据压缩技术[M]. 北京: 中国国际广播出版社. 1999
- [15] 郭斌. 视频压缩中的熵编码研究[硕士学位论文]. 杭州: 浙江大学, 2007
- [16] 戴祥. USB 数字摄像平台的软件开发[硕士学位论文]. 天津: 天津大学, 2004
- [17] 金纯, 陈林星, 杨吉云. IEEE802.11 无线局域网[M]. 电子工业出版社. 2004
- [18] 王峰博, 崔慧娟. IEEE802.11b 无线局域网视频传输特性分析. 电视技术, 2004.2
- [19] Tiejun Wang, Proakis J.G, Zeidler J.R. Techniques for suppression of intercarrier interference in OFDM systems .Wireless Communications and Networking Conference.2005 IEEE. 13-17 March 2005, Volume 1 :39-44 .
- [20] Hsiao-Chun Wu. Analysis and characterization of intercarrier and interblock interferences for wireless mobile OFDM systems .Broadcasting.IEEE Transactions on, 2006, Volume 52 (Issue 2) :203-210 .
- [21] IEEE Std.802.11-1999, IEEE Standards for Local and Metropolitan Area Network, Part 11: Wireless LAN Medium Access Control(MAC) and Physical Layer (PHY) Specifications, 1999

- [22] 江汉红, 王征, 李庆, 梁学俊. 基于 IEEE802.11g 标准的 WLAN 性能分析与测试[J]. 武汉理工大学学报, 2005, (04).
- [23] Cali, F, Conti, M., Gregori, E. IEEE 802.11 protocol: design and performance evaluation of an adaptive backoff mechanism .IEEE Journal on Selected Areas in Communications, 2000, 18 (9) :1774~1786 .
- [24] Cornelia R, Vahid T. A construction of OFDM 16QAM sequences having low peak powers .IEE Trans Inform Theory, 2001, 475, 47(5) :2091~2094 .
- [25] 美国国家标准学会(ANSI). ANSI/IEEE 802.11b-1999/Corrigenda1-2002. Information Technology - Telecommunications and Information Exchange Between Systems - Local and Metropolitan Networks - Specific Requirements - Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications: Higher Speed Physic. 美国: ANSI, 2002
- [26] 汪卫国. 全球广播方式手机电视运营市场分析[J]. 通信世界, 2007, (24):
- [27] 盛彦瑾. 基于 Pocket Pc 的无线流媒体系统的研究与实现[硕士学位论文]. 武汉: 武汉大学, 2005
- [28] Tu-Chih Wang, Hung-Chi Fang, Liang-Gee Chen.Low Delay, Error Robust Wireless Video Transmission Architecture for Video Communication.IEEE 2005
- [29] Michael Blome, Mike Wasson. Core Media Technology in Windows XP Empowers You to Create Custom Audio/Video Processing Components .MSDN Magazine. 2002,
- [30] 陆其明. Windows Media 编程导向[M].北京:清华大学出版社, 2005.
- [31] Andy Wigley, Stephen Wheelwright. .NET compact Framework 技术内幕[M].北京:清华大学出版社, 2006. 1.12-13
- [32] 张文俊等主编.数字媒体技术基础[M].上海:上海大学出版社, 2007.3.70-79.
- [33] 马川, 刘志镜.视频实时传输的 RTP 实现.现代电子技术[J].2004 (20) .41-43.
- [34] (美)特南鲍姆(Tanenbaum, A.S.)著;潘爱民译.计算机网络:第四版(Computer Networks, Fourth Edition) [M].北京:清华大学出版社, 2004.8.
- [35] 温小明.无线视频监控系统的研究与实现[硕士学位论文], 上海:东华大学, 2006.
- [36] 陆其明, DirectShow 开发指南[M].北京:清华大学出版社, 2003.
- [37] 陆其明, DirectShow 实务精选[M].北京:科学出版社, 2004.
- [38] 张睿萍.基于 DirectShow 的数字视频监控系统的研究与实现[硕士学位论文]. 武汉: 武汉理工大学, 2008.
- [39] 邵 曦, 王绍棣.多点对点无线视频图像传输系统的研究和实现[J], 南京:南京邮电学院计算机科学与技术系, 2003.
- [40] 贺昕 李斌.异构无线网络切换技术.北京邮电大学出版社.2008.80-81
- [41] 刘乃安主编.无线局域网(WLAN)原理、技术与应用[M].西安电子科技大学出版社.2004 465-466
- [42] Bononi,L. ,Conti,M. ,Gregori,E. Runtime optimization of IEEE 802.11 wireless LANs performance .IEEE Transactions on Parallel and Distributed Systems, 2004, 15 :66~80 .
- [43] (美)斯托林斯(Stallings, W.)著, 林琪, 阎慧译.计算机网络与 Internet 协议和技术[M].北京:中国电力出版社, 2005.

- [44] 邹纯宏, 苑惠娟, 赵婷婷. 基于 RTP 与 DirectShow 的网络视频监控系统[J]. 国外电子测量技术, 2007, (03).
- [45] Schulzrinne H, Casner S, Frederik R, etc. RTP: A Transport Protocol for Real - Time Applications[A ]. Chae hoon lim. Information Security Applications [C ]. Jeju sland Korea: 5 th International Workshop. 2004
- [46] Alex Feinman. Hosting ActiveX Controls in Compact Framework 2.0 Applications
- [47] Intel PXA26X Processor Family Design Guide, March 2004 Intel Co. ,Ltd.
- [48] A Neri, S Colonnese, G Russo et al. Automatic moving object and background separation[J], Signal Processing, 1998, 66:219-232.
- [49] R Mech, M Wollborn. A noise robust method for segmentation of moving objects in video sequences[C] , In: Proc International Conference on Acoustics, Speech and Signal Processing, 1997, 41-45.
- [50] Ramesh Jain, H H Nagel. On the Analysis of Accumulative Difference Picture from Image Sequences of Real World Scenes. IEEE Trans. on PAMI. 1979, 1(2). 206~214.
- [51] F Moscheni, S Bhattacharjee, M Kunt. Spatial temporal segmentation based on region merging [J], IEEE Transaction on pattern analysis and machine intelligence, 1998, 20:897-915.
- [52] Ying Ren, Chin-Seng Chua, Yeong-Khing Ho. Motion detection with non-stationary background Image Analysis and Processing .2001, Proceedings, 11th International Conference. 2001, :78~83 .

## 致 谢

首先要特别感谢我的导师徐东平教授三年来的悉心指导。他严谨的治学态度、精益求精的工作作风让我受益匪浅，终身难忘。他在学术上的高深造诣、为人师表的风范和宽以待人的高尚品德，永远是我学习、工作和生活的榜样。在他的指导下，实验室形成了浓厚的学术研究氛围，这种氛围促使我不断进取，定期的实验室内的交流和探讨，使我获得很多。本文正是在此种研究氛围中得以完成的。在此特向导师致以最诚挚的敬意和最衷心的感谢。

另外，还要感谢所有老师和实验室同学在我研究过程中遇到困难时对我的种种帮助和鼓励，实验室内的友好、和谐和团结使我可以保持身心愉快，大大地提高了学习效率。其中特别需要感谢张睿萍师姐，她的无私帮助保证了系统功能可以如期实现。

感谢同寝室的刘鑫同学、王锋同学、胡林同学。和他们一起生活得三年，我生活开心，学习刻苦。他们在我失落的时候鼓励我，在我困难的时候帮助我，在我骄傲的时候鞭策我。在此衷心的感谢。

感谢方正国际软件有限公司的杨卫东经理，给与我在此公司实习的机会。在实习期内，我的理论技术知识和实际动手能力得到极大的增强。并感谢方正国际京瓷项目组的同事。

最后感谢我的父母在我整个攻读硕士研究生期间，尤其是在进行毕业设计过程中，给予我莫大的鼓励和支持，有了父母的坚强后盾我才可以全身心投入我的研究生活。再次谢谢所以给予我帮助的人们！