

基于 Android 的移动视频监控客户端^①

卜振江^{1,2}, 杨海波¹, 贾军营¹

¹(中国科学院 沈阳计算技术研究所, 沈阳 110168)

²(中国科学院大学 计算机与控制学院, 北京 100049)

摘要: 近年来, 移动通讯网络的迅速发展和 Android 系统智能手机的普及使得移动视频监控成为一个重要的研究方向. 在深入研究流媒体传输技术、视频编解码技术和 SIP 协议的基础上, 设计了一种具有语音通话功能的移动视频监控客户端, 并重点对基于 Android 平台的移动视频监控客户端进行研究与设计.

关键词: 移动视频监控; Android 平台; 流媒体技术; 语音通话

Mobile Video Monitoring Terminal Based on the Android System

BU Zhen-Jiang^{1,2}, YANG Hai-Bo¹, JIA Jun-Ying¹

¹(Shenyang Institute of Computing Technology, Chinese Academy of Sciences, Shenyang 110168, China)

²(School of Computer and Control Engineering, University of Chinese Academy of Sciences, Beijing 100049, China)

Abstract: In recent years, mobile video monitoring is an important research field because of the development of mobile communication network and the popularity of Android smart phone. This paper deeply studies the streaming media transmission technology, video coding-decoding technology and SIP protocol, on this basis, it designs a mobile video monitoring system owning voice function which mainly carries on the research and design of mobile video surveillance client on the basis of the Android platform.

Key words: mobile video monitoring; Android platform; streaming media technology; audio-based communication

1 引言

随着移动互联网络的快速发展和智能手机的处理能力越来越强,移动视频的需求正在爆炸式的增长,同时传统的视频监控已经不能满足用户随时、随地和根据需要进行实时监控与通信的需求. 基于手机等便携式设备的无线网络视频监控系统,将成为下一个视频监控领域的热点. 同时 Android 平台又以其系统开放、性能优越、用户群体多等优点成为研究设计移动视频监控客户端的首选.

为了满足人们在繁忙工作的同时,可利用智能手机远程随时、随地、实时对视频监控区域状况了解,并可以迅速做出分析与处理. 本文研究设计了一种基于 Android 平台的移动视频监控客户端. 该客户端结合了流媒体传输技术, 视频编解码技术和 SIP 语音通信技术, 不仅能实时接收服务器端发送的视频数据, 并解码播放, 还可以实时与视频采集智能终端进行语音通话.

2 系统设计方案的可行性及优势分析

首先, 随着我国无线 Wifi 网络和 4G 移动通信技术的普及以及 5G 网络的出现, 移动网络的传输速率也得到了非常大的提升, 已经完全满足移动设备上音视频数据传输的要求. 这就为研究与设计移动视频监控客户端提供了一定的技术可行性.

第二, 本文研究与设计的移动视频监控客户端是基于 Android 系统的. 对比其他移动端操作系统, Android 系统优势主要有:

(1) Android 是一种基于 Linux 的自由及开放源代码的操作系统. Android 系统的开放性不仅仅体现在对 Android 操作系统的授权使用上, 同时, Android 系统的上层应用程序可以自由地访问 Android 系统的各个 API.

(2) Android 系统具有丰富的多媒体类库, 并且这些类库都是开放源代码的, 可以很方便的对这些多媒

① 收稿时间:2016-07-29;收到修改稿时间:2016-10-10 [doi: 10.15888/j.cnki.csa.005768]

体类库进行扩展及二次开发,这就为移动视频监控系统的快速开发提供了便利条件。

(3) Android 系统的市场占有率非常高,据市场研究公司 IDC 的全球手机市场季度报告,2016 年第一季度,Android 系统的市场占有率为 89%,所以开发基于 Android 的移动视频监控系统具有较好的市场应用前景。

(4) Android 系统具有用户体验优越的可视化界面,可以把移动监控客户端的 UI 界面设计的非常漂亮、美观且简单易于操作。

第三,本文的设计方案有效的结合了音视频编解码技术、流媒体技术与语音通信技术。音视频编解码主要应用了 FFmpeg 开源框架,FFmpeg 是在 Linux 平台上开发的,而 Android 系统也是基于 Linux 系统开发的,所以比较容易根据业务需求将扩展好的 FFmpeg 框架移植到 Android 系统中;流媒体技术应用了当前比较成熟的 RTP、RTCP、RTSP 流媒体传输协议;语音通信技术对比了 SIP 协议和 H.323 协议,选取了简单灵活,分布式控制的 SIP 协议,中国电信集团也在其《中国电信家庭视频监控业务技术规范》中,推荐将 SIP 协议作为首选控制协议来实现。

最后,本文设计的移动视频监控系统相对比一般的移动视频监控系统具有如下优势:

(1) 可以随时随地灵活的在移动端查看清晰的视频监控画面,同时还可以实现实时语音通话与互动,这样可以很好的满足人们随时随地远程看护家里老人,小孩的需求和医生远程指导医疗服务的需求,广泛应用于智能家居安防领域和远程医疗领域。

(2) 该监控系统的扩展性非常强,视频采集端可以是固定的摄像头,也可以是带有摄像头的移动设备,可广泛应用车载移动视频监控系统中交通领域。

(3) 由于在监控系统中设计了具有保存、推送、拉取视频数据功能的流媒体服务器,可以在移动设备上实时回看视频监控画面。

3 移动视频监控关键技术研究

3.1 流媒体传输技术

3.1.1 流媒体

流媒体是指采用流式传输的方式在因特网上播放的媒体格式。流媒体的特点是:在播放前不需要全部下载完成,只需要缓存部分内容即可播放,在播放的

过程中,后台继续下载音视频媒体内容。这种对多媒体文件边下载边播放的流式传输方式不仅使延迟大幅度的缩短,极大的减少用户等待的时间,而且对系统缓存容量的需求也大大降低,非常适合应用于存储容量不大的手机系统。流媒体的关键技术是流式传输,流式传输又按播放方式不同分为两种:实时流式传输(Real-time Streaming)和顺序流式传输(Progressive Streaming)。两者性能对比如表 1 所示。

表 1 流媒体传输方式对比

	实时流式传输	顺序流式传输
服务类型	Media Server	HTTP Server
等待时间	时间短	时间长
播放控制	可以	不可以
带宽对播放视频的影响	影响比较大	对播放无影响
对于直播和多播的支持	支持	不支持

由于实时流式传输可以保证音视频信号的带宽和网络连接相匹配,我们就可以实时观看到监控视频。实时流式传输的突出特点是:它与普通的 HTTP 流式传输不同,它需要配置专用的流媒体服务器和相应的传输协议;流媒体流式传输还总是实现实时传输,这样它就非常适合现场事件,比如应用于实时监控、直播系统中,也支持随机访问。用户可以通过快进和后退按钮观看前面或后面的视频内容。而顺序流式传输只支持顺序下载,在某一时刻,用户就只能观看自己已经下载的部分视频内容,不可以快进观看已经下载的视频内容。

3.1.2 流媒体传输协议

流媒体系统各部分之间通过传输协议来协调管理整个流媒体通信过程。实时流式传输主要涉及到的协议有实时传输协议(Real-time Transport Protocol, RTP)、实时传输控制协议(Real-time Transport Control Protocol, RTCP)和实时流传输协议(Real-time Streaming Protocol, RTSP),这三种协议构成了实时流式传输的基础。

RTP 协议为那些传输实时数据的应用提供端到端的网络传输功能,包括在单播或多播环境中传输音视频或仿真数据。RTP 协议并不解决资源预留,也不保证实时服务的服务质量,数据的传输质量由 RTCP 协议保证。RTP 可以看成是传输层的一个子层。如下,图 1 显示了一个流媒体传输的典型协议体系结构。

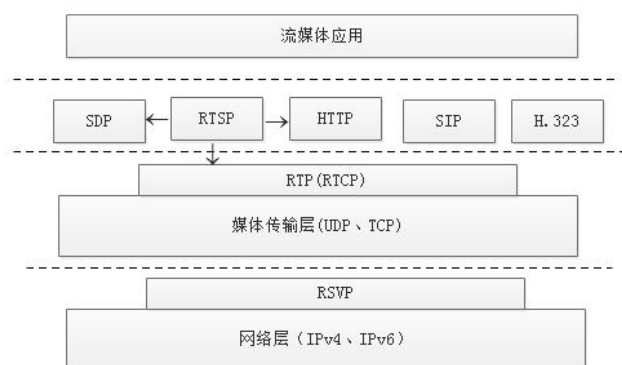


图 1 流媒体传输协议体系结构

RTP 基于 UDP 协议, 也有固定的封装形式, 实现端到端的实时传输。RTP 协议需要 RTCP 协议控制服务质量, 另外, 与 RTP 协议相关的协议还有 RTSP 协议 (Real-time Streaming Protocol) 和 RSVP 协议 (Resource Reservation Protocol)。

RTCP 主要负责服务质量的监控、反馈、流媒体之间的同步, 以及多播场景中各个成员的标识。在每一个 RTP 会话中, 各个参与终端会周期性的发送 RTCP 数据包。RTCP 数据包中主要包含了已发送的数据包个数、丢失的数据包个数等信息, 各个参与终端可以根据这些信息自动改变传输策略或载荷类型。RTP 需要 RTCP 协议的配合使用, 才可以达到最优的传输效率。

RTSP 协议属于应用层协议, 与 HTTP 协议的流控制协议类似, RTSP 协议规定了一对多应用程序有效地通过 Internet 传送多媒体数据的方式。RTSP 协议的语法也类似于 HTTP 协议, 但不同的是 RTSP 协议是有状态的协议, 而 HTTP 协议是无状态的协议。RTSP 通过维护一个会话以维护状态转换过程, 其默认端口号为 554, 默认承载协议为 TCP 协议。

3.2 FFmpeg

FFmpeg 是一个集音视频编码、解码、转换、录制功能为一体的开源框架。FFmpeg 是在 Linux 平台下开发的, 所以继承了 Linux 平台高性能的优点。且在长期的开发中, 开源社区的各种开发者和程序爱好者们又对编码和解码的算法进行了优化, 其性能在所有的开源组件中基本是最好的, 再加上它对系统资源的需求又比较小, 因此非常适合移植到 Android 系统中, 被移动监控客户端解码模块所使用。FFmpeg 的主要组成部分如表 2 所示。

表 2 FFmpeg 的主要组成部分及描述

组成部分	描述
libavcodec	包含了对音频、视频数据进行编码和解码的所有函数, 并且支持常见的音频、视频数据的编解码工作。
libavformat	该库用来分析音频和视频的格式, 在解码之前需要分析视频格式, 然后交由解码库进行解码。
ffplay	是 FFmpeg 自带的播放器, 用来对解码以后的视频内容进行播放。
ffserver	是一个能够支持 RTP、RTSP、HTTP 等多种协议的多媒体流服务器。它不仅可以作为单独的服务器使用, 实现视频点播服务, 而且可以与 ffopeg 互相结合, 实现视频直播服务。
ffmpeg	它是用于转换视频格式的工具。在 FFmpeg 下, 可以根据这个命令行工具对不同的视频格式进行转换。
libavutil	工具函数库, 包含了一些重要的公共函数, 基本每个工程中都需要包含这个工具函数库。
libavswscale	主要用于对图像的颜色空间进行转换、缩放等处理。

3.3 SIP 通信协议

SIP (Session Initiation Protocol, SIP) 是会话发起协议, 它是一个基于文本的应用层控制协议, 由 IETF 在 1999 年提出来的。它是用于建立、修改和终止一个或者多个基于互联网平台的多媒体会话业务。现在 SIP 协议在实时聊天、视频电话、视频会议这些领域都应用非常的广泛。SIP 在三个方面支持创建、终止多媒体通信:

- (1) 用户可用性: 决定客户是否有这个意愿加入此次的会话通信;
- (2) 用户协商: 通信双方对数据参数和数据类型的确定;
- (3) 呼叫创建: 呼叫双方通过会话参数的传递进行建立。

4 视频监控系統总体设计

本文提出的移动视频监控系統是基于 C/S 架构, 主要分为三个主要模块: 视频采集端、服务器, 移动监控客户端。移动视频监控系統的设计总框架如图 2 所示。

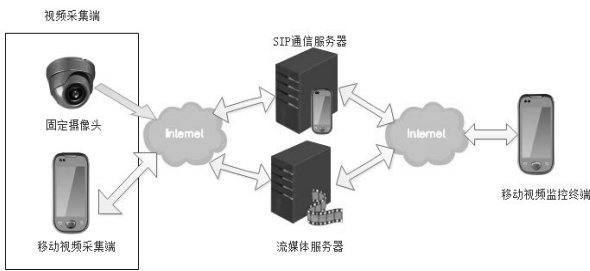


图 2 移动视频监控系统的总体设计框架

(1) 视频采集端: 主要负责监控区域的视频数据的采集, 压缩编码后上传送给服务器. 为了适应不同的应用场景, 视频采集端又分为固定摄像头采集端和移动设备采集端. 固定摄像头采集端主要由摄像头, 服务器监控终端组成, 作为传统的视频采集方式, 摄像头负责采集视频并上传到服务器; 移动视频采集终端是基于 Android 操作系统的智能手机上开发的应用软件, 是一种新型的视频采集方式, 通过 Android 提供的音视频 API 和开源的音视频编解码解决方案, 实现音视频的采集和编码, 然后通过移动无线网络上传到服务器.

(2) 服务器: 包括了流媒体服务器和 SIP 通信服务器两部分, 它是音视频采集端和监控终端的沟通的纽带. 流媒体服务器实现根据用户请求对视频采集端的视频数据进行转发; SIP 信令服务器其主要负责接收与处理移动设备采集端和移动监控终端发来的 SIP 请求信令, 是系统通信协议实现的关键部分.

(3) 移动监控客户端: 主要部署在 Android 移动平台上, 通过用户操作的界面, 可以实现对监控视频的随时查看, 以及设备的控制, 信息查询与配置等, 还可以进行语音通话, 方便及时相互通信.

5 移动监控客户端的设计

由于本文的重点是对移动监控客户端的研究和设计, 以下内容主要是针对移动监控客户端各个子模块研究和设计. 由于移动视频监控终端设备是用户直接交互的对象, 须要求界面美观、操作简单、功能稳定. 其设计结构主要由三层构成: 视图层、控制层和业务层, 如图 3 所示.

5.1 视频解码模块

由于网络带宽和存储设备的限制, 视频数据传输过程之前经过压缩编码. 所以移动监控客户端在经过

视频数据的接收和组包之后, 就可以获得一个视频帧, 之后就要对其进行解码. 视频解码模块是视频处理中的核心部分.

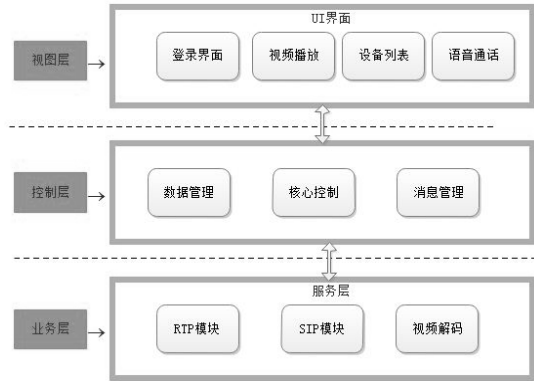


图 3 移动监控客户端的 MVC 结构图

本文设计的视频采集终端采用的编码格式是 H.264 格式, 编码之后视频帧的分辨率是 352×288, 图像格式为 YuV420. 视频数据解码模块主要负责从缓冲区中读取视频帧, 将其进行解码, 然后依据手机分辨率调整视频帧的分辨率, 并将其转换成 RGB 格式后, 把视频数据交送视频显示模块.

本模块视频的解码和分辨率转换通过调用 FFmpeg 底层解码函数实现, 接下来将详述下 FFmpeg 的解码流程, 如图 4 为 FFmpeg 的视频解码流程图.

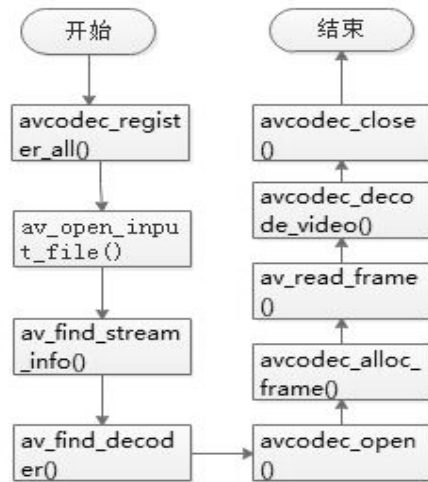


图 4 FFmpeg 解码流程图

(1) avcodec_register_all(): 完成 FFmpeg 支持的编解码格式注册.

(2) `av_open_input_file()`: 打开本地或网络服务端的多媒体文件.

(3) `av_find_stream_info()`: 分别找音/视频的流索引 ID 信息.

(4) `avcodec_find_decoder()`: 根据 ID 信息寻找相应的解码器.

(5) `avcodec_open()`: 打开解码器.

(6) `avcodec_alloc_frame()`: 开辟一片类型为 AVFrame 的缓存, 来存储解码后的帧数据.

(7) `av_read_frame()`: 从码流中读出帧数据, 为音频流或者视频帧.

(8) `avcodec_decode_video()`: 解码视频帧.

(9) `avcodec_close()`: 解码完后, 释放解码器.

5.2 语音通话模块

研究对比了多种语音编解码器, 发现 G.729 编解码器具有非常好的语音质量和非常短的时间延迟, 因此, 在语音通话模块中我们采用 G.729 编解码. 按照语音通话模块的实现进程进行划分的话, 音频通话模块可以划分为语音数据的采集、音频数据的编码、RTP 实时传输、语音数据接收、语音解码、播放等部分组成. 在移动监控客户端上拨打电话成功后, 即发起 sip 会话请求成功后, Android 系统通过本身所带的麦克风功能进行数据采集原始的语音数据, 再进行相对于复杂程度较低的 G.729 编码, 编码后减少了数据的存储空间. 然后再将这个编码后的数据流进行 RTP 封装, 接着实时发送出去. 在发送的同时也实时接收着 RTP 数据包, 最后, 通过 G.729 转码, 得到原始的音频码流, 让手机的通过播放器播放出来, 这样, 就实现了语音通话的功能.

6 结语

本文在对移动流媒体传输、FFmpeg 框架和 SIP 协议等关键技术研究的基础上, 根据实际项目需求, 首先设计了移动视频监控系统的整个系统架构, 并侧

重设计了移动视频监控客户端, 能够较好地提供远程监控服务和语音通话功能, 因此该客户端设计具有一定的实际意义和广泛的市场应用前景.

参考文献

- 1 李江. 基于 Android 的 4G 网络移动高清视频监控系统关键技术的研究[硕士学位论文]. 杭州: 浙江大学, 2016.
- 2 董杰, 辛吉涛, 连捷. 基于 Android 系统的 H.264 视频直播技术研究. 电视技术, 2015, 39(4): 11-15.
- 3 汪雨秋. 基于 Android 的视频监控终端系统的设计与实现[硕士学位论文]. 武汉: 华中科技大学, 2014.
- 4 龚虹瑞. 基于 Android 的移动视频监控[硕士学位论文]. 成都: 西华大学, 2015.
- 5 丁峰, 刁鸣. FFmpeg 的音视频格式转换设计. 应用科技, 2013, (2): 11-4.
- 6 Fu SH, Fu HQ, Wang Z. Design of mobile phone network video surveillance system based on 3G. Computer Applications, 2011: 1-6.
- 7 Ratabouil S. Android NDK Beginner's Guide - Discover the native side of Android and inject the power of C/C++ in your applications. Packt Publishing, 2012: 6-16.
- 8 Zeng H, Che HX, Cao GL. Design of mobile video surveillance based on Android. 2012 International Conference on Computer Science & Service System (CSSS). IEEE. 2012. 765-768.
- 9 杨飞, 陈德艳, 黄国宏. 基于 Android 智能终端的移动视频监控研究. 应用开发研究, 2013: 1-5.
- 10 Michael. 模拟视频监控发展现状及趋势探讨. 中国安防, 2013, (9): 68-87.
- 11 辛吉涛. 基于 Android 平台的移动视频监控系统的研究[硕士学位论文]. 大连: 大连理工大学, 2015.
- 12 黄静, 朱欣远. 4G 移动通信关键技术及其展望探究. 中国新通信, 2014, (3): 101-118.