

一种基于 Android 智能手机的视频上传软件

陈伟军¹, 郑睿², 虞鸿²

(1. 万向钱潮股份有限公司, 浙江 杭州 311215;

2. 浙江工业大学 信息工程学院, 浙江 杭州 310023)

摘要: 提出了一种基于 Android 智能手机的视频监控系统视频上传客户端软件设计方案, 介绍了整个系统的框架和客户端软件开发过程。基于 Android 操作系统平台, 通过扩展会话初始化协议(SIP)实现信令控制, 采用系统多媒体框架实现视频采集与编码, 使用 LocalSocket 本地映射获取编码视频数据, 并以实时传输协议(RTP)实现数据传输。

关键词: Android; 视频上传; 会话初始化协议; LocalSocket; 实时传输协议

中图分类号: TP393

文献标识码: A

文章编号: 1674-7720(2013)24-0050-03

A video upload software based on Android smart phone

Chen Weijun¹, Zheng Rui², Yu Hong²

(1. Wanxiang Qianchao Co., Ltd., Hangzhou 311215, China;

2. College of Information Engineering, Zhejiang University of Technology, Hangzhou 310023, China)

Abstract: A design of video uploading client software for video surveillance system based on Android smart phone technology is put forward, and the framework of the whole system and client software development process are introduced. Based on the Android operating system platform, the client extends the session initiation protocol (SIP) for signaling control, achieves video capture and encoding by using multimedia framework, gets the encode data with LocalSocket, and uses the real-time transport protocol (RTP) for data transmission.

Key words: Android; video upload; session initiation protocol; LocalSocket; real-time transport protocol

视频监控^[1]以其直观、准确、及时和信息内容丰富而广泛应用于许多场合。国内外很多公司都投入了很大的精力在视频监控系统的研究上, 如国外的博世、松下, 国内的海康和大华等, 他们研发供应监控产品, 并提供解决方案。近年来 Android 操作系统^[2-3]发展迅速, 拥有广大的用户群体。当前市场上基于 Android 的视频上传软件, 通过调用第三方库实现视频数据编码, 由 CPU 实现编码数据算法, 资源占用率高。本文设计了一种基于 Android 智能手机的视频上传客户端软件, 以 SIP^[4]协议作为控制信令, 采用手机自带摄像头录制视频, 基于系统多媒体框架实现视频数据编码, 并上传数据到服务器。

1 视频监控系统框架

系统由视频播放客户端、服务器和视频上传设备端 3 部分组成。视频播放客户端包括手机客户端和 PC 客户端, 播放设备端上传的视频; 服务器由一个 SIP 服务器和一个媒体转发服务器组成, SIP 服务器负

责对设备端和播放客户端的管理, 媒体转发服务器实现媒体数据转发, 将设备端的视频数据转发给播放客户端; 设备端通过 WiFi 或者 3G 连接网络, 登录到 SIP 服务器, 可以接收播放客户端的视频邀请并与媒体转发服务器建立连接, 上传视频到媒体转发服务器。系统框架图如图 1 所示, 实线为 SIP 信令流, 虚线为 RTP 媒体流。

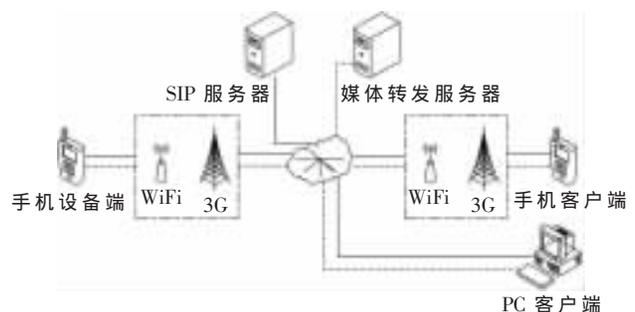


图 1 视频监控系统框架

网络与通信 Network and Communication

2 客户端软件设计

客户端设计包括顶层用户界面设计和底层的功能模块设计。用户界面用于实现人机交互;功能模块主要有信令控制模块、视频采集编码模块和视频上传模块,分别实现客户端与 SIP 服务器间交互、视频数据采集编码和将视频上传到媒体转发服务器的功能。

2.1 界面设计

用户界面主要有登录界面、等待视频邀请界面和视频采集界面,是通过调用应用程序框架层的 API 接口和 View 组件,配合 XML 布局文件实现的 Activity。在登录界面输入账号和密码,登录成功后跳转到等待视频邀请界面;在等待视频邀请界面,用户可以选择登出服务器;若收到视频邀请消息,可以跳转到视频采集界面;在视频采集界面进行视频采集、编码与上传;若收到结束邀请消息,则断开与媒体转发服务器的连接,回到等待视频邀请界面,等待下一次的视频邀请。

2.2 信令控制模块

Android 手机客户端与 SIP 服务器之间的信息控制采用 SIP 协议,采用 XML 扩展 SIP 消息体。SIP 消息实现的功能有设备注册、设备保活、邀请视频和结束邀请,如图 2 所示。



图 2 客户端与 SIP 服务器的信令交互过程

设备注册时,客户端向服务器发送两次 REGISTER 消息,第一次客户端向服务器发送 REGISTER 消息,服务器回复带有密文消息体的 200 OK 消息;客户端解析密文,通过 REGISTER 将解析后的密文发回服务器,服务器回复 200 OK 表明注册成功。注册成功后,客户端需要向服务器发送保活信息。客户端发送 REGISTER 消息,消息体中是保活请求信息;服务器回复带有保活响应消息体的 200 OK 消息。视频邀请时,服务器向客户端发送 INVITE 消息,消息体中有媒体转发服务器的 IP 和数据接收端口号;客户端回复 200 OK,并在消息体中设置上传视频的相关信息;服务器准备就绪后,向客户端发送 ACK 消息;客户端回复 200 OK 表示客户端准备就绪,可以进行视频上传。结束视频邀请时,服务器向客户端发送 BYE 消息,客户端回复 200 OK 消息表示收到结束视频请求。

2.3 采集编码模块

由于网络传输带宽的限制,视频需经过编码发送,本文采用 H.264 编码标准^[5]。Android 系统 MediaRecorder 类支持的文件封装格式有 AAC ADTS、AMR NB、AMR WB、MP4、RAW_AMR、3GPP,本文采用 MP4 封装格式进行研究。MP4 封装格式^[6]基于 QuickTime 容器格式定义,媒体描述与媒体数据分开,目前被广泛应用于封装 H.264 视频和 ACC 音频,是高清视频的代表。

本文采用 MediaRecorder 录制视频,基于 Android 系统的多媒体框架进行视频 H.264 编码,将录制的视频流映射到本地 LocalSocket 上,获取编码数据。为了获取 H.264 码流,通过启动两次 MediaRecorder 实现。第一次启动,设置视频录制相关参数,并将录制视频以 MP4 格式保存到当前应用目录下,从 MP4 文件的尾部找到 SPS、PPS 和视频数据存放起始位置;第二次启动后将录制的视频映射到 LocalSocket 的发送端,在 LocalSocket 的接收端获取视频流,并通过之前第一次启动 MediaRecorder 得到的视频存放位置从视频流中截取 H.264 码流,每次提取一帧数据,由视频上传模块发送给媒体转发服务器。H.264 码流获取流程如图 3 所示。

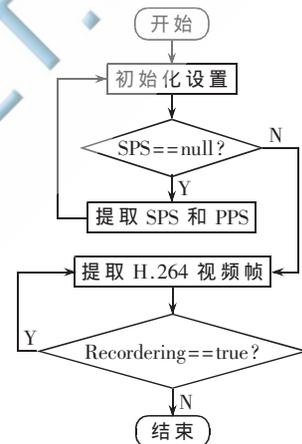


图 3 H.264 码流获取流程

2.4 视频上传模块

为了拥有较好的实时性,视频数据基于 RTP 协议传输,根据 RFC3984 协议进行操作。手机底层硬件是按照相同速率进行视频采集的,故每一帧的停留时间是相同的。如果采用编码后直接打包发送的策略,数据大的帧其分片包多,此期间发包速率高;数据小的帧分片包少,此期间发包速率低。可见此种发包机制的发包速率不稳定,包发送速率是由帧的大小决定,这样会出现发包高峰期,造成网络阻塞,从而导致丢包率升高,影响了接收端的图像质量。

针对这一情况,本文设计了一种平稳发包策略,通过开辟相应的发送缓存,以较平稳的速率来发送视频数据包。缓存模仿队列进行设计,采用先进先出的原则,实现了一个缓存链表。开启两个线程,一个线程负责将分片后的数据包从缓存尾部放入缓存,另一个线程负责从

欢迎网上投稿 www.pcachina.com 51

网络与通信 Network and Communication

缓存头部提取数据包进行发送。缓存通过设计两个类实现:StreamBufNode.java 类和 StreamBuf.java 类。其中 StreamBufNode.java 类定义了缓存链表节点的数据类型和相关操作方法;StreamBuf.java 类定义了整个缓存链表的数据结构、加入节点到链表和从链表取出节点等操作方法。发包的缓存示意图如图 4 所示。

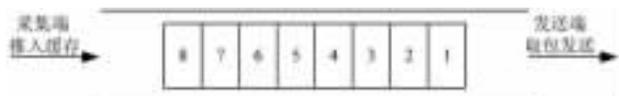


图 4 发包缓存

3 效果演示

本客户端在 HTC A310e (Android2.3.5) 智能手机上实现,网络环境为 WiFi,通过手机播放客户端播放上传视频,如图 5 所示。图 5(a)为视频上传客户端软件视频采集界面,图 5(b)为手机播放客户端视频播放画面,视频经过媒体转发服务器转发。



图 5 客户端效果演示

本文设计实现了一种基于 Android 智能手机的视频

监控系统视频上传客户端软件,并介绍了视频监控系统的整体框架和视频上传客户端软件设计方案。主要介绍了客户端软件的用户界面和功能模块设计,包括指令控制模块、采集编码模块和视频上传模块。软件基于日常使用的 Android 智能手机开发,移动性强,成本低。未来是移动的世界,基于智能手机设备端视频监控软件的设计和实现具有一定的现实意义。

参考文献

- [1] 周毅.基于 Android 系统的视频监控客户端软件的设计与实现[D].浙江:浙江工业大学,2012.
- [2] 杨丰盛.Android 应用开发揭秘[M].北京:机械工业出版社,2010.
- [3] 金仙力,陈晶晶.Android 平台上的实时图像采集与远程存储系统设计[J].无线互联科技,2012(10):64-65.
- [4] IETF.RFC 3261 SIP: Session initiation protocol[EB/OL]. [2002-07-19].<http://www.ietf.org/rfc/rfc3261.txt>.
- [5] 周恒.H.264 的研究与软件实现[D].南京:南京邮电学院,2004.
- [6] 周瑾,支铮,宋利.流媒体应用中 TS 和 MP4 格式分析[J].信息技术,2007,31(7):16-19.

(收稿日期:2013-09-30)

作者简介:

陈伟军,男,1968 年生,高级工程师,主要研究方向:计算机应用。

郑睿,男,1988 年生,硕士研究生,主要研究方向:移动多媒体通信。

虞鸿,男,1988 年生,硕士研究生,主要研究方向:网络与多媒体通信。