

# 一种基于 Android 手机的移动视频直播系统的设计与实现

杨名阳<sup>1</sup>, 张志刚<sup>2</sup>, 叶保留<sup>1</sup>

(1 南京大学 计算机软件新技术国家重点实验室, 江苏 南京 210046;

2 国家电网公司运营监控中心, 北京 100031)

**摘 要:** 无线网络及移动计算技术的发展为面向新型网络环境的在线视频直播提供了机遇, 但网络的动态性及流服务的实时性也对系统实现带来挑战. 针对上述问题, 本文设计并实现了一个基于 Android 手机的移动视频直播系统, 该系统支持用户通过智能手机内嵌摄像头实时采集现场视频信息, 并通过 Wi-Fi 或 3G 等上传视频流至服务器, 提供在线直播或点播服务. 文中详细描述了视频采集、视频发布处理和视频播放等核心功能的设计与实现机制, 并从视频编码和网络传输两个方面对系统性能进行了分析.

**关键词:** 视频直播系统; 移动环境; Android; H. 264; RTMP

**中图分类号:** TP302

**文献标识码:** A

**文章编号:** 1000-7180(2013)04-0164-05

## The Design and Implementation of Smartphone Based Mobile Live Video Streaming System

YANG Ming-yang<sup>1</sup>, ZHANG Zhi-gang<sup>2</sup>, YE Bao-liu<sup>1</sup>

(1 National Key Laboratory for Novel Software Technology, Nanjing University, Nanjing 210046, China;

2 Operation Monitoring Control Center of State Grid, Beijing 100031, China)

**Abstract:** The Advances in wireless communication and mobile computing techniques provide new venues to support novel live streaming systems. However, the dynamics of network as well as the realtime requirement of streaming service impose additional challenges on the implementation. In this paper, we design and implement a mobile live video streaming system over Android based smartphones. The implemented system enables users to capture the on-spot video with the video camera embedded in smartphone and upload to the media server via either WiFi or 3G networks, providing live and/or on-demand streaming services to remote users. We describe the design and implementation mechanism of the key components involved in the system, including video capturing/encoding, video processing/publishing and video playback. In addition, we also evaluate and compare the encoding and transmission performance of the system over different scenarios.

**Key words:** live video streaming; mobile environment; android; H. 264; RTMP

### 1 引言

视频直播是借助流媒体技术和网络通信技术, 通过采集、编码、发布和播放等步骤来实现直播的一种方式. 近年来, 无线通信网络的发展及移动便携设备的流行为构建面向移动环境的实时视频直播系统提供了新的机遇. 一方面, 无线接入点的密集部署及

带宽能力的提高为移动视频流传输提供了网络保障; 另一方面, 移动便携设备处理能力的提高及内嵌视频设备的普及为移动视频直播的实现提供了设备支撑. 用户可以使用智能手机来采集图像, 并编码成视频流, 通过移动通信网络上传至服务器, 再由服务器为终端用户提供视频直播服务. 理论上, 在任何移动通信网络覆盖的地方, 用户都可借助随身携带的

收稿日期: 2012-07-15; 修回日期: 2012-08-21

基金项目: 国家自然科学基金项目(61170069)

智能手机随时发起一次视频直播.因此,基于智能手机的视频直播系统可极大降低视频直播的硬件成本,且易于部署、实时性强.

本文设计并实现了一个基于 Android 智能手机的视频直播系统,图 1 描述了系统结构.

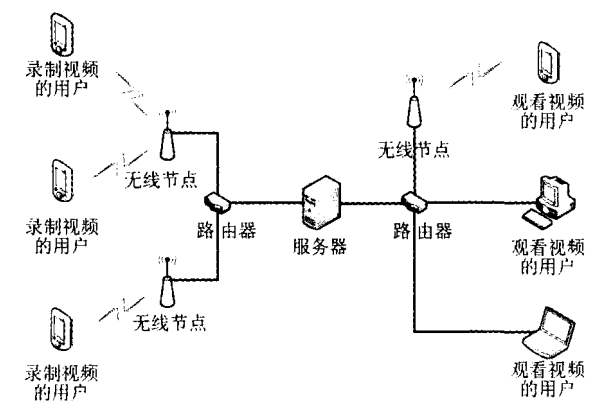


图 1 系统架构

2 系统软件结构

整个系统由视频采集模块、视频发布处理模块、视频播放模块等模块组成,图 2 描述了系统的软件结构及模块交互关系图.

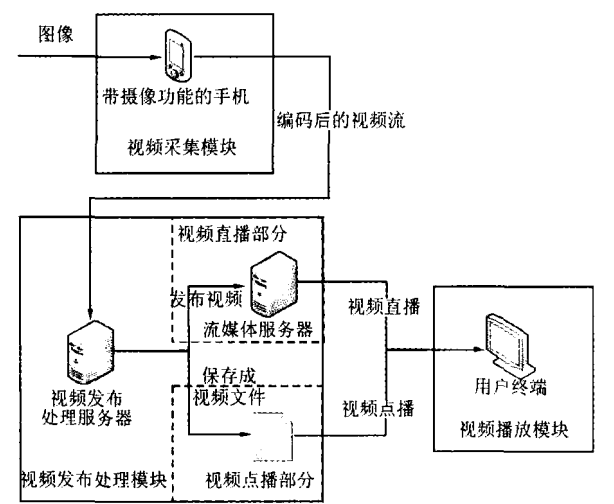


图 2 系统软件模块结构图

视频采集模块驻留于智能手机上,将摄像头采集的图像信息按照一定的格式编码成视频流,并以“边录制边上传”的形式实时上传至远程流媒体服务器端的视频发布处理模块.视频发布处理模块接收到视频流之后,作如下两种形式处理:一是将视频流按照一定的流媒体传输协议发布至流媒体服务器,由流媒体服务器向终端用户提供视频直播服务;二是将视频流按照媒体格式组织成视频文件保存到流媒体服务器中,以便在视频直播结束后向用户提供基于 Web 的视频点播服务.视频播放模块位于用户

终端,通过流媒体服务协议与服务器端的视频发布处理模块交互,获取在线视频直播流列表和视频点播列表,并选择合适媒体源进行播放.

流媒体服务器接收到视频发布程序发送的 H. 264 解码信息和 H. 264 码流以后,可向用户提供基于 RTMP 协议的流媒体视频直播服务.

3 软件模块设计与实现

3.1 视频采集

视频采集模块 VideoClient 基于 Activity 组件实现,其核心功能是构建面向智能手机内置摄像头的视频流接收机制,并向用户提供可视化界面,与远程服务器动态交互,实时发布实时视频流. Activity 是 Android 操作系统的内嵌图形化编程组件,每个 Activity 提供给用户一个用于交互的屏幕界面,在 Activity 中用户可以执行诸如拨打电话、发送电子邮件、查看地图等操作. Activity 之间可以相互跳转,从而实现屏幕界面切换和交互功能改变.为实现屏幕布局管理,每个 Activity 在创建时都需要提供一个后缀名为 xml 的界面布局文件来进行布局控制. VideoClient 由 MainActivity、InfoActivity、RecordActivity 和 OptionActivity 等 4 个功能界面组成.图 3 描述了 VideoClient 的结构图,其中 MainActivity 是主界面,负责在各功能界面间切换.

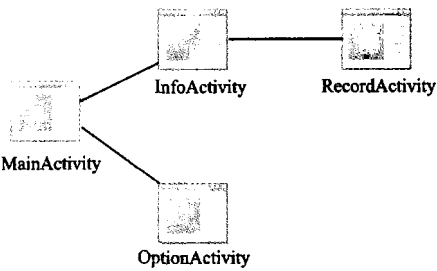


图 3 视频采集模块结构图

OptionActivity. 提供上传服务器地址的输入功能.

InfoActivity. 提供智能手机与远程媒体服务器的交互功能. 用户在其中输入待上传视频流的元信息(包括上传者信息及视频描述信息)并发送给服务器,随后跳转到 RecordActivity.

RecordActivity. 为上传者提供视频流预览界面,用于显示录制视频的即时画面.

MediaRecorder 默认将录制的视频保存成本地文件,但在视频直播需要将视频流实时推送到远程服务器,并在服务器端转发和保存成文件.

具体实现时,RecordActivity 通过一个线程来

处理视频录制工作. 该线程在 RecordActivity 创建以及预览视图开启后创建. 当用户切换到其它应用程序时, 该线程自动终止, 同时停止 MediaRecorder 并断开 HTTP 连接和 LocalSocket 连接, 视频流上传也随之停止.

### 3.2 视频发布

视频发布模块基于 J2EE 实现, 驻留于远程流媒体服务器上, 负责接收智能手机上传的视频流并提供直播和点播服务. 其模块结构如图 4 所示, 由 VideoServlet、VideoPublisher、MySQL 数据库和 .h264 视频文件管理器等四部分组成, 其中 MySQL 数据库保存视频流的元信息.

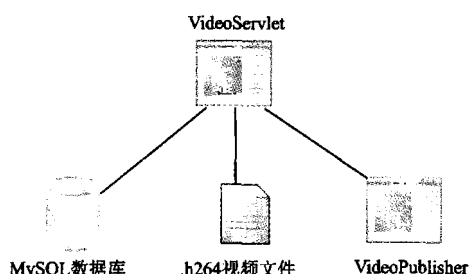


图 4 视频发布模块结构图

VideoServlet 通过继承 HttpServlet 实现的 servlet 类, 负责与智能手机的视频采集模块交互, 接收上传视频流的元信息及其视频流数据. VideoServlet 接收到 Android 手机发送的 Post 请求后, 首先判断 Post 的是视频流元信息, 还是 H. 264 码流. 如果是前者, 将其保存到数据库中, 并记录其起始时间戳; 如果是 H. 264 码流, 则转发给 VideoPublisher 进行发布并将其保存成后缀名为 .h264 的视频文件, 文件名格式为“用户-描述-时间戳”.

为支持视频点播, 还须在文件头部加入 H. 264 码流的解码信息, 包括序列参数集 (Sequence Parameter Set, SPS) 和图像参数集 (Picture Parameter Set, PPS), 解码信息包含 H. 264 编码时的 profile、level、width、height 等信息<sup>[10]</sup>. 由于 MediaRecorder 编码需要硬件支持, 而不同型号设备的 SPS 和 PPS 不尽相同, 因此需要预先采集一段测试视频来分析获取 SPS 和 PPS. VideoServlet 的工作流程如图 5 所示.

VideoPublisher 负责将上传到流媒体服务器 H. 264 码流按照一定的流媒体传输协议发布给视频观看用户. VideoPublisher 通过 C 语言开源库 RTMPDump (librtmp. lib) 编写实现, 用于支持创建 RTMP 帧. 当发送视频数据 (包括 AVC 序列头部和 NALU) 时, 需将 m\_packetType 设置为 RTMP\_

PACKET\_TYPE\_VIDEO, 表示发送的是包含视频数据的 RTMP 帧. 此外, 还需修改 m\_body 和 m\_nBodySize, 分别设置为存有要发送数据的缓冲区和要发送数据的长度. 其中 m\_body 的前五个字节需要特殊处理, 第一个字节的前四个比特是帧类型 (Frame Type), 应该设置成 0x1, 表示是 AVC 帧 (seekable), 后四个比特是 Codec ID, 应设置成 0x7, 表示编解码器是 AVC; 第二个字节是 AVC Packet Type (此时 Frame Type 为 0x1), 发送 AVC 序列头部时应设置成 0x00, 而发送 NALU 时需设置成 0x01; 第三至第五个字节是 Composition Time, 当 AVC Packet Type 取 0x01 时设置为时间戳, 否则全置为 0<sup>[8]</sup>.

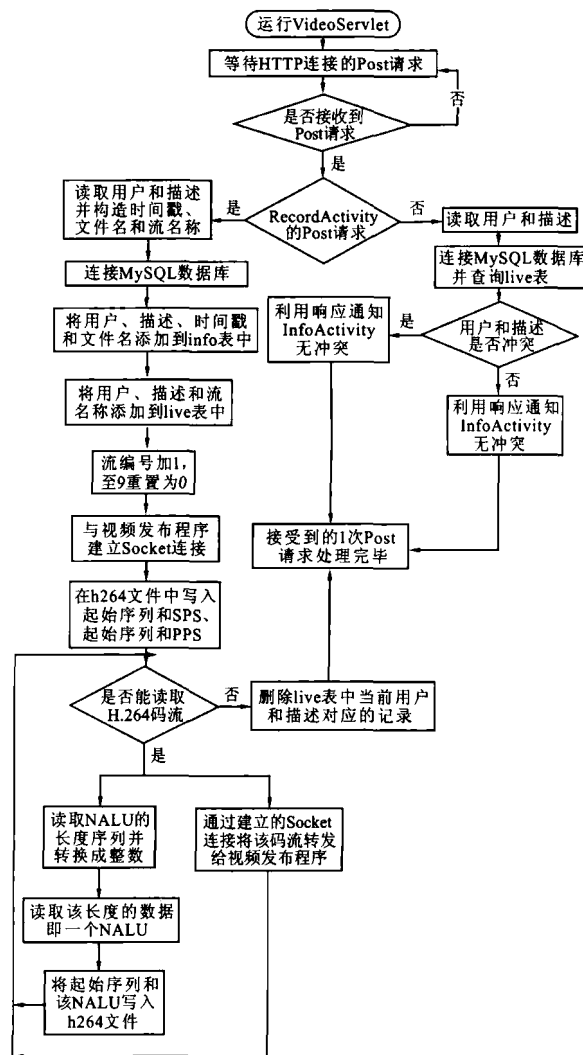


图 5 VideoServlet 流程图

VideoPublisher 的流程如图 6 所示. 为支持同时发布多个 H. 264 码流, serverSocket 接收到 Socket 连接请求时, 启动一个新线程来完成当次的视频流发布工作. 每个线程都调用 Publish 函数, 该函数的参数 (lpParameter) 包括视频流发布地址和

接收到的 Socket,并通过该函数与流媒体服务器建立 RTMP 连接.随后接收封装有 AVC 序列头部<sup>[10]</sup>的 RTMP 帧,用于解码.将 AVC 序列头部封装进一个 RTMP 帧并发送给 Wowza 服务器后,即可开始发送 NALU. NALU 通过 Publish 函数参数中的 Socket 来获取.需注意的是,每次构造封装有 NALU 的 RTMP 帧时,都要设置 RTMP 帧的 m\_nTimeStamp 属性(即时间戳),时间戳可以是相对时间戳,如从 0 开始,每发送一个 RTMP 帧就以常量递增.视频直播流发布结束后将断开与 Wowza 服务器的 RTMP 连接及与 VideoServlet 的 Socket 连接,并终止相应视频直播发布线程.

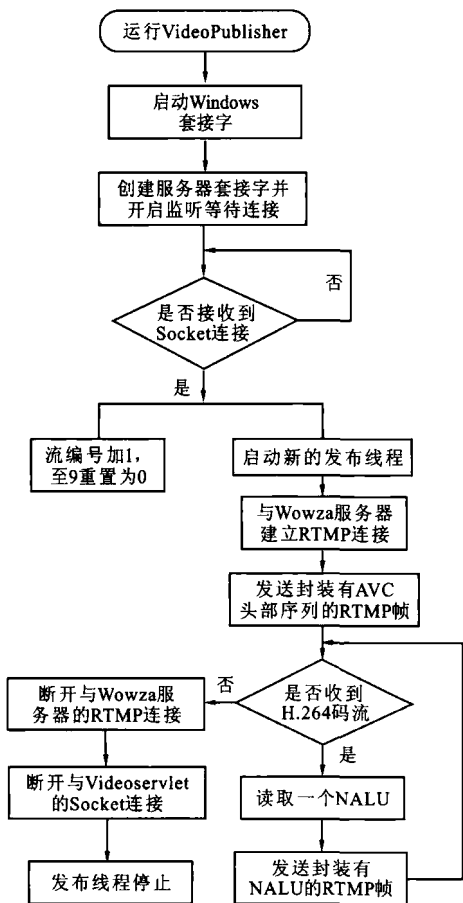


图 6 VideoPublisher 流程图

3.3 视频播放

视频播放模块为视频观看用户提供客户端播放软件模块,负责接收来自流媒体服务器的直播/点播视频流.由于流媒体服务器采用 H. 264 格式和 RTMP 协议编码和发布视频流,因此任何支持上述编码和传输协议的视频播放器都可播放直播视频流.

4 系统性能分析

视频直播系统的服务质量与系统性能包括画面清晰度/流畅度、播放延迟,而视频编码和网络传输

是影响上述性能的主要因素.

4.1 视频编码性能分析

视频编码的参数设置会直接影响本地视频的播放质量,核心参数包括比特率、帧率和分辨率.为测试编码参数对服务质量的影响,我们在硬件条件和网络传输环境相同条件下,分别采用高质量编码和低质量编码进行了测试.测试时采用 Android 手机 MB525 运行应用程序 VideoClient,并通过 Wi-Fi 连接到远程流媒体服务器.

高质量编码情形下的编码参数设置如下:比特率设置成 256kbps,帧率设置成 20fps,分辨率设置成 352×288(视频监控领域的 CIF 分辨率).低质量编码情形下的编码参数设置如下:比特率设置成 196kbps,帧率设置成 15fps,分辨率设置成 176×144(视频监控领域的 QCIF 分辨率).

两种编码情况的效果对比可以看出,高质量编码配置的图像清晰度比低质量编码配置的图像清晰度要高.在测试过程还发现,两者流畅度和延迟时间基本一样,这主要由于两种情形下都通过 WiFi 接入互联网,WiFi 的带宽富裕性保证了视频传输的时效性.

4.2 网络传输的性能分析

系统支持通过 Wi-Fi 和 3G 网络模块上传视频流.在两种不同网络条件下对不同编码参数下的传输延迟进行了对比分析,图 7 给出了性能测试结果.测试结果表明,Wi-Fi 环境下的流畅度优于 3G 环境,特别在 Android 手机抖动比较剧烈时,3G 环境下的画面更容易出现卡顿.在高质量编码情形下,3G 传输延迟约为 7 秒,而 Wi-Fi 约为 5 秒.这表明上传速度对视频直播的流畅度和延迟时间有一定影响.

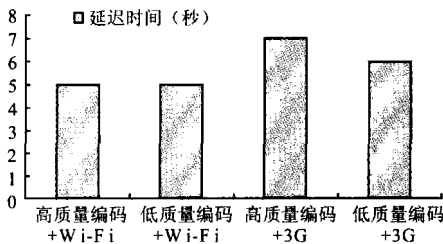


图 7 延迟时间柱状图

5 结束语

本文基于 Android 手机的移动视频直播系统,该系统支持用户通过智能手机内嵌摄像头实时采集现场视频信息,并通过 Wi-Fi 或 3G 等上传视频流

至服务器,提供在线直播或点播服务.在软件实现上,系统将智能手机内置摄像头采集的原始图像编码成 H.264 格式视频流,并通过 HTTP 长连接上传流媒体服务器.流媒体服务器将接收到的视频流保存成视频文件,通过视频发布模块、基于 RTMP 协议向观看者提供视频直播和视频点播服务.论文详细描述了系统实现所涉关键技术,并从视频编码和网络传输两个方面对该系统进行了性能分析.

进一步工作将对视频编码质量和网络传输性能进行优化,通过采用特定的视频编码配置,使本系统能够在网络传输不稳定的时候仍然可以观看清晰、流畅、延迟时间尚可的视频直播,进一步提升用户体验.

#### 参考文献:

- [1] 阳永清,黄立新.基于校园网的便携式视频直播系统的实现[J].现代教育技术,2011,21(12):100-103.
- [2] 朱莹.基于 WEB 的视频直播系统的应用研究[J].电脑知识与技术,2009,5(4):968-970.
- [3] 彭凯,武娟,杨宗凯,等.基于 P2P 的流媒体直播技术研究及展望[J].计算机科学,2009,36(1):10-15.
- [4] 廖宏越,赵蓝兰,周武章.流媒体协议的研究与实现[J].计算机与数字工程,2007,35(4):106-108.
- [5] 李长洲,周曼丽.基于 Web 的流式视频直播技术[J].微型机与应用,2001,20(6):52-55.

- [6] Adobe real-time messaging protocol (RTMP) specification[EB/OL]. [2012-07-15]http://www.adobe.com/content/dam/Adobe/en/devnet/rtmp/pdf/rtmp\_specification\_1.0.pdf, 2009-6.
- [7] [MS-MMSP]: Microsoft Media Server (MMS) Protocol Specification[EB/OL]. [2012-07-15]. http://download.microsoft.com/download/9/5/E/95EF66AF-9026-4BB0-A41D-A4F81802D92C/[MS-MMSP].pdf, 2012-3.
- [8] Adobe Flash Video File Format Specification[EB/OL]. [2012-07-15]. http://download.macromedia.com/f4v/video\_file\_format\_spec\_v10\_1.pdf, 2010-09.
- [9] ISO/IEC 14496-10. Information technology — Coding of audio-visual objects-Part 10: Advanced Video Coding[s]. 2004-10.
- [10] ISO/IEC 14496-15. Information technology-Coding of audio-visual objects-Part 15: Advanced Video Coding (AVC) file format[s]. 2004-04.

#### 作者简介:

杨名阳 男,(1990—),硕士生.研究方向为流媒体服务、无线网络.

张志刚 男,(1970—),硕士.研究方向为电网信息化、智能电网.

叶保留 男,(1976—),博士.副教授.研究方向为移动流媒体计算、无线网络.

(上接第 163 页)

- [4] Wangtong T, Cheung PYK, Luk W. Comparing three heuristic search methods for functional partitioning in hardware-software codesign[J]. Design Automation for Embedded Systems, 2002,6(4):25 - 449.
- [5] Kennedy J, Eberhart R. Particle swarm optimization [C]//Proceedings of IEEE Conference on Neural Networks, Australia;IEEE,1995: 1942-1948.
- [6] Kennedy J, Eberhart R. A discrete binary version of the particle swarm algorithm[C]//Proceeding of IEEE International Conference on Systems, Man, and Cybernetics. Orlando;IEEE,1997: 4104-4108.

- [7] Muhammad Ilyas MENHAS, FEI Min-rui, WANG Ling, et al. Adaptive mutation based probability binary PSO for control of ball mill pulverizing system [J]. Journal of System Simulation, 2011,23(8):1568-1574.

#### 作者简介:

朱向华 男,(1986—),硕士研究生.研究方向为软硬件协同设计.

朱智林 男,(1963—),教授.研究方向为软硬件协同设计.