

• 计算机软件理论、技术与应用 •

基于 H264 的网络视频监控系统设计与实现

林海东 余 强*

(西华大学数学与计算机学院 四川 成都 610039)

摘 要: 介绍一种采用 C/S 架构的 H264 视频流网络监控系统的模块化设计和实现方法。系统服务器端以 ARMS3C6410 为硬件平台,在嵌入式 Linux 操作系统下,利用 V4L2 接口技术对视频进行采集,利用 6410 内部 MFC 模块按照 H264 标准对视频流编码,通过 RTP/RTCP 实时传输协议对视频进行实时传输。客户端通过 FFmpeg 和 SDL 库完成对接到的视频数据的解码与播放。实验结果表明,本系统运行稳定、图像清晰、实时性好,具有广阔的应用前景。

关键词: 视频监控; 视频压缩; 流媒体传输

中图分类号: TP37; TP391.41 文献标志码: A 文章编号: 1673-159X(2014)02-0022-05

doi: 10.3969/j.issn.1673-159X.2014.02.005

Design and Implementation of an Network Video Surveillance System Based on H264

LIN Hai-dong, YU Qiang*

(School of Mathematics and Computer Science Technology, Xihua University, Chengdu 610039 China)

Abstract: This paper introduces a modular design and implementation of the H264 video streaming real-time transmission system based on C/S architecture. ARMS3C6410 is used as the hardware platform for the server of the system, V4L2 interface technology is used to capture video image data, the MFC module of 6410 is used to code video stream according to H264 standard, RTP/RTCP real-time transport protocol is used for real-time video transmission in the embedded environment of Linux. The libraries of FFmpeg and SDL are used to complete decoding and playing of the received video data for client. Experimental results show the system has stable, high-definition and real-time features and has a strong practical and broad application prospect.

Keywords: video surveillance; video compression; streaming media transmission

随着计算机信息技术的发展和人们安全意识的增强,视频监控以其直观、准确、及时和信息内容丰富而广泛应用于许多场合,成为社会安全防范系统的重要组成部分。同时,对视频监控的清晰度、网络化、智能化、功耗节能以及传输设备的体积大小等方面提出了更高的要求。本文根据视频监控的发展现状,设计了一种采用新一代 H.264 视频编码标准的嵌入式 Linux 的远程视频监控系统。该系统采用模块化思想设计,模块之间高内聚低耦合,便于维护和升级。

1 系统总体结构设计

本文提出的视频监控系统整体采用客户机/服务器架构,其设计主要包括 2 方面。1) 服务器设计。服务器端的硬件平台建立在 ARMS3C6410 为主控芯片的开发板上,主要包括视频采集模块、视频压缩模块、视频传输模块。采集模块通过 USB 摄像头采集视频图像。视频压缩模块采用新一代 H.264 视频压缩算法,通过 6410 内部的 MFC (multi format codec) 专用编码芯片把视频图像压缩成 H264

收稿日期: 2013-09-27

基金项目: 四川省教育厅科研项目(10226206)。

* 通信作者: 余 强(1973-),男,副教授,博士,主要研究方向为嵌入式系统、分布式计算。E-mail: yuqiang1973@163.com

NALU 视频流。H264 实时视频的传输协议采用的是 RTP/RTCP 协议模式。RTP 模块主要负责按照 RFC 3984 协议把 NALU 封装成 RTP 数据包发送到客户端,保证流媒体传输的实时性。RTCP 模块主要用于数据分发质量反馈信息,提供流量控制和拥塞控制^[1-5]。2) 客户端设计。其主要包括视频接收模块、视频解压模块和视频显示模块。视频解压模块主要用 FFMPEG 实现,显示模块用 SDL 实现^[6-7]。

2 服务器端视频采集和压缩

2.1 视频采集

视频采集程序是基于 Video4Linux 模块提供的 API 函数。Video4Linux 是 Linux 内核中提供给用户视频采集的 API 接口。它支持绝大多数的 USB 摄像头,对设备的操作简单直观,基本遵循打开视频设备、设置设备属性、设置视频数据格式、设置设备输出方法、循环输出视频信息、关闭设备的顺序。其中,大多数操作都是通过调用 ioctl 函数实现的。图 1 是视频采集的流程图。

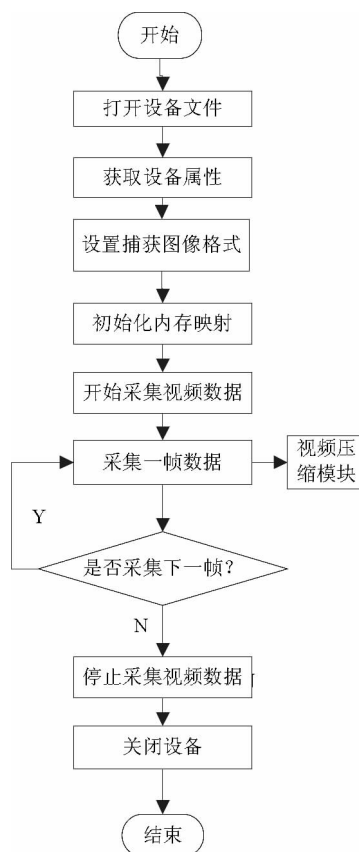


图 1 视频采集流程

1) 打开设备。调用 `open(device_name, flags)` 函数打开设备,打开模式分为阻塞模式和分阻塞模式。

2) 获取设备属性。调用 `ioctl(fd, VIDIOC_QUERYCAP, struct v4l2_capability * argp)` 函数将设备性能参数放到 `v4l2_capability` 结构体中,获取设备是否具有视频捕获和流式 IO 性能。

3) 设置图片格式。调用 `ioctl(fd, VIDIOC_S_FMT, struct v4l2_format * argp)` 设置图像的存储格式、带宽大小、像素等信息。

4) 初始化内存映射。调用 `ioctl(fd, VIDIOC_REQBUFS, struct v4l2_requestbuffers * argp)` 申请缓存,并通过 `mmap` 函数映射到用户地址空间。

5) 开始采集视频。调用 `ioctl(fd, VIDIOC_QBUF, struct v4l2_buffer * argp)` 函数,将申请到的帧缓冲全部放入 V4L2 数据缓冲队列,以便存放采集到的数据,并用 `ioctl(fd, VIDIOC_STREAMON, int * argp)` 使摄像头开始捕获视频。

6) 获取 1 帧数据。调用 `ioctl(fd, VIDIOC_DQBUF, struct v4l2_buffer * argp)` 从视频缓冲区的输出队列中取得一个已经保存有 1 帧视频数据的视频缓冲区,从而获得视频信息。

7) 停止采集视频。调用 `ioctl(fd, VIDIOC_STREAMOFF, int * argp)` 使摄像头停止捕获视频。

8) 关闭设备。调用 `close` 关闭设备。

2.2 视频压缩

视频压缩通过 S3C6410 内部的 MFC 模块完成。MFC 是一个高能的视频编解码器 IP,由嵌入式位处理器和视频编解码器核心模块组成,支持 H.263P3、MPEG-4 SP、H.264 和 VC-1 的编解码。在 Linux 内核中加载 MFC 驱动模块,就可以 `/dev/s3c-mfc` 设备文件对其访问,用 `ioctl` 函数来实现相应的指令操作。MFC 模块的 H264 编码流程如图 2 所示,主要用到的函数如下。

1) `fd = open(MFC_DEV_NAME, O_RDWR | O_NDELAY)`。此函数打开 MFC 设备,如果打开设备成功,则返回该设备的文件描述符。

2) `ioctl(fd, IOCTL_MFC_H264_ENC_INIT, &mfc_args)`。此函数初始化 MFC,设置视频压缩图像的大小、比特率、帧率等。

3) `ioctl(fd, IOCTL_MFC_GET_FRAM_BUF_ADDR, &mfc_args)`。此函数获得 MFC 设备中用于存放原始 YUV420 帧格式的 FRAM_BUF 的存储区的地址,用 `memcpy` 函数把原始视频帧放入以这个首地址开始的存储区中就可以对帧进行压缩。

4) `ioctl(fd, IOCTL_MFC_H264_ENC_EXE, &mfc_args)`。此函数对放入 FRAM_BUF 中的视频帧

进行压缩。MFC 按照 H264 编码标准的基本档次,把原始视频帧压缩并封装到 NALU 中,然后把压缩数据存放 STRM_BUF 的存储区。为减少无用码率,MFC 只对第 1 帧 H264 码流上添加序列参数集 SPS (sequence parameter set) 和图像参数集 PPS (picture parameter set) 这里面包含了如图像的大小、可采用的可选编码模式、宏块到片组的映射等信息,对客户端的解码至关重要。对压缩数据进行发送时,应首先发送第 1 帧编码的 NALU。

5) `ioctl(fd , IOCTL_MFC_GET_LINE_BUF_ADDR , &mfc_args)`。此函数获得存放 MFC 压缩视频帧的 STRM_BUF 的存储区的地址,用 `memcpy` 函数获取压缩后的视频数据。MFC 首次编码存储区中存储着编码图像 NALU、SPS 和 PPS 3 个 NALU,之后的编码过程中缓冲区只包含一个完整的 NALU。

6) `close(fd)`。关闭设备。

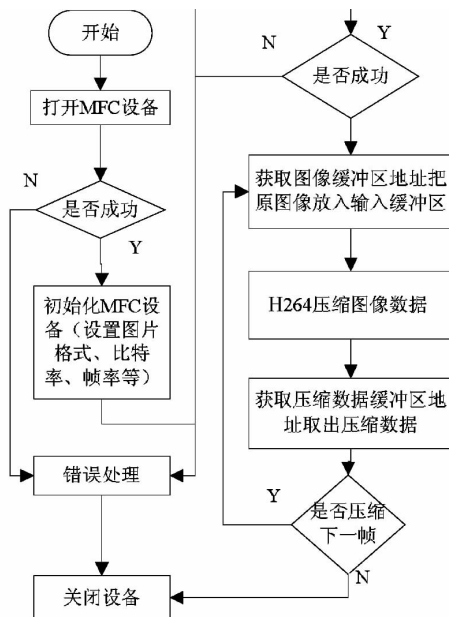


图2 MFC 模块的 H264 编码流程

3 视频传输

为保证视频传输的实时性,在视频的传输中采用实时传输协议 RTP (real-time transport protocol)。RTP 提供了端到端的网络传输功能,适合于通过多播或单播网络视频服务的实时数据传输应用。RTP 的典型应用建立在 UDP 协议上,本身只保证实时数据的传输,并不能为数据传输提供可靠的传送机制,也不提供流量控制或拥塞控制。它依靠 RTCP (RTP control protocol) 提供这些服务。RTCP 负责管理传输质量在应用进程之间交换控制信息。在 RTP 会话期间,各参与者周期性地传送 RTCP 包,包中含

有已发送的数据包的数量、丢失的数据包的数量等统计资料,发送端可以利用这些信息动态地改变传输速率。RTP 和 RTCP 配合使用,能以有效的反馈和最小的开销使传输效率最佳化,故特别适合视频数据的实时性传输。RFC3550 协议详细描述了 RTP 和 RTCP 协议。

本文的视频传输由 ORTP 提供实现。ORTP 是一个支持 RFC3550 协议的开源库,完全由 C 语言实现,除了能保证传输的实时性外,还具有跨平台可移植的特点。在 ORTP 的实现中提供了一个表示 RTP 会话的结构 `RtpSession`。它定义了 RTP 会话中所用到的所有信息,所有操作都是围绕它展开的。

发送视频的实现步骤如下。

1) 在进行 RTP 传输前,首先要对 ORTP 库进行初始化。初始化函数如下:

`ortp_init()`: 初始化 ORTP 库。

`ortp_scheduler_init()`: 初始化任务调度。

2) 对 RTP 收发进行初始化。首先通过 `rtp_session_new` 创建一个 RTP 会话实例,并用一个 `RtpSession` 结构 `session` 表示这个会话;然后通过返回的结构设置任务调度模式、发送模式、目标地址和端口号、负载类型等,如下所示:

`rtp_session_set_scheduling_mode(session , 1)`; 设置任务调度支持阻塞

`rtp_session_set_blocking_mode(session , 1)`; 设置发送接收为阻塞模式

`rtp_session_set_connected_mode(session , TRUE)`; 设置连接模式

`rtp_session_set_remote_addr(session , argv [2] , atoi(argv [3]))`; 设置发送地址

`rtp_session_set_payload_type(session , 0)`; 设置负载类型

3) 发送压缩数据。数据的发送主要用到 `rtp_session_send_with_ts(session , buffer , len , userts)`。`buf` 是要发送 RTP 包的负载部分,在本应用中负载就是 NALU 的数据部分。`len` 是负载的长度。`userts` 是发送 RTP 包的时间戳。S3C6410 的 MFC 在对原始帧进行 H264 编码后,在每一个 NAL 单元前添加起始码 `0X00000001`,可根据这个找到 NALU 的起始头部。如果 NALU 的长度大于 1400,则要按照 RFC3984 对其分割后再发送,以避免 IP 层对大分组的分割。由于在 ORTP 中要自己管理时间戳,因此在对一个 NALU 分割发送的 RTP 包应使用相同的时间戳。

4) 终止 RTP 会话。程序如下:

```
rtp_session_destroy( session );
ortp_exit( );
```

接收视频的实现如下。

初始化和终止的步骤和发送端一样,只是在初始化时要把接收 IP 设置为“0.0.0.0”,端口和发送端口保持一致。数据的接收使用 `rtp_session_recv_with_ts(session, buffer, len, ts, have_more)`。这个函数将根据时间戳 `ts` 接收发送端发送的具有相同时间戳的 RTP 包,并把负载数据放到 `buferr` 里面,如果提供的 `buferr` 太小不足以存下所有的数据, `have_more` 将被设置为 1。把同一时间戳内接收的数据封装在一起就是发送的 NALU 单元。特定的时间戳的数据包接收完后,应根据时间戳增量增加时间戳,然后接收下一个 NALU。

4 客户端开发

客户端的任务主要是把接收到的 H264 视频帧解码成 YUV420 的图像格式,并把图像转化为需要的格式,并在播放器的窗口里播放。本系统采用的基于开源的 FFmpeg 进行 H264 的解码和 SDL 用于解码后视频的播放。

FFmpeg 在 Linux 平台下开发,是一套可以用来记录、转换数字音频、视频,并能将其转化为流的开源计算机程序。它包括了目前领先的音/视频编解码库 libavcodec。用 FFmpeg 解码时首先用 `av_register_all()` 函数注册所有的编解码器,通过 `avcodec_find_decoder(AV_CODEC_ID_H264)` 找到程序所用的 H264 解码器,并返回 H264 解码器的句柄 `pcodec`,然后调用 `avcodec_alloc_context3(pcodec)`,初始化编解码器的一些固定参数,并返回编码上下文句柄 `pcodecctx`。利用上面得到的 `pcodec` 和 `pcodecctx`,通过 `avcodec_open2(pcodecctx, pcodec, NULL)` 打开 H264 解码器就可以进行 H264 解码了。解码时,定义一个 `AVPacket` 结构,把接收到的 NALU 存储的地址和长度传递给 `AVPacket`,利用 `avcodec_decode_video2(pcodecctx, pFrame, &frameFinished, &avpkt)` 函数对 NALU 进行解码,解码得到的 YUV420 格式的图像数据将会存储在 `pFrame` 中。

SDL 是一套开放源代码的跨平台多媒体开发库,使用 C 语言写成。它提供了数种控制图像、声音、输入输出的函数,多用于媒体播放器、开发游戏模拟器等多媒体应用领域。SDL 的播放流程如图 3 所示。

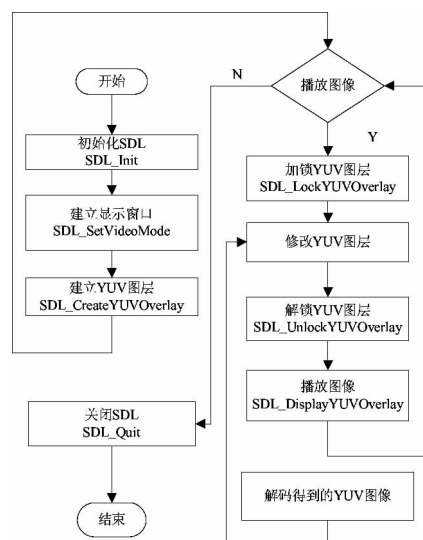


图3 SDL播放

5 实验测试

本文提出了一套基于 H264 的远程视频监控系统方案。系统采用 Linux2.6.28 作为操作系统,以 S3C6410 作为主控处理器实现了对视频数据的采集、压缩和传输。由于采用了新一代的 H.264 视频编码标准,因此,在同等质量图像下,其压缩比是传统压缩标准 MPEG-2 的 2 倍以上,是 MPEG-4 的 1.5~2 倍,提高了系统的数据压缩比。H.264 在具有高压缩比的同时还拥有高质量流畅的图像,因此在网络传输过程中所需的带宽更小,极大地提高了数据的处理与传输效率,使整个系统具有良好的实时性。在客户端将图像帧率设置为 25(f/s),用局域网模拟公网,前端设备与用户终端设备间的信息延迟大约为 2 s。使用 openRTSP 工具观测系统性能,系统运行 2 h,传输速率为 156 kb/s,平均丢包率为 2.3%,满足国家视频监控的标准。在客户端看到的监控视频画面满足视频监控系统的实时性、流畅性和画面清晰的要求。采集图像,如图 4 所示。实验结果表明,本系统成本低、功耗小、功能可扩展、运行稳定可靠,可适用于各个领域的视频监控。



图4 远程视频采集图像

参 考 文 献

[1]李珊珊,王旭国.基于V412的远程视频采集系统设计与实现[EB/OL].(2011-11-24).<http://www.paper.edu.cn/download/downPaper/201103-1124>.

[2]白立朋,李秋林,程磊,等.嵌入式ARM下的USB摄像头监控系统[J].计算机系统应用,2011,20(6):122-125.

[3]刘健敏,杨斌.嵌入式Linux下基于FFmpeg的视频硬件编解码[J].单片机与嵌入式系统应用,2011,11(6):28-31.

[4]毕厚杰.新一代视频压缩编码标准:H.264/AVC[M].北京:

人民邮电出版社,2009:189-213.

[5]S3C6400/6410 HW Multimedia Codec (MFC) User's Guide [M]. [S. I.]: Samsung Electronics, 2008: 2-24.

[6]RFC 1889 - RTP A Transport Protocol for Real-Time Applications [EB/OL]. (1996-01-25). <http://www.packetizer.com/rfc/rfc1889/>.

[7]Doxygen, ORTP APT Documentation Rev0.14.2 [EB/OL]. (2008-02-02). <http://www.antisip.com/doc/ortp>.

(编校:饶莉)

(上接第10页)

3 结 论

1) 在 $\text{Ba}_{0.98}\text{Bi}_{0.02}(\text{Ti}_{0.9}\text{Zr}_{0.1})_{1-x}\text{Co}_x\text{O}_3$ (0.00, 0.005, 0.015, 0.02) 陶瓷样品中, $x=0.00$ 、0.005 时, 陶瓷样品均体现出强烈的弛豫性, $x=0.015$ 处有轻微的弛豫现象, 当掺杂量 $x=0.02$ 时, 陶瓷样品仅体现出弥散相变铁电体特征。

2) 随着 Co^{3+} 掺杂溶度的增加, $\text{Ba}_{0.98}\text{Bi}_{0.02}(\text{Ti}_{0.9}\text{Zr}_{0.1})_{1-x}\text{Co}_x\text{O}_3$ (0.00, 0.005, 0.015, 0.02) 陶瓷内部缺陷偶极子存在形式发生改变, 依次为 $[2\text{Bi}_{\text{Ba}}^{\bullet}-\text{V}_{\text{Ba}}^{\prime\prime}]$ 、 $[\text{Bi}_{\text{Ba}}^{\bullet}-\text{Co}_{\text{Ti/Zr}}^{\prime}]$, $x=0.02$ 时, 陶瓷仅体现出弥散相变铁电体特性。

3) 结合结晶化学特性, 发现随着掺杂量的增加, B 位离子键价和与介电常数呈反比例关系, 介电常数总体增大。

参 考 文 献

[1]Lines M E, Glass A M. Principles and Application of Ferroelectrics and Related Materials [M]. Oxford: Oxford Uni Press, 1977: 241-250.

[2]涂伟, 丁士华, 宋天秀, 等. 包覆 MgO 对 BaTiO_3 基陶瓷性能的影响[J]. 西华大学学报: 自然科学版, 2012, 31(3): 77-80.

[3]Luan W ling, Gao Lian, Guo Jingkun. Size Effect on Dielectric Properties of Fine-grained BaTiO_3 Ceramics [J]. Ceramics International, 1999, 25: 727-729.

[4]Wang Hong, Du Huiling, Yao Xi. Structure Study of $\text{Bi}_2\text{O}_3-\text{ZnO}-\text{Nb}_2\text{O}_5$ based Pyrochlores [J]. Materials Science and Engineering, 2003, B99: 20-24.

[5]Hennings D, Schnell A, Simon G. Diffuse Ferroelectric Phase Transition in $\text{Ba}(\text{Ti}_{1-y}\text{Zr}_y)\text{O}_3$ Ceramics [J]. Am Ceramic Soc, 1982,

65(11): 539-544.

[6]Ke S M, Fan H Q, Huang H T, et al. Lorentz-type relationship of the temperature dependence dielectric permittivity in ferroelectrics with diffuse phase transition [J]. Appl Phys, 2008, 104: 034108(1)-034108(7).

[7]Eichel R A, Erhart P, Traskenlin P, et al. Defect-Dipole Formation in Copper-Doped PbTiO_3 Ferroelectrics [J]. Phys Rev Lett, 2008, 100: 095504(1)-095504(4).

[8]Warren W L, Dimos D, Pike G E, et al. Defect-dipole Alignment and Tetragonal Strain in Ferroelectrics [J]. Appl Phys Rev Lett, 1995, 67: 1689-1691.

[9]Westphal V, Kleemann W, Glinchuk M D. Diffuse Phase Transitions and Random-field-induced Domain States of the "Relaxor" Ferroelectric $\text{PbMg}_{1/3}\text{Nb}_{2/3}\text{O}_3$ [J]. Physics Review Letters, 1992, 68(6): 847-850.

[10]Kleemann W, Klossner A. Glassy and Domain States in Random Dipolar Systems [J]. Ferroelectrics, 1993, 150(1): 35-45.

[11]Kleemann W. Random-field induced antiferromagnetic ferroelectric and structural domain states [J]. International Journal of Modern Physics B, 1993, 7(13): 2469-2507.

[12]Gale J D. GULP: A computer program for the symmetry-adapted simulation of solids [J]. J Chem Soc Faraday Trans, 1997, 93: 629-637.

[13]Lewis G V, Catlow C R A. Defect Studies of Doped and Undoped Barium Titanate Using Computer Simulation Techniques [J]. Phys Chem Solids, 1986, 47: 89-97.

[14]Pirovano C, Islam M S, Vannier R N, et al. Modeling the Crystal Structures of Aurivillius Phases [J]. Solid State Ionics, 2001, 140: 115-123.

[15]Brown I D, Shannon R D. Bond-valence Parameters Obtained from a Systematic Analysis of the Inorganic Crystal Structure Database [J]. Acta Cryst, 1973, A29: 266-282.

(编校: 夏书林)