

# 基于 RTP 的实时视频传输系统

许华荣<sup>1,2</sup>, 李名世<sup>3</sup>

(1. 福建师范大学福清分校, 福建 福清 350300; 2. 厦门大学物理系, 福建 厦门 361005;  
3. 厦门大学计算机科学系, 福建 厦门 361005)

**摘要:** 实时视频流技术在可视电话、远程教育、视频点播等方面得到广泛应用。网络实时视频系统一般分为 4 大模块: 采集子系统、编码压缩子系统、传输控制子系统和输出子系统。其中传输控制子系统是实时视频流的一项关键核心技术。提出了基于 RTP/RTCP 协议构建实时视频传输控制子系统, 传输层通信使用 UDP Socket 完成。对于 RTP/RTCP 协议的封装和 UDP Socket 使用标准 C++ 和 Berkeley 标准 Socket 库完成, 不依赖于具体的平台, 具有良好的可移植性。

**关键词:** 实时视频流; RTP; 传输控制系统

中图分类号: TP393.1 文献标识码: A 文章编号: 1000-7024 (2005) 04-0876-03

## Real-time video transmission subsystem based on RTP

XU Hua-rong<sup>1,2</sup>, LI Ming-shi<sup>3</sup>

(1. Fuqing Branch, Fujian Normal University, Fuqing 350300, China; 2. Department of Physics, Xiamen University, Xiamen 361005, China; 3. Department of Computer Science, Xiamen University, Xiamen 361005, China)

**Abstract:** The real-time video streams technology is used widely in such aspects as videophone, distance education and video on demand. The network real-time video system is generally divided into 4 big modules: capturing subsystem, compressed coding subsystem, transmission and controlling subsystem and output subsystem. Among them transmission and controlling subsystem is the key technology of real-time video streams. How to built transmission and controlling subsystem is proposed based on RTP/RTCP agreement, and UDP socket is utilized to realize the communication between transport layers. Standard C++ and berkeley standard socket storehouse are used to finish the encapsulation of RTP/RTCP agreement and UDP socket. This subsystem does not depend on the concrete platform so it has good portability.

**Key words:** real-time video streams; RTP; transmission controlling system

## 1 引言

流媒体技术是近几年来随着宽带网的兴起而发展起来的一种即时播放的网络媒体传播技术。它的出现改变了下载后播放的传统技术, 客户端可以接收、缓冲、解码、播放同时进行。这种流媒体技术广泛应用于可视电话、远程教育、视频点播等方面。

流媒体的实时传输对带宽和延迟比较敏感, 要求有一定的服务质量保证。它可以容忍一定程度的丢包, 不要求传输的数据 100% 的正确; 但如果数据包的延迟超过一定时间, 它就没有用。所以不能将 TCP 作为传输协议应用到流媒体的传输。因其建立时需 3 次握手的过程, 而导致了较大的时延。流媒体的传输也不能用 UDP 协议。虽然它的传输时延较小, 但 UDP 数据包没有编号, 无法提供差错控制, 也无法加载流的时间信息, 因而 RTP 协议应运而生。RTP 协议提供了为数据包编号、加载时间信息和多点投送等功能, 使 RTP/UDP 在流媒

体领域得到了广泛的应用。

本文提出基于 RTP/RTCP 协议构建实时视频传输控制子系统, 传输层通信使用的 UDP Socket 完成, 经实践检验运行良好。

## 2 网络实时视频流传输系统概述

一般实时视频系统可以分为 4 大模块: 采集子系统、编码压缩子系统、传输控制子系统和输出子系统。整个实时视频系统的组织如图 1 所示。



图 1 实时视频系统图

采集子系统主要指用于视频捕捉的视频捕捉卡。视频捕捉卡的接口, 可以是以 PCI 或 AGP 的方式插入 PC 机箱, 也可

收稿日期: 2004-10-10。 基金项目: 国家 973 重点基础研究发展规划基金项目 (G199803040803)。

作者简介: 许华荣 (1970-), 男, 福建莆田人, 讲师, 博士生, 研究方向为网络通信、IP QoS 和神经网络; 李名世, 男, 副教授, 硕士生导师, 研究方向为计算机网络和多媒体通信。

以直接以USB接口的方式外挂,还有就是通过1394接口与PC机相连的数码摄像机等。

编解码系统有两种形式,一种是纯软件的Codec,由操作系统提供或者用户安装第三方Codec,其编码压缩处理全部依靠用户计算机的中央处理器CPU来处理,因此对于用户计算机的性能要求比较高,特别是在多路视频的情况下,对于一般的用户计算机更是严峻的考验。另一种解决方案就是把编码部分直接固化入嵌入式芯片或者单片机之中,可以独立使用也可以和视频捕捉卡集成,形成功能强大的多功能视频捕捉编解码设备。

输出子系统包含对视频的解码和视频图像的显示。

传输控制子系统是实时视频流的一项关键核心技术。当前网络条件仍然不是非常理想,拥塞和延迟等问题仍然严重,因此在当前的网络条件下,传输实时流媒体信息不能简单地按照顺序发送媒体流数据,而是要充分考虑网络分包发送数据的特点。

### 3 实时传输协议

RTP(实时传输协议 Real-Time Transport Protocol)/RTCP(实时传输控制协议 Real-Time Transport Control Protocol)属于一种应用型的传输层协议,它并不提供任何传输可靠性的保证和流量的拥塞控制机制。

RTP是由IETF(Internet Engineering Task Force)为视音频的实时传输而设计的传输协议。RTP协议位于UDP协议之上,在功能上独立于下面的传输层和网络层,但不能单独作为一个层次存在,通常是利用低层的UDP协议对实时视音频数据进行组播(Multicast)或单播(Unicast),从而实现多点或单点视音频数据的传输。

RTP协议一般由两个部分组成:数据报文部分(RTP报文)和控制报文部分(RTCP)。RTP协议携带具有实时特征的数据,与之作为配套的另一个协议是RTCP协议。RTCP是RTP的控制协议,它用于监视服务质量和正在进行的与会者会话上传递信息,它单独运行在底层协议上。RTCP是由接收方向发送的报文,它负责监视网络的服务质量、通信带宽以及网上传送的信息,并将这些信息发送给发送端。RTCP报文共有5类:SR(发送报告)、RR(接收报告)、SDES(源描述项)、BYE(结束标志)、APP(应用特定函数)。

RTCP的基本做法是周期性地向会话的所有参加者进行通信,采用和数据包分配传送的相同机制来发送控制包。RTCP的主要功能如下:

- (1) 数据传输的质量提供反馈,并提供QoS的检测;
- (2) 提供不同媒体间的同步;
- (3) 在会话的用户界面上显示会话参与者的标识。

### 4 利用RTP/RTCP协议完成实时控制

传输控制子系统是基于RTP/RTCP协议构建,通过传输层通信使用的UDP Socket完成。

对RTP/RTCP协议的封装和UDP Socket使用标准C++和Berkeley标准Socket库完成,不依赖于具体的平台,具有良好的可移植性。

```
int nStatus = m_gSession.Create(nPortBase);
m_gSession.SetMaxPacketSize(SESSION_PACKET_SIZE);
m_gSession.SetReceiveMode(RECEIVEMODE_ALL);
m_gSession.ClearDestinations();
```

程序运行开始需要初始化RTP Session,创建底层Socket,清空传输目标列表。

RTP需要根据时间戳来计算是否反馈RTCP报文。这里以1ms作为时间戳单位。

```
m_gSession.SetTimestampUnit(1.0/1000.0);
```

RTP Session对象维护了一个传输目标列表,支持数据组播功能,如果要往某个地址发送数据,则把改地址可以添加到传输目标列表之中。发送的时候会自动往所有传输目标列表内的地址依次发送数据报。

```
unsigned long addr = ntohl(inet_addr("127.0.0.1"));
m_gSession.AddDestination(addr,5000);
```

使用RTP发送数据不能简单地把数据直接发送出去,而是需要打包为一个RTP报文。RTP报文携带了控制视频数据实时控制的信息。RTP报文由报文头和数据部分组成。在RTP头格式中,固定头报文头开始的12个字节出现在每个RTP包中,而CSRC标识列表仅出现在混合器插入时。

一个RTP报文中携带的实际数据被称为有效载荷(Payload)。载荷本身是没有任何实施控制信息的。由于数据报传输过程中有可能丢失或出现差错,我们需要根据RTP报文的报头信息来处理这些意外情况。

报文之中最重要的是几个数据字段:有效载荷类型PT、标志位Mark、16位序列号以及时间戳timestamp。有效载荷类型代表了RTP报文数据的类型,一般的视频会议系统都会涉及到视频、音频、文字等不同的实时数据,这就需要有一个类型标识进行区分,从而接收方可以选择正确的解码器进行处理。在试验平台之中,我们定义了如下载荷类型:

```
#define RTP_PAYLOAD_NONE 0
#define RTP_PAYLOAD_VIDEO 1 //视频数据载荷
#define RTP_PAYLOAD_VIDEOKEY 2 //视频关键帧载荷
#define RTP_PAYLOAD_AUDIO 3 //音频载荷
#define RTP_PAYLOAD_MSG 4 //文字信息载荷
#define RTP_PAYLOAD_CONTROL 0x20 //控制信息载荷
```

时间戳代表数据报产生的惟一时间标志,时间戳相同的数据报不论到达的先后次序,都被认为是同一个时刻产生的数据。

例如,在视音频传输服务中,RTP源可能会有几种媒体(如视频和音频)需要传输,这些不同的媒体之间的同步需要依靠RTP时间戳信息来进行同步。序列号代表了RTP数据报携带数据的发送次序。

我们知道,网络传输的不可靠性的表现之一就是数据报“乱序到达”,先发送数据报有可能比后发送的数据报更迟到达,而这对于一个连续序列的实时视频数据是非常致命的。因此,我们就必须依靠这个序列号来调整到达数据报的顺序,使之和发送数据一致才可以正确获取我们需要的数据。

此外,由于一个视频采样压缩之后仍然有可能相当巨大,可能超出预先约定规定的 RTP 数据报大小,这个时候就需要拆分数据报了。

但是由于这些数据报都属于同一个整个视频采样,因此这些数据报的时间戳应该是一致的。此外,Mark 标志用来作为区分这些拆分数据报和普通数据报的分界线。发送视频帧的流程如图 3 所示。

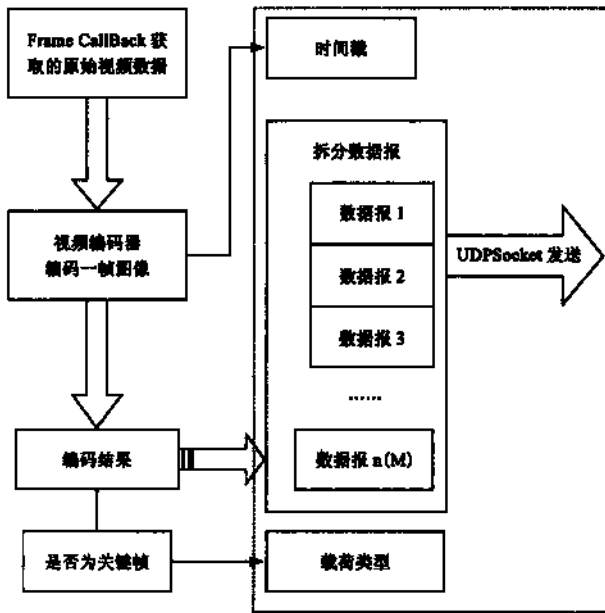


图 3 发送视频帧的流程图

在数据接收一方,当接收到一个完整 RTP 报文之后,就根据 RTP 报头有效载荷把该数据报派发给不同的载荷类型处理函数,直到完成接受任务为止。同时所有的接收方把它最近的接收情况报告给所有发送者,这些信息包括所接收到数据包的最大顺序号、丢失的包数、乱序包的数量以及用于估计传输时延的时间戳的信息。而这些信息反映了当前的网络状况,发送方在接收到这些信息后自动地调整它们的发送速率。接受一方流程如图 4 所示。

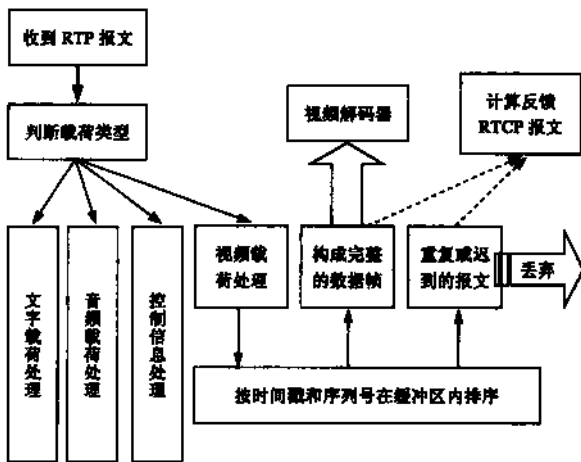


图 4 接收视频帧流程图

其中,在缓冲区按照时间戳和序列号重组数据报是最关键的环节。只有当受到 Mark 标志的多个数据报都到达了才能构成一个完整的数据帧提交解码器。因此,某些数据报虽然迟到,但仍然有可能构成一个完整的视频帧。当然,数据报也可能彻底丢失,这就需要缓冲区定期清理一些许久都无法拼凑完整的视频帧,在清理掉之后,那些重复和迟到的数据报将被简单地丢弃。

## 5 结论

经测试,在常见的网络状况和带宽下进行点对点实时视频图像传输,画面质量和实时性达到比较理想的效果。根据程序测试在运行中实时计算视频采样速率、编码后带宽、关键帧间隔 IFD,以及接收速率、丢帧率等运行状态,可以看到如下两个特点:

(1) 采用的 RTP/RTCP 协议,可以方便地实现多播(Multi-Case)处理和 RTCP 自动反馈,有利于进一步扩展程序成为视频会议系统;

(2) 对于平均每个 RTP 数据包应该携带多少个视频帧的问题,与 IFD 和最后的数据流带宽都有关系。由于 RTP 报头大小固定,为了提高利用率每个 RTP 包内应携带尽可能多的视频帧,但是由于整个 RTP 包大小受限于 UDP 包大小,太大的 RTP 包会导致传输过程中被路由器拆分,反而导致丢包率大大提高,而且对于加密模块而言,太大的 RTP 包也会造成延时的增加。

经测试,一个 RTP 包携带 2 个视频帧可以到达一个比较理想的平衡点,测试结果如图 5 所示。

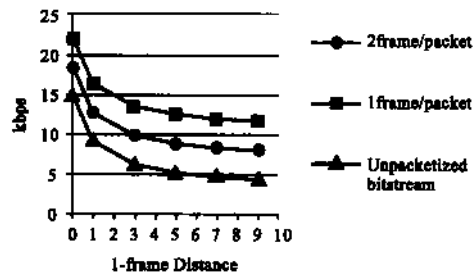


图 5 数据流带宽随关键帧的变化

## 参考文献:

- [1] 张明德, 王永东. 视频会议系统原理与应用[M]. 北京: 希望电子出版社, 1999.
- [2] 翟海涌. 数字视频网络传输层协议的选择[Z]. 2002, (2), 61-64.
- [3] Wei Dai. Crypto++ Library 5.1 [EB/OL]. <http://www.eskimo.com/~weidai/cryptlib.html>.
- [4] VIC-Video conferencing tool [Z/OL]. University of California, Berkeley <http://www-nrg.ee.lbl.gov/vic/>.
- [5] 许先斌, 朱平, 安晖. 运用 RTP 协议实时传输 MPEG4 流[J]. 计算机工程与设计, 2003, 24(2):57-59.
- [6] 张艳珍, 巩玉国, 欧宗瑛, 等. 基于 Windows 平台的视频捕捉技术研究[J]. 计算机工程与设计, 2002, 23(3):10-12.