文章编号： 1000-5013(2011)04-0389-04

音视频实时同步传输技术研究与实现

陈志颖，谢维波，王磊

（华侨大学 计算机科学与技术学院，福建 泉州 362021）

摘要： 提出一种基于实时传输协议的音视频同步的编码方法，在同步播放的音频和视频数据之间建立起一种对应关系，在数据不同步时能迅速地找到下一个同步点继续播放．在 TCP/IP 网络上，基于 H.264 视频编解码、G.729A 音频编码标准和实时传输协议 RTP/RTCP，设计并实现一个音视频实时传输系统．实验结果表明，音视频同步的解决方案能够使音频数据和视频数据时间偏移保持在±80 ms 之间，同步前与同步后的同步相位失真有明显的改善．

关键词： 实时传输；同步；音频；视频；H.264；RTP/RTCP

中图分类号： TN 919.8 **文献标志码：** A

如何在 TCP/IP 网络上公平地提供流媒体服务与传统数据业务，是网络传输控制协议需要考虑的核心问题．另外，TCP/IP 网络传输中的延迟、抖动、网络拥塞等，使得流媒体发送端的发送速度与接收端的接收速度不匹配等，对实时传输的媒体服务质量产生重要的影响，致使接收端的媒体演播中存在媒体内抖动（不连续性）和媒体间非良好匹配（异步）[1]．研究多媒体实时传输和同步是保证多媒体服务质量的重要的步骤，是多媒体研究的关键技术之一．解决音视频同步问题一般有 2 种方法：一是对相关媒体进行同质处理，二是对各相关媒体分开进行处理．总体上说，多媒体同步方案的设计应满足多种通信模式，应选择灵活的同步方法，同时其算法应简单，额外开销小．因此，需要找到一种方法，能直接体现音频和视频之间的同步关系，而且当两者的播放速度不一致时，可以迅速找到新的同步播放点．文中提出了一种基于实时传送协议（RTP）的音视频同步的编码方法．

1 音视频同步解决方案

采用对各相关媒体分开进行处理的方法，即对音频流和视频流分别进行处理．通常方法是通过记录初始的播放时间和 RTP 时间戳，用接收端来协调音频和视频数据的播放．当接收端接收到一帧数据时，计算该帧的时间戳和初始的 RTP 时间戳之间的差值，以及当前时间和初始放播时间之间的差值，据此判定该帧是否播放及是否同步．

然而，随着接收端不断地接收到新数据，需要不停地计算和比较每一帧音频和视频数据的时间差值．这种方式中的音频和视频之间没有直接的同步关系，而只是参考各自的时间戳和同一个本地时间单独控制播放自己的数据．另外，当音频或视频数据的播放速率滞后时，需要连续向后处理多个 RTP 包以寻找新的同步点，即找到一帧能与本地时钟对应的数据．

因此，需要找到一种方法，能直接体现音频和视频之间的同步关系，而且当两者的播放速度不一致时，可以迅速找到新的同步播放点．音视频采集在理论上是同时开始的，但由于程序的顺序执行，视频与音频的采集起始时刻一定不同，导致编码的开始时刻也不相同．首先，启动视频采集线程，再马上启动音频采集线程，记录下视频采集时间与音频采集时间的时间差；其次，根据视频采集速率和音频采集速

收稿日期： 2009-11-23

通信作者： 谢维波（1964-），男，教授，主要从事嵌入式技术和数字信号处理的研究．E-mail：xwblxf@hqu.edu.cn.

基金项目： 福建省自然科学基金资助项目（2010J01334）；福建省厦门市科技计划项目（3502Z20083047）；福建省厦门市重点产学研基金资助项目（厦经技[2009]233-03）

率计算出应丢弃的视频帧数,然后启动视频和音频编码线程.视频编码时应先丢弃视频帧数后再开始编码.系统的视频码率为29.97帧·s^{-1},音频帧大小为10 ms.封装好的视频RTP包和音频RTP包分别放于发送端的视频发送缓存和音频发送缓存中,同时发送序列号相同的视频包和音频包.

采用时间戳的方法在传输数据时不用改变数据流,不需要附加同步信道.其缺点是选择相对时标和确定时间戳操作较为复杂,需要一定的开销用于同步操作.另外,当视频或音频数据包的传输速率滞后时,需要连续向后处理多个RTP包以寻找新的同步点.

文中提出的方法能直接体现视频和音频之间的同步关系,而且当两者的传输速度不一致时,可以迅速找到新的同步播放点.该方案在音频帧的RTP数据包头部引入了一个对视频包的索引.这个索引表明了音频帧同时播放的视频帧的第1个RTP包序列号,可以通过RTP包头的扩展结构加以实现.

音频帧与视频帧之间的索引关系,如图1所示.图1中:音频数据包中上面的序号是其自身的RTP序列号,而下面的序号是相应视频数据包的RTP序列号.选择在音频包中加入索引号的原因是:一方面是考虑到人们对声音的变化更敏感,另一方面是视频的数据量比音频的数据量大,而且通常都是由视频的滞后引起不同步.

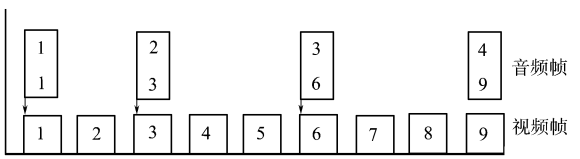

图1 音频帧与视频帧之间的索引关系

Fig.1 Index between audio frame and video

采用这种同步方案需要考虑如下3个主要问题:(1)音频帧的大小与其采样频率有关,一个音频帧可能包含在多个RTP包中,也可能一个RTP包中有多个音频帧.前一种情况下,多个RTP包中的索引号是一样的;后者,音频帧RTP的索引号取值为第1个音频帧对应的索引号;(2)因为音频帧RTP包里面有指向视频帧的索引号,这时显得音频帧RTP包对同步尤为重要,而RTP协议可以提供网络的QoS,保证了音频帧RTP包索引号的完整;(3)如果在同步过程中对应的视频数据包没有到达,音频则继续正常播放.

采用这种同步方案具有如下3个优点:(1)信宿端视频和音频不同步时,不需要连续处理多个RTP数据包,直到找到新的同步点,这不但减轻了信宿端的处理负担,也加快了同步的速度;(2)有利于信源端调整数据的发送速率,适应网络带宽的变化.当发生网络拥塞时,发送方需要降低发送的速率,此时可以采用帧丢弃策略,即只发送音频数据索引的关键帧,可以更快地适应网络带宽的抖动;(3)索引号是直接通过RTP包头的扩展结构加以实现的,方法操作简单、开销小、实时,不需要同步时钟且兼容性好.

2 H.264实时同步传输系统

2.1 总体设计

系统采用G.729A音频编解码、H.264视频编解码和RTP实时传输协议[2-3],分为音视频采集、编解码、传输控制和回放4个模块.图2为系统的结构层次框图.

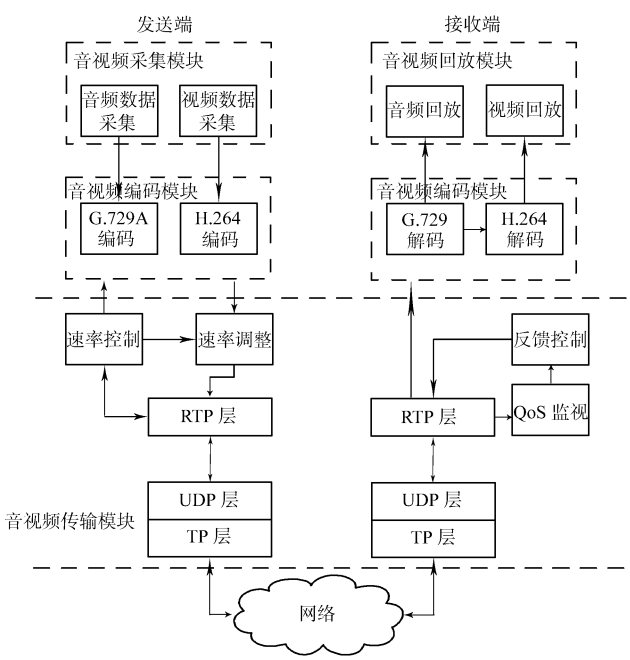

图2 音视频实时传输系统框图

Fig.2 Real-time transmission system of audio and video

在发送端,对音视频进行采样,获得数据流经过音视频编码,再经过RTP协议的打包发送出去;然后,RTP协议会根据网络反馈信息,估计网络的可用传输带宽,自适应地调整编码器的编码输出速率(包括信源码率的调整与信道码率的调整),使得音视频码流能够满足当前网络传输可用带宽的限制.在接收端,对接收的音视频流进行解码,重构音视频

信号，计算当前网络传输参数（如传输中的丢包率等）并发送反馈控制信息.

2.2　音视频码流 RTP 封装

图 3 为 RTP 的包头格式. 图 3 中：RTP 的包头格式中前 12 字节出现在每个 RTP 包中，其各域：V（版本位，2 bit），定义了 RTP 的版本；P（填充位，1 bit）；X（扩展位，1 bit）；CC（CSRC 计数，4 bit）；M（标志位，1 bit）；PT（负载类型，7 bit）；Sequence Number（序列号，16 bit）；Timestamp（时间戳，32 bit）；SSRC（用以识别同步源，32 bit）. CSRC 列表：0 到 15 项，每项 32 bit，CSRC 列表识别在此包中负载的所有贡献源，仅仅在被混合器插入时才出现 CSRC 识别符列表.

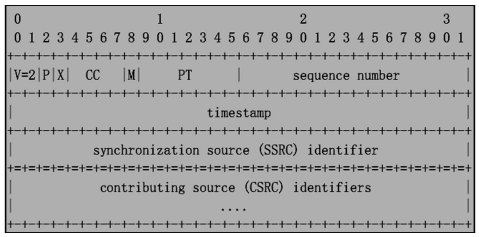

图 3　RTP 包头格式

Fig. 3　Format of the RTP packet header

若设置扩展比特位 X，固定头（包头格式中的前 12 字节）后面跟随一个头扩展，这样 RTP 提供了扩展机制以允许实现个性化：某些新的与负载格式独立的功能要求的附加信息在 RTP 数据包头中传输[4]. 提出的同步方案运用了音频 RTP 包的扩展比特位 X，音频帧 RTP 包头里的头扩展（Header Extension）位于图 3 的 CSRC 域（本文方案没有使用 CSRC 域），记录着音频帧对视频帧的索引号. 这个索引号是相应同步视频帧的序列号（Sequence Number）.

采用平均分割的思想，代码实现如下：

```
int NALUSplit(int nalu_length)
{   int n;
    if((nalu_length%MTU_SIZE)>0)
      n=(nalu_length/MTU_SIZE)+1;
    else
      n=nalu_length/MTU_SIZE;
    if((nalu_length%n)>0)
      nalu_size=(nalu_length/n)+1;
    else
      nalu_size=nalu_length/n;
    return nalu_size;
}
```

2.3　RTP 编程实现

设计一个函数，专门用来处理音频帧与视频帧索引号对应关系 SetIndexNum(AudioData * aData, VideoData * vData). 首先，调用 Create()函数创建一个 RTP 会话，并指明要用的端口号. 设置恰当的时间戳单元，通过调用 RTPSession 类的 SetTimestampUnit()方法来实现. 其次，对采集来的音频数据和视频数据进行封装，分别调用 VideoForRTP()和 AudioForRTP(). 其中，AudioForRTP()中调用了 SetIndexNum()函数.

当 RTP 会话成功建立起来后，就可以开始进行流媒体数据的实时传输. 这需要设置好数据发送的目标地址，通过调用 RTPSession 类的 AddDestination()，DeleteDestination()和 ClearDestinations()方法来完成. 目标地址指定后，就可以调用 RTPSession 类的 SendPacket()方法，向指定的目标地址发送流媒体数据. RTP 实现流程如图 4 所示.

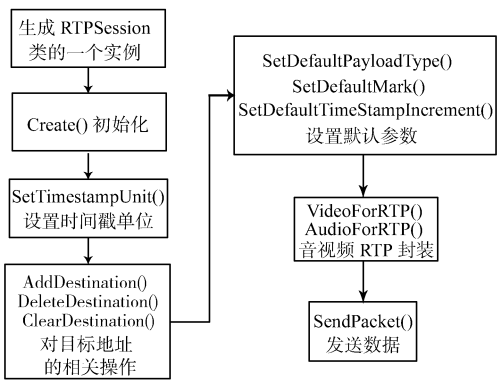

图 4　RTP 的实现流程

Fig. 4　Process of RTP implementation

3　实验结果分析

采用同步相位失真（Synchronization Phase Distortion）来客观衡量多媒体同步系统同步性能，即媒体失步程度可用同步相位失真度 D_{SP} 来表示[5-6]. 同步相位失真定义为两个强相关对象，也即两个时间上最邻近的对象与其原始时间间隔，以及发生的时间间隔变化. 即

$$D_{av} = [P_a(m) - P_v(n)] - [G_a(m) - G_v(n)].$$

其中：$G_v(n)$，$P_v(n)$是视频流中第 n 个媒体单元的产生时间和播放时间；$G_a(m)$，$P_a(m)$是音频流中第 m 个媒体单元的产生时间和播放时间[7].

实验中,音频是基于G.729A标准的,其编码速率为8 kbit·s^{-1},每10 ms语音为一帧音频;视频采用H.264编解码,其编码图像为QCIF格式(176 px·144 px,YUV为4∶1∶1),正常播放帧率约为25帧·s^{-1}.记录了音、视频数据帧数、采集时间、播放时间,所得的同步前与同步后的SPD数据,如表1所示.

表1 同步前后的同步相位失真度比较

Tab.1 Comparison of the synchronization phase distortion before and after synchronized

帧数	D_{SP}/ms	
	同步前	同步后
1	0	0
20	−127.35	−12.88
40	−108.97	−23.56
60	−110.29	−13.43
80	−147.13	−51.02
100	−139.22	−39.23
120	−104.77	−28.31
140	−109.62	−24.68

从表1的结果可知,同步前、后的同步相位失真有明显的同步改善,并处在同步区域内.此外,主观衡量结果表示,观众不会明显感觉到音频和视频在时间上的偏移.说明音频和视频时延偏移在−80～+80 ms之间,处于同步区域.

4 结束语

虽然只在音频和视频媒体数据上进行同步,未对包含文本,图片,图形等媒体数据构成的复合信息实体进行同步,但是所设计方案依然可以对其他类型的媒体数据建立对应实现同步.未来的工作进一步优化音视频同步方案,并在此基础上移植到嵌入式平台上面.

参考文献:

[1] 陈霞, 蔡灿辉. 网上视频传输拥塞控制策略的实现[J].华侨大学学报:自然科学版,2008,29(2):208-212.

[2] 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,2008.

[3] 许华荣,李名世.基于RTP的实时视频传输系统[J].计算机工程与设计,2005,26(4):876-878.

[4] 王少燕.多媒体通信中的音视频同步问题研究与实现[D].西安:西安电子科技大学,2003.

[5] SCHULZRINNE H,CASNER S,FREDERICK R,et al. RTP: A transport protocol for real-time applications[EB/OL].[2003-07-01]. http://www.cnpaf.net/Class/Rfcen/200502/4612_4.html

[6] 孙文彦.实时传输中的多媒体同步技术研究[D].北京:北京航空航天大学,2001.

[7] XIE Yong,LIU Chang-dong,lEE M J,et al. Adaptive multimedia synchronization in a teleconference system[J]. Multimedia Systems,1999,7(4):326-337.

Research and Implementation on the Synchronization of Real-Time Audio and Video Transmission

CHEN Zhi-ying, XIE Wei-bo, WANG Lei

(College of Computer Science and Technology, Huaqiao University, Quanzhou 362021, China)

Abstract: A coding method of synchronization between audio and video based on the real-time transport protocol is proposed, with establishing the correspondence relationship between audio and video data broadcasting synchronously and quickly finding the next synchronization point to continue playing when the data is not synchronized. In the TCP/IP network, a real-time audio and video transmission system is designed and implemented based on the H.264 video coding standard, G.729A audio and the real-time transport protocol RTP/RTCP. Experimental results show that this program will enable the synchronization of audio and video data maintained time shift between the ±80 ms, the synchronization phase distortion synchronized before and after has significantly improved.

Keywords: real-time transmission; synchronization; audio; video; H.264; RTP/RTCP

(**责任编辑**: 黄晓楠 **英文审校**: 吴逢铁)