

多源流媒体课件同步编辑系统的研究与实现^{*}

朱郑州 吴中福 王 茜

(重庆大学计算机学院 重庆 400044)

摘要 为了提高多源流媒体课件的质量,本文研究了 Advanced Streaming Format 和 Real Media 等流媒体的同步切割技术和多流合并技术,提出了流媒体编辑的关键帧微调技术,设计了两种同步编辑多源流媒体课件的方法:基于公共参考时间轴的同步编辑方法和基于关键帧的同步编辑方法。开发了一套多源流媒体课件的同步编辑系统,实现了多源流媒体的同步切割和合并。

关键词 多源流媒体课件,同步编辑,多流合并,关键帧微调技术

Research and Realization of the System of Synchronization Editing Multi-Source Streaming Media Courseware

ZHU Zheng-Zhou WU Zhong-Fu WANG Qian

(College of Computer Science and Technology, Chongqing University, Chongqing 400044)

Abstract To improve the quality of multi-source streaming media courseware, this paper studies the technology of synchronization cutting and merging of advanced streaming format and real media. The paper proposes a technology of key frame fine tuning, and designs two synchronization editing methods of multi-source streaming media courseware; the one is based on common timeline, and the other is based on key frame. The paper also presents a system of multi-source streaming media courseware synchronization editing.

Keywords Multi-source streaming media courseware, Synchronization editing, Multi-stream merging, Key frame fine tuning

1 引言

课件可以再现教师授课的效果,是现代远程教育的重要组成部分。多源流媒体课件因其具有文件小、可边下载边播放两大优点而得到课件制作者和使用者的青睐,成为现代远程教育课件格式的首选^[1]。一般情况下,教师的教学形式多样,有 PowerPoint、应用软件、网页等。多源流媒体课件通过对教师教学活动的实时录制,获得教师的教学媒体流(包括屏幕流、教学形象视频流、声音流和学生学习场景流等),录制生成的流媒体一般采用 ASF 或 RM 格式(这里我们假设教师视频流为 ASF 格式,教学媒体流为 RM 格式),从而把实施授课现场场景的多源流媒体提供给学习者的一种基于 Internet/Intranet 授课方式的网络多媒体课件。然而由于课件录制过程中可能会遇到各种突发事件或者不经意事件(如鼠标突然失灵、麦克风突然倒下,或教师打喷嚏、挖耳朵等),导致录制生成的课件质量大大降低。为了提高课件的质量,需要对原始的多源流媒体课件进行编辑处理。

2 多源流媒体课件的编辑方案选择

目前多源流媒体课件的编辑处理方案主要有两类:脚本优化控制方案和物理编辑方案。

方案一:脚本优化控制方案。这种方案不改变流媒体课件资源自身的任何内容,而是在外部采用脚本控制技术,在课件的点播时,对流媒体课件的播放进行有选择的控制。常用的脚本语言有:SMIL、ASX、XML、TIME、JavaScript 等^[2,3]。

此方案的优点是操作简单,能保持视频元数据的原始特性,同时达到优化控制流媒体的目的。但是缺点也很明显:(1)元数据资源的质量差;(2)课件可重用性差;(3)课件后期编辑复杂;(4)对课件制作人员要求较高。

方案二:物理编辑方案。这种方案是分析编辑流媒体课件,从物理上直接切除有瑕疵的教师视频流或有瑕疵的屏幕流。但是这种方案有一个致命的缺点,那就是多源流媒体课件中多个媒体流的同步性被破坏,导致课件目录、教师教学演示媒体流(即屏幕流)与教师的音视频不同步,大大降低了课件的质量。

基于此,作者提出了第三种解决方案,即多源流媒体课件的同步编辑方案。该方案也是属于物理编辑方案,它结合了方案一的优点而对方案二进行了改进,在切除瑕疵教师视频流的同时也切除相应的教学媒体流(尽管教学媒体流是好的),或者在切除瑕疵教学媒体流的同时也同步切除相应的教师视频流(尽管教师视频流是好的),保证了编辑后的多源流媒体课件与原始的多源流媒体课件的同构性和一致性。对多源流媒体课件的同步编辑处理所用到的关键技术有:流媒体切割技术、多个相同格式媒体流的合并技术、流媒体关键帧微调技术和多个不同格式媒体流的同步编辑技术。

3 关键技术研究

3.1 流媒体切割技术

流媒体的切割技术如图 1 所示,主要包括以下三个过程:
(1)文件头对象的分析和文件属性的确定——读取器。

^{*}基金项目:国家发改委科学研究计划项目(CNGI-04-15-3A)。朱郑州 讲师,博士研究生,主要研究方向:现代远程教育,网络计算;吴中福 教授,博士生导师,主要研究方向:现代远程教育,计算机网络安全,网络计算;王 茜 副教授,硕士生导师,主要研究方向:现代远程教育,计算机网络安全,电子商务。

一般流媒体文件逻辑上包含三种顶级对象:头对象(Header Object)、数据对象(Data Object)和索引对象(Index Object)。头对象用于描述文件的全局信息,包括文件大小、流的数量、编解码信息等;数据对象以固定大小的数据包存储流数据;索引对象主要保存时间索引和数据位置的映射信息,便于应用程序的随机访问。其中头对象和数据对象是每个流媒体文件所必须的,而索引对象则是可选的。

(2)基于时间轴的媒体流数据信息的选择与复制——生成器。依据系统设置或者用户自定义切割片断的起始时间和结束时间选定待切媒体流数据信息,根据时间计算出数据信息的大小,然后根据数据块大小计算出预计的复制时间。具体操作步骤为:

- ①获取源流媒体文件中的 Profile 信息;
- ②设置源流媒体文件中的属性信息、编码信息、脚本信息、标志信息;
- ③根据时间轴顺序,把读取器输入的流媒体数据传送给生成器。生成器根据 Profile 的描述交替打包,并写入流媒体文件的数据对象;



图1 流媒体切割技术

④给目标流媒体文件自动添加索引,以提供文件的随机定位能力。

(3)包装新生成的媒体流——形成目标流媒体课件。根据原始流媒体文件的 Profile 属性设置切割出来的新的目标流媒体文件的 Profile 属性,同时为目标流媒体文件添加索引等信息。

3.2 流媒体合并技术

在逻辑上,流媒体的合并是流媒体切割的逆操作,然而在实际应用中,流媒体的合并也涉及到更多的相关技术。对于N个相同格式相同大小(假设为2MByte)的流媒体文件的合并,其解决方案有三种策略:

策略1:串行合并策略。如图2-a所示,在合并处理的过程中,先执行流1和流2的合并,然后把合并的结果流5与流3合并……以此类推。显然,合并过程中只需要启动1个线程就足够了。需要复制的数据大小为:

$$S1 = (1+2+3+\dots+N-1) \cdot 2\text{MByte} = N(N-1) \text{ Mbyte}$$

这种合并策略的合并次数为N-1,优点是算法简单易操作,节省空间。缺点是合并的效率低,合并耗时长。

策略2:两两并行合并策略。在合并处理的过程中,先把N个流媒体文件分为 $\lceil N/2 \rceil$ 组,每组2个文件。先执行流1和流2、流3和流4、...流N-1和流N的合并操作,并把前面合并的结果再次进行两两分组,然后继续执行合并操作……合并操作的流程如图2-b所示,是一个二叉树,该二叉树总共有 $\log_2 2N$ 层,假设 $N=2k, k \in \mathbb{Z}$,则该二叉树是一个完全二叉树。如果每一层的合并操作并行处理,则至少需要启动的线程数是 $\lceil N/2 \rceil$,而需要合并的次数依然是N-1。需要复制的数据量大小为:

$$S2 = N \cdot \log_2 2N \cdot 2 \text{ MByte} = 2N \cdot \log_2 2N \text{ MByte}$$

该策略的优点是算法灵活,便于并行处理;缺点是系统控制复杂,并行处理时需要耗费较多的空间。

策略3:全并行策略。如图2-c所示,该策略是同时读取多个流媒体文件的数据,同步合并到新的流媒体文件中。

$$S3 = N \cdot 2\text{MByte} = 2N \text{ MByte}$$

优点是算法结构简单,数据流的重复拷贝次数少,处理速度快,缺点是算法实现困难,处理过程中需要消耗的内存资源多。

三种合并策略的对比如表1所示。需要指出的是,对于后两种策略,其合并效率不受待合并流媒体大小的影响,而对于第一种合并策略,因为有部分流文件需要重复拷贝,且重复拷贝的次数不等,所以,在各个代合并流媒体大小不等的情况下,为了减少数据的总拷贝量,缩短编辑时间,采用把多个待合并文件按照从小到大的顺序排列的方式,尽量使得文件较小的流媒体首先合并,即重复拷贝次数较多的流媒体较小。

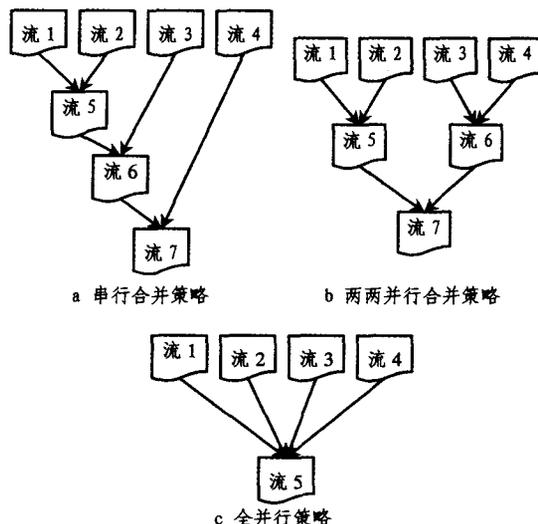


图2 流媒体合并策略

表1 三种多流合并策略的对比

	串行合并	两两并行合并	全并行
复制数据量 (Mbyte)	$N(N-1)$	$2N \cdot \log_2 N$	$2N$
占用空间	小	中	大
并行性	不支持	支持且易操作	支持但不易操作
合并次数	$N-1$	$N-1$	1
算法实现	简单	较简单	复杂

3.3 关键帧微调技术

流媒体切割的核心问题是切割生成的流媒体在播放的起始阶段容易出现黑屏、屏幕画面静止或者没有声音等现象,这不但影响切割的效果,同时也会给使用本课件学习的学员带来挫折感。为了解决这一有重要影响的问题,我们根据视频流媒体文件的编解码原理,提出了关键帧微调技术。

视频编码基于运动补偿(Motion Compensation, MC)技术^[4]。在运动补偿中,通过预测与最新编码的视频帧处于同一区域的视频帧中各宏模块的像素来实现压缩。例如,背景区域通常在各帧之间保持不变,因此不需要在每个帧中重新传输。运动估计(Motion Estimation, ME)是确定当前帧——即与它最相似的参考帧的过程。ME通常是视频压缩中最消耗性能的功能。有关当前帧中各模块最相似区域相对位置的信息(“运动矢量”)被发送至解码器。在视频编解码器中,单个帧可以采用三种模式中的一个进行编码——即I、P或B帧模式(见图3)。几个称为Intra(I)的帧单独编码,无需参考任何其他帧(无运动补偿)。某些帧可以利用MC编码,以前一个帧为参考(前向预测)。这些帧称为预测帧(P)。B帧或双向预测帧通过当前帧及后续帧进行预测^[5]。其中的I帧和P

帧是关键帧,如果切割点选择在关键帧的位置,那么就可以避免黑屏和静止画面现象的出现。但是用户选择的切割点并不能总是选在关键帧的位置,而且用户也不知道关键帧的位置在哪里,这就需要计算机采用一定的算法对用户选择的切割点进行微调,以调节到关键帧的位置。

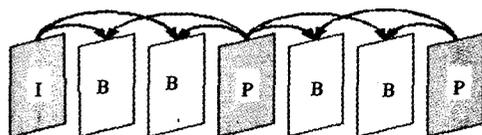


图3 运动补偿

关键帧微调步骤:

- (1) 获取切割点时间所在的视频帧;
- (2) 以该帧所在的位置为中心,向前、向后分别查找距离该帧最近的关键帧;
- (3) 把上述两个关键帧提供给用户选择,或者由系统自动选择其中一个关键帧;
- (4) 修改切割点时间为该关键帧所在的时间,完成关键帧微调。

3.4 多媒体流同步编辑技术

多源流媒体课件的同步编辑和处理包括两种操作:同步切割和同步合并,其核心是多流的同步播放和时间点的同步采集。最终达到的目标是:编辑后的媒体流仍然保持其固有属性(如流类型、视频大小、压缩格式)及其同步结构。通过使用层次化和 OCPN(Object Composition Petri Net)来描述多源流媒体的同步,并基于媒体流之间的时间关系确定同步点,实现时间同步。根据多个媒体流同步播放时的关系,我们提出了两种同步编辑方法:基于公共参考时间轴的同步编辑方法和关键帧的同步编辑方法。这两种方法在某种程度上可看成是与 OCPN 模型一致的,它们实现同步的关键是确定同步点。同步点可以是时间轴上的刻度,也可以看成是媒体本身携带的标记。

(1) 基于时间轴的同步编辑

把多个具有时间关系的媒体流按同一时间轴联系在一起,且首先建立一个公共时间轴,在其上做好多媒体的同步标记(该时间轴的选择可以是独立时间轴,也可以是以其中的一个媒体流为基准),然后,将每个媒体流映射到自己的局部时间轴上,并沿此局部时间轴进行编辑处理。

(2) 基于关键帧的同步编辑

基于关键帧的同步编辑是将与时间相关的动态音视频流媒体对象分解成离散的子单元序列(视频帧、音频抽样值),对象之间的同步由不同对象的相关子单元来说明。一个对象的子单元位置称为参考点,一个参考点可以是一个绝对自然数(如视频帧的帧号,分组数据的分组号等)。这些相关子单元要同时显示,可根据序列号来同时显示或给同时显示的子单元分配一个同步标记。多源流媒体课件点播时,在信息发送端有一同步标记实体,为多媒体数据流中需要同步的地方插入同步标记。同步表现实体在接收端识别出各媒体流中的同步标记单元,然后,就可采取相应的措施来对各媒体流进行同步播放和同步编辑控制。

因为课件的切割点是用户手动输入的,而用户是根据播放过程中课件的瞬间时刻确定的,其格式是时间格式,所以本文采用的是第一种方案——基于时间轴的同步编辑。

4 系统设计与实现

系统中 RM 编辑模块的实现是基于 Real Networks 公司的 Helix Producer SDK 和 Microsoft 公司的 WMP SDK。多源流媒体课件的同步编辑流程如图 4 所示:(1)系统初始化:对控件 RealPlayer G2 和 WMP 进行初始化,同时设置播放进度条;(2)同步播放该课件,设置切除片段的起始时间和结束时间;(3)利用关键帧微调技术对该时间点进行微调;(4)根据时间点进行切割操作,并显示切割进度;(5)把切割后的多个流媒体片段重新合并,合并前需要去掉切除的部分。

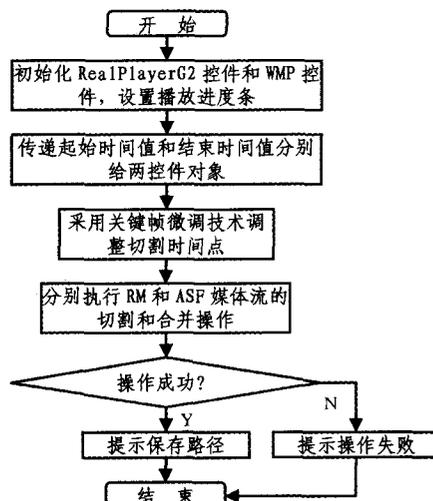


图4 RM 流媒体的切割

结束语 本文研究了 Advanced Streaming Format 和 Real Media 等流媒体的同步切割技术和多流合并技术,提出了流媒体编辑的关键帧微调技术,设计了两种同步编辑多源流媒体课件的方法:基于公共参考时间轴的同步编辑方法和基于关键帧的同步编辑方法。开发了一套多源流媒体课件的同步编辑系统,实现了多源流媒体的同步切割和合并。该系统具有以下特点:

(1) 实现了多源流媒体课件的同步编辑,编辑生成的流媒体课件保持了它固有的同步性,为开发高效优质的多源流媒体课件提供了可能。

(2) 对多源流媒体课件的切割效果较好,消除了流媒体课件切割中经常出现的黑屏、静止画面和静音现象。

下一步研究的方向:拟采用基于关键帧的同步编辑方案,并结合基于内容的音视频检索技术,实现自动选择和剔除不规范的音视频片段。

参考文献

- 1 朱郑州,王茜,吴中福. 流媒体标签技术在网络课件制作中的应用及实现. 计算机工程[J], 2005,31(4):231~233
- 2 Yang C C, Yang Y Z. Design and implementation of the just-in-time retrieving policy for schedule-based distributed multimedia presentations. Journal of Systems and Software [J], 2004,71(1-2): 49~63
- 3 Mahmoud Q H, Dobosiewicz W, Swayne D. Making computer programming fun and accessible [J], Computer, 2004,37(2):106~108
- 4 Zhang Jiajun, Omair A M, Swamy M N S. Quadtree structured region-wise motion compensation for video compression. Circuits and Systems for Video Technology [J]. IEEE Transactions on, 1999,9(5):808~822
- 5 戴彦泓,郭常杰,钟玉琢,孙立峰. 基于细粒度可扩展编码的多源视频流化方案[J]. 清华大学学报(自然科学版), 2004,44(4): 558~562