



(12) 发明专利

(10) 授权公告号 CN 102982832 B

(45) 授权公告日 2015. 05. 27

(21) 申请号 201210484133. 1

CN 1845591 A, 2006. 10. 11, 全文.

(22) 申请日 2012. 11. 24

JP 2000113652 A, 2000. 04. 21, 全文.

(73) 专利权人 安徽科大讯飞信息科技股份有限公司

审查员 樊云

地址 230088 安徽省合肥市高新开发区望江西路 666 号

(72) 发明人 操灿 王松松 张旭 张磊
聂小林

(74) 专利代理机构 北京科迪生专利代理有限公司 11251

代理人 成金玉

(51) Int. Cl.

G11B 27/10(2006. 01)

(56) 对比文件

CN 201611939 U, 2010. 10. 20, 全文.

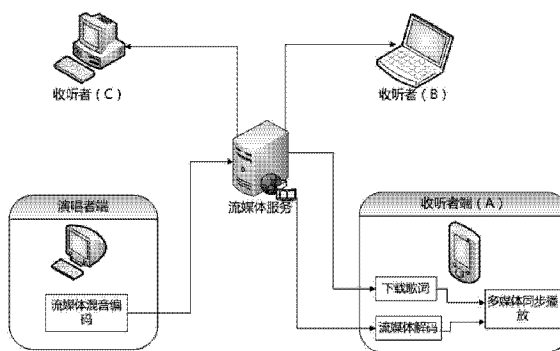
权利要求书1页 说明书4页 附图3页

(54) 发明名称

一种在线卡拉 OK 伴奏、人声与字幕的同步方法

(57) 摘要

一种在线卡拉 OK 伴奏、人声与字幕的同步方法, 在演唱者端进行人声、伴奏混音, 加上播放的时间戳编码后发送到流媒体服务器; 在收听者端, 流媒体播放器线程启动后, 首先从服务器端加载歌词, 然后接收流媒体数据, 采用接收到媒体数据量的大小来同步字幕的播放, 从而实现收听者端与演唱者端卡拉 OK 伴奏、人声与字幕的同步。本发明减少网络带宽的实时传输, 在接收者端多媒体播放线程启动前, 从服务器上下载完字幕; 同时降低 CPU 的计算消耗, 在演唱者端将伴奏与人声进行混音、加上播放时间压缩编码形成多媒体数据块, 接收者端进行解码播放, 且按播放数据量的大小进行字幕同步。



1. 一种在线卡拉 OK 伴奏、人声与字幕的同步方法,其特征在于实现如下步骤;

(1) 在演唱者端,启动混音线程,定义两个缓冲区 MusicBuffer、SoundBuffer 用于缓存伴奏和用户录音数据,当两个缓冲区中的数据到达指定的量 MaxBuffer 时,记录当前的播放时间 PlayedTime,对录音数据先进行 AGC 处理然后再与伴奏进行混音处理,混音数据的时长为单位时间 Unit1Time 的整数倍,混音的数据长度 ProcessTime 为:

$$\text{ProcessTime} = \text{Int}(\text{Min}(\text{MusicBuffer}, \text{SoundBuffer}) / \text{Unit1Time}) * \text{Unit1Time}$$

将混音后的数据进行 MP3 编码,把编码后的数据与已播放的时间戳 TimeStamp 一起通过 AMF 数据格式封装,当演唱者端一首歌曲演唱结束时,达到设定的 TimeStamp 的结束阈值,采用 RTMP 协议发送到流媒体服务器,由流媒体服务器广播给各个收听者端,分别将未混音完成的伴奏数据和录音数据重新加入 MusicBuffer、SoundBuffer 缓冲区,按步骤(1)进行循环处理;

(2) 在接收者端,定义一个播放缓冲区 PlayingBuffer 用于接收演唱者端发送过来的数据,启动数据接收线程、音频播放线程、字幕渐近线程;接收线程将接收到的数据解码成音频数据及时间戳 TimeStamp;播放线程开始准备播放,依据接收到的时间戳与缓冲数据量实时计算出当前播放器相对应的演唱者端的时间 PlayTime,计算公式如下:

$$\text{PlayTime} = \text{TimeStamp} - \text{播放缓冲区 PlayingBuffer 剩余的数据时长};$$

字幕渐近线程依据计算出的播放时间进行字幕渐近,所述进行字幕渐近方法如下:

a. 根据当前播放器相对应的演唱者端的时间 PlayTime,定位到当前播放的字,并提取出当前字的开始时间 Word.StartTime 和结束时间 Word.EndTime;

b. 进行字幕的渐变,处理的长度 = $(\text{PlayTime} - \text{Word.StartTime}) * (\text{字的宽度像素}) / (\text{Word.EndTime} - \text{Word.StartTime})$;

c. 每隔单位时间 Unit2Time,循环执行步骤 a、b;

d. 当当前播放器相对应的演唱者端的时间 PlayTime 的值为设定的 TimeStamp 的结束阈值时,表明演唱已结束,并进行结果的展示。

一种在线卡拉 OK 伴奏、人声与字幕的同步方法

技术领域

[0001] 本发明涉及在线卡拉 OK、多媒体技术领域，公开了一种互联网在线卡拉 OK 模式下，卡拉 OK 伴奏、人声、字幕的同步方法。

背景技术

[0002] 卡拉 OK 是人们休闲娱乐最重要的一种方式，也是线下社交中的保留节目。计算机网络通信作为一种先进的通信方式，一直在改变着人们的生活。而建立在计算机网络通信基础上的在线卡拉 OK，凭借其自身方便、低成本和互动性强等特点，日益受到人们的关注和认同。利用本发明，可以实现网络 KTV，让天涯海角的朋友通过网络，可以聚在一起进行 K 歌，较好地增强了互动，节约交流所需的时间和成本。

[0003] 在线卡拉 OK 系统中，伴奏、人声、字幕的同步技术是一项极为产键的技术，它直接影响到用户体验的好坏。现在的在线卡拉 OK 伴奏、人声、字幕同步方法一般采用传输三路多媒体数据，在每路数据中增加演唱者端演唱的时间，通过流媒体发送到收听者端，收听者端启动多媒体播放器，按照演唱者端演唱的时间，分别播放。

[0004] 在互联网高速发展的今天，从演唱者端同时传输三路数据到收听者端，都能稳定、及时到达。随着移动终端，如智能手机、PDA 等便携电子设备的出现，人们也期望该类设备也能随时随地参与在线卡拉 OK 带来的完美体验。一方面受运营商网络的影响，移动互联网的带宽远远不及传统互联网的带宽，在多路数据传输时，经常造成数据延迟，要么只听到伴奏，要么只听到人声，要么只看到字幕渐进。另一方面移动终端等小型设备的 CPU 性能无法与 PC 比拟，采用多路数据同时播放及相互同步，对 CPU 的计算能力要求高，会造成卡顿现象，用户体验效果非常差。

发明内容

[0005] 本发明技术解决问题：克服现有技术的不足，提供一种在线卡拉 OK 伴奏、人声与字幕的同步方法，减少网络带宽的实时传输，在接收者端多媒体播放线程启动前，从服务器上下载完字幕；同时降低 CPU 的计算消耗，在演唱者端将伴奏与人声进行混音、加上播放时间压缩编码形成多媒体数据块，接收者端进行解码播放，且按播放数据量的大小进行字幕同步。

[0006] 本发明技术解决方案：一种在线卡拉 OK 伴奏、人声与字幕的同步方法，实现步骤如下：

[0007] (1) 在演唱者端，启动混音线程，定义两个缓冲区 MusicBuffer、SoundBuffer 用于缓存伴奏和用户录音数据，当两个缓冲区中的数据到达指定的量 MaxBuffer 时，记录当前的播放时间 PlayedTime，对录音数据先进行 AGC 处理然后再与伴奏进行混音处理，混音数据的时长为单位时间 Unit1Time 的整数倍，混音的数据长度 ProcessTime 为：

[0008] $ProcessTime = \text{Int}(\text{Min}(\text{MusicBuffer}, \text{SoundBuffer}) / \text{Unit1Time}) * \text{Unit1Time}$

[0009] 将混音后的数据进行 MP3 编码，把编码后的数据与已播放的时间戳 TimeStamp 一

起通过 AMF 数据格式封装,当演唱者端一首歌曲演唱结束时,达到设定的 TimeStamp 的结束阈值(-1),TimeStamp 是个整数值,单位是毫秒(ms),用于说明演唱者端当前演唱的时间,也就是伴奏已经播放了的时间,当收听者端接收到这个时间,就会以这个时间为准来同步进行播放声音;采用 RTMP 协议发送到流媒体服务器,由流媒体服务器广播给各个收听者端,分别将未混音完成的伴奏数据和录音数据重新加入 MusicBuffer、SoundBuffer 缓冲区,按步骤(1)进行循环处理;

[0010] (2) 在接收者端,定义一个播放缓冲区 PlayingBuffer 用于接收演唱者端发送过来的数据,启动数据接收线程、音频播放线程、字幕渐近线程;接收线程将接收到的数据解码成音频数据及时间戳 TimeStamp;播放线程开始准备播放,依据接收到的时间戳与缓冲数据量实时计算出当前播放器相对应的演唱者端的时间 PlayTime,计算公式如下:

[0011] $PlayTime = TimeStamp - \text{播放缓冲区 PlayingBuffer 剩余的数据时长};$

[0012] 字幕渐近线程依据计算出的播放时间进行字幕渐近,所述进行字幕渐近方法如下:

[0013] a. 根据当前播放器相对应的演唱者端的时间 PlayTime,定位到当前播放的字,并提取出当前字的开始时间 Word.starttime 和结束时间 Word.EndTime;

[0014] b. 进行字幕的渐变,处理的长度 = $(PlayTime - \text{Word.StartTime}) * (\text{字的宽度像素}) / (\text{Word.EndTime} - \text{Word.StartTime});$

[0015] c. 每隔单位时间 Unit2Time,循环执行步骤 a、b;

[0016] d. 当当前播放器相对应的演唱者端的时间 PlayTime 的值为设定的 TimeStamp 的结束阈值(-1)时,表明演唱已结束,并进行结果的展示。

[0017] 本发明与现有技术相比的优点在于:

[0018] (1) 现有在线卡拉 OK 技术都是在采用 flash 客户端 + 流媒体服务的方式来实现,将伴奏、人声、字幕、同步时间戳通过各个通道进行广播,开发实现起来比较便捷,但也造成以下问题:

[0019] 由于终端设备性能差别较大,flash 客户端的时间戳因性能问题会自动调整,就会产生演唱者端实际播放的时间与录音数据量的偏差,因而造成在收听者端,伴奏、人声、字幕三者不同步的现象;

[0020] 由于网络的延迟,流媒体服务的各个通道上的数据到达的时间也难以保证一致,这样就会出现像伴奏先到,人声迟一点才到收听者端,也会造成在收听者端,伴奏、人声、字幕三者不同步的现象。

[0021] (2) 本发明采用在演唱者端就将伴奏、人声混音完成,在接收者端启动播放线程时先从流媒体服务器上下载完成歌词,当接收到一定量的数据后才开始播放,有效保证了在收听者端,伴奏、人声、字幕三者同步,增强了用户体验。

附图说明

[0022] 图 1 为本发明方法实现的系统的组成框图;

[0023] 图 2 为本发明中在演唱者端将伴奏、人声进行混音编码的实现流程图;

[0024] 图 3 为本发明中在接收者端进行音频解码播放、字幕同步的实现流程图。

具体实施方式

[0025] 下面结合附图及实施例对本发明进一步详细说明。

[0026] 一、系统整体架构流程图如下：

[0027] 在图 1 中可以看到,实现本发明在线卡拉 OK 伴奏、人声、字幕的同步方法的主要流程为：

[0028] 1、在演唱者端进行人声、伴奏混音,加上播放的时间戳编码后发送到流媒体服务器。

[0029] 2、在收听者端,在流媒体播放器线程启动后,就从服务器端加载歌词,然后接收流媒体数据,采用接收到的数据量大小来同步字幕的播放。

[0030] 二、本发明实现详细说明

[0031] 1、图 2 所示,考虑到在实际的录音过程中存在着网络状况不佳,容易导致录音数据与背景音之间存在不同步,需要在混音过程中消除。在实际的唱歌过程中人声会在每一句唱词结束时都会有一定的停顿,为了保证音乐的连续性,使用伴奏作为基准在每句结束处进行声音同步处理。详细的同步逻辑如下：

[0032] (1) 定义 2 个缓冲区 MusicBuffer、SoundBuffer 用于缓存伴奏和用户录音数据,当混音线程开启后实时向缓冲区中添加数据。

[0033] (2) 当缓冲区中的数据到达指定的量 MaxBufferTime 时,记录当前的播放时间 PlayTime,对录音数据先进行 AGC 处理然后再与背景音进行混音处理,处理数据的时长要为单位时间 UnitTime 的整数倍,故处理的数据时长为 $\text{Int}(\text{Min}(\text{MusicBuffer}, \text{SoundBuffer})/\text{UnitTime}) * \text{UnitTime}$ 剩余的数据重新加入缓冲区,等待循环处理。

[0034] (3) 当到达一句结尾时,以伴奏数据为基准进行对齐,当录音数据比伴奏多时,将录音数据多的时候截取,少的时候补零,从而使数据对齐。

[0035] (4) 将混音处理好的数据进行 MP3 编码,把编码后的数据与已播放的时间戳 TimeStamp 一起通过 AMF 数据格式封装,注:时间戳 TimeStamp 精确到毫秒,当演唱者端一首歌曲演唱结束时,TimeStamp 为 -1。采用 RTMP 协议发送到流媒体服务器。

[0036] 2、如图 3 所示,收听者端收到流媒体数据后,启动多媒体播放线程和字幕渐近线程,多媒体播放线程进行解码播放,字幕渐近线程根据播放的数据量及时间戳,进行字幕同步。详细的同步逻辑如下：

[0037] (1) 数据接收线程在获取到数据后将数据交由数据解码线程进行数据解析,得到音频数据及时间戳；

[0038] (2) 将音频数据块放入缓冲队列 PlayingBuffer,当缓存的音频数据达到一定量时,播放线程开始进行播放；

[0039] (3) 在播放时,依据接收到的时间戳 TimeStamp 与缓冲数据量实时计算出当前播放器相对应的演唱者端的时间 PlayTime,计算公式如下：

[0040] $\text{PlayTime} = \text{TimeStamp} - \text{播放缓冲区 PlayingBuffer 剩余的数据时长}$ ；

[0041] (4) 字幕渐近线程依据计算出的播放时间进行字幕渐近,字幕渐近方法如下：

[0042] a. 根据当前播放器相对应的演唱者端的时间 PlayTime,定位到当前播放的字,并提取出当前字的开始时间 Word.endtime 和结束时间 Word.EndTime。

[0043] b. 进行字幕的渐变,处理的长度 = $(\text{PlayTime} - \text{Word.StartTime}) * (\text{字的宽度像})$

素) / (Word.EndTime-Word.StartTime)

[0044] c. 每隔 50 毫秒, 循环执行步骤 a、b。

[0045] d. 当当前播放器相对应的演唱者端的时间 PlayTime 的值为 -1 时, 表明演唱已结束, 并进行结果的展示。

[0046] 本发明未详细阐述部分属于本领域公知技术; 且以上所述, 仅为本发明部分具体实施方式, 但本发明的保护范围并不局限于此, 任何熟悉本领域的人员在本发明揭露的技术范围内, 可轻易想到的变化或替换, 都应涵盖在本发明的保护范围之内。

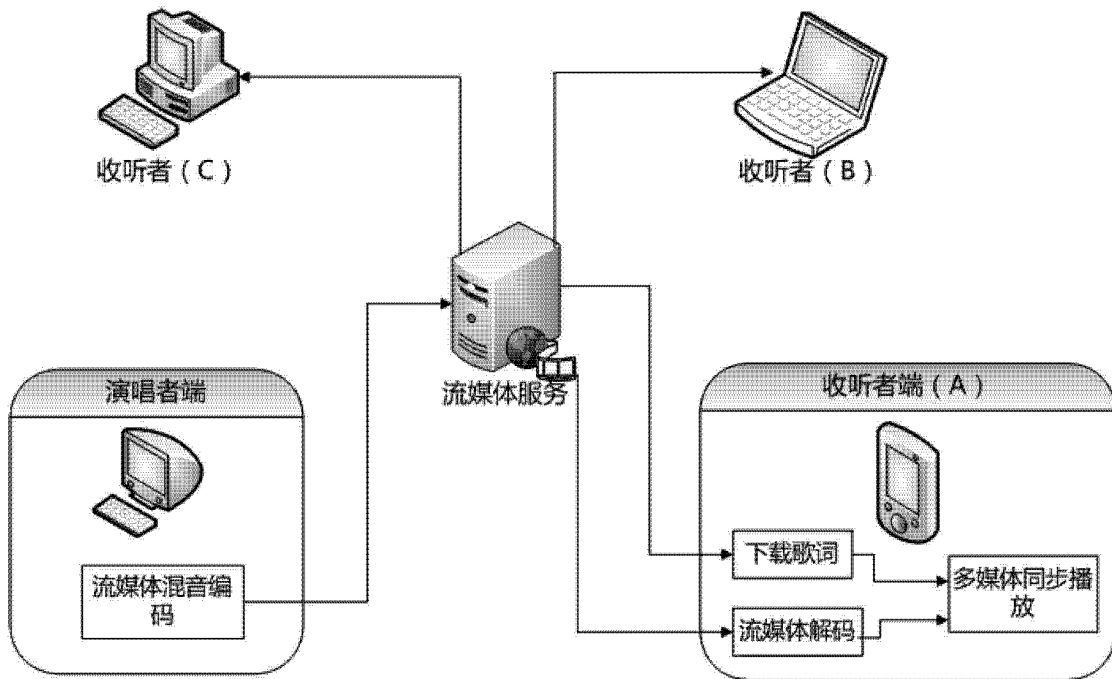


图 1

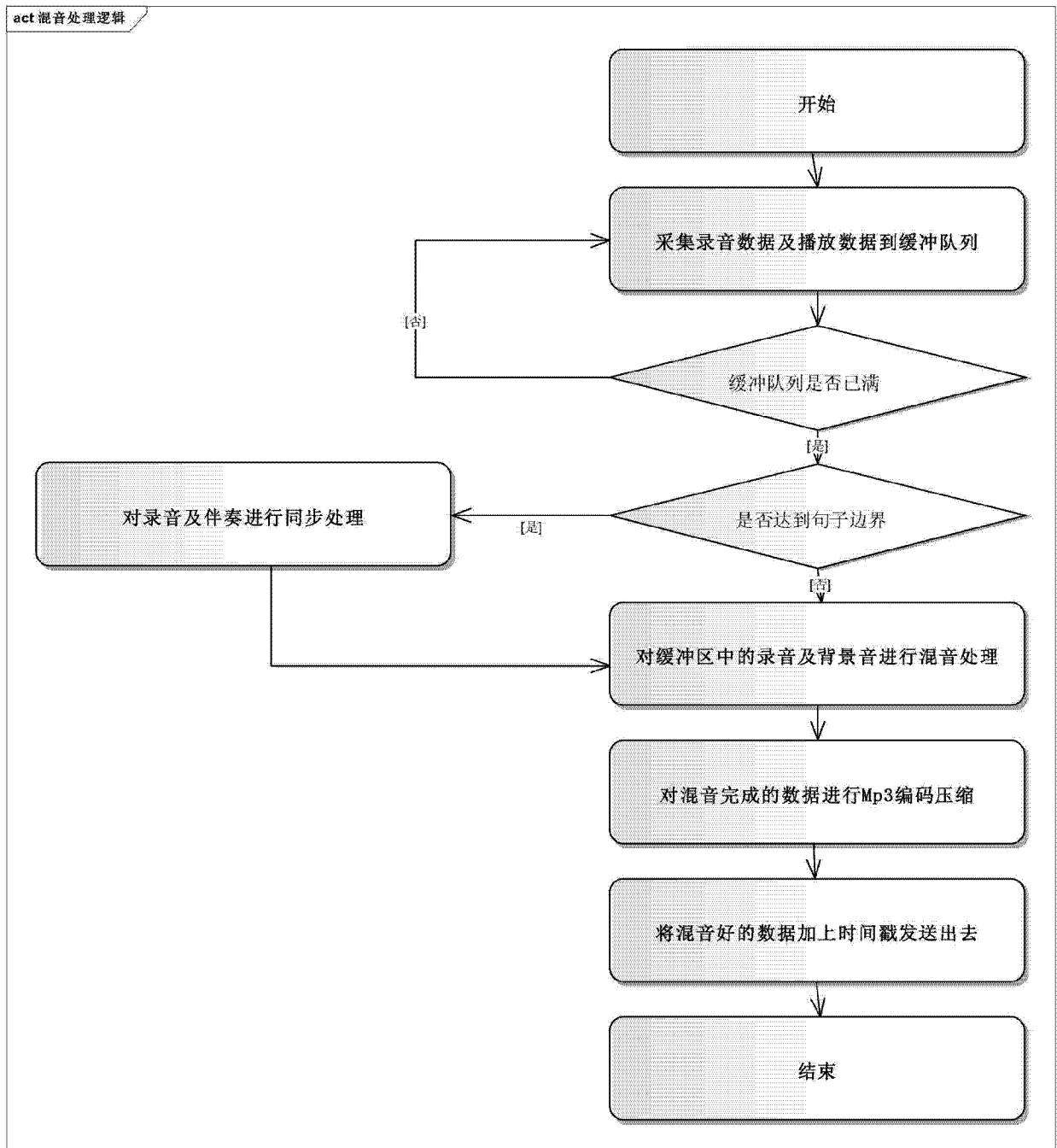


图 2

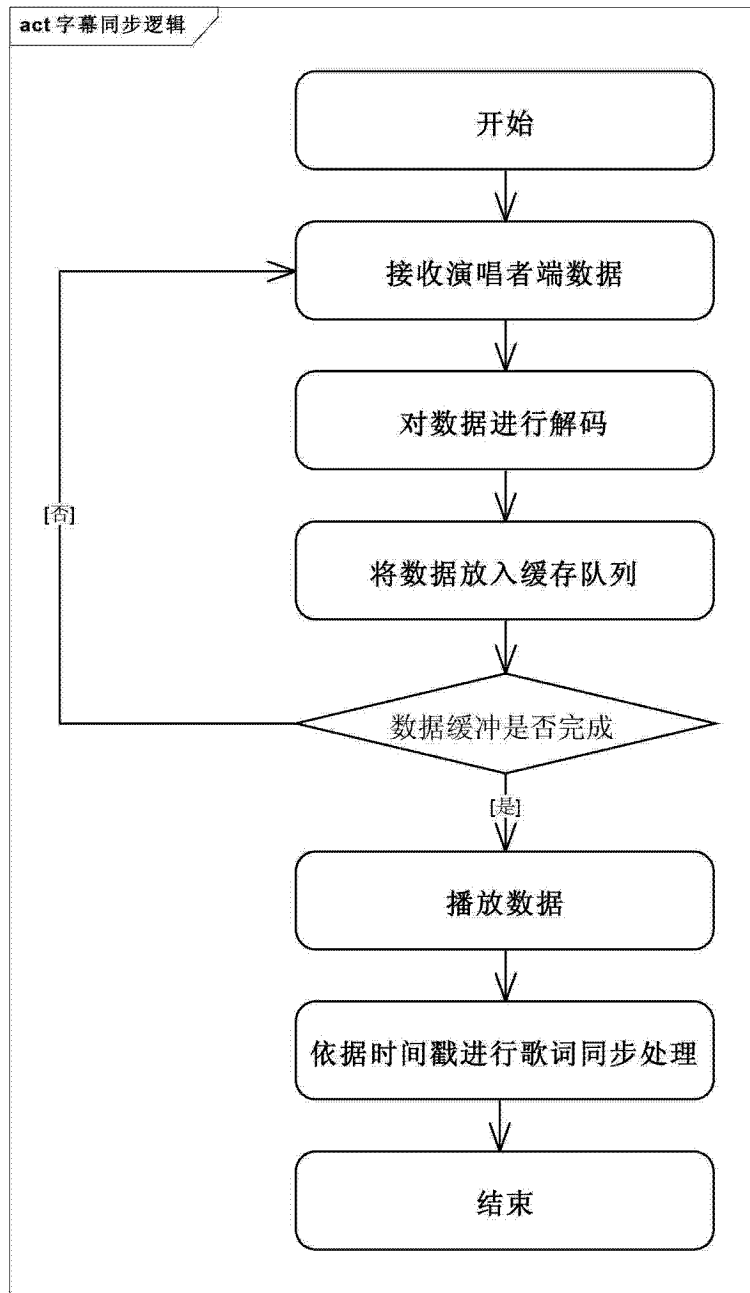


图 3