



(12)发明专利

(10)授权公告号 CN 104796796 B

(45)授权公告日 2018.03.16

(21)申请号 201510190241.1

H04N 21/858(2011.01)

(22)申请日 2015.04.21

H04N 21/6437(2011.01)

(65)同一申请的已公布的文献号

申请公布号 CN 104796796 A

(56)对比文件

WO 2015013411 A1,2015.01.29,

CN 102882865 A,2013.01.16,

CN 103986975 A,2014.08.13,

CN 104243430 A,2014.12.24,

(43)申请公布日 2015.07.22

(73)专利权人 范文鲜

审查员 闫晓宁

地址 518000 广东省深圳市福田区福华三路星河国际花园D2栋6B

(72)发明人 范文鲜

(74)专利代理机构 深圳市中知专利商标代理有限公司 44101

代理人 孙皓

(51)Int.Cl.

H04N 21/845(2011.01)

H04N 21/433(2011.01)

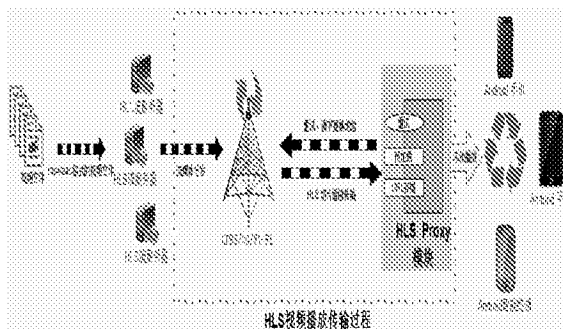
权利要求书1页 说明书7页 附图4页

(54)发明名称

提高Android平台的HLS流播放器容错的方法

(57)摘要

一种效率高且可提高Android平台的HLS播放器容错的方法。包括以下步骤:1)在应用Android操作系统的移动智能终端的网络连接端设置HLS Proxy模块;2)将HLS流服务器提供的播放地址传送到HLS Proxy模块;3)HLS Proxy模块将由网络上下载的流媒体进行缓冲、转码、适配并自动处理流媒体切片的下载、拼接、校验、码流自适应操作并将处理后的流媒体以本地播放地址或文件保存;4)HLS Proxy模块以本地播放器的方式向客户端播放HLS Proxy模块生成的保存在本地播放地址或文件中的流媒体。本发明能显著提升用户的播放体验,有效解决死地址、坏地址、循环嵌套地址等多种问题或副作用。



1. 一种提高Android平台的HLS流播放器容错的方法,包括常规的视频采集,生成视频文件,将该视频文件发送到HLS流服务器,再由HLS流服务器通过互联网对外提供播放地址的方法,其特征在于:还包括以下步骤:

1) 在应用Android操作系统的移动智能终端的网络连接端设置HLS Proxy模块,该HLS Proxy模块用于将不稳定的网络视频转化为稳定的本地流;

2) 将HLS流服务器提供的播放地址传送到HLS Proxy模块;

3) HLS Proxy模块将由网络上下载的流媒体进行缓冲、转码、适配并自动处理流媒体切片的下载、拼接、校验、码流自适应操作并将处理后的流媒体以本地播放地址或文件保存;

所述HLS Proxy模块包括HLS协议的解析和下载模块、视频缓存和组装模块以及WEB容器模块;

所述HLS协议的解析和下载模块,包括进行404处理、死链重试、不规范URL的适配、m3u8文件更新请求和ts片段预加载模块;

所述视频缓存和组装模块,针对采用不同容器格式封装的mp4、3gp、flv文件,按容器的格式规范对片段进行按顺序组装,最终形成单个文件;

所述WEB容器模块,采用简易轻便的mongoose服务器,对局域网内的所有终端提供服务,为本地文件,提供一个局域网在线播放地址;

4) HLS Proxy模块以本地播放器的方式向客户端播放HLS Proxy模块生成的保存在本地播放地址或文件中的流媒体。

2. 根据权利要求1所述的提高Android平台的HLS流播放器容错的方法,其特征在于:所述HLS Proxy模块为采用C++语言开发,基于Linux的嵌入式设备编程方式并供Android程序通过JNI方式实现Java调用C++程序;该模块包括执行404处理、死链重试、不规范URL的适配、m3u8文件更新请求和ts片段预加载模块。

3. 据权利要求1所述的提高Android平台的HLS流播放器容错的方法,其特征在于:所述本地播放地址或文件中的流媒体采用mp4、3gp、flv文件的容器格式封装。

提高Android平台的HLS流播放器容错的方法

技术领域

[0001] 本发明涉及流媒体技术领域,具体涉及一种提高Android平台的HLS流播放器容错的方法。

背景技术

[0002] 流媒体是指采用流式传输的方式在Internet播放的媒体格式。流媒体又叫流式媒体,它是指商家用一个视频传送服务器把节目当成数据包发出,传送到网络上,用户通过解压设备对这些数据进行解压后,节目就会像发送前那样显示出来。

[0003] 流媒体实现的关键技术就是流式传输。

[0004] 流式传输定义很广泛,主要指通过网络传送媒体(如视频、音频)的技术总称。其特定含义为通过Internet将影视节目传送到PC机。实现流式传输有两种方法:实时流式传输(Real time streaming)和顺序流式传输(progressive streaming)。

[0005] 实时流式传输指保证媒体信号带宽与网络连接匹配,使媒体可被实时观看到。实时流式传输根据网络情况调整输出音视频的质量从而实现媒体的持续的实时传送,用户可快进或后退以观看前面或后面的内容。

[0006] 顺序流式传输是顺序下载,在下载文件的同时用户可观看在线媒体,在给定时刻,用户只能观看已下载的那部分,而不能跳到还未下载的前头部分,在传输期间不会根据用户连接的速度对下载顺序做调整。

[0007] 常用的流媒体协议主要有HTTP渐进下载(其为顺序流式传输方式)和基于RTSP/RTP的实时下载(其为实时流式传输方式)流媒体协议。

[0008] 目前适用于复杂的移动互联网环境的是HTTP渐进下载的方法。其中尤以apple公司的HTTP Live Streaming(简称HLS)为代表。HLS最初是苹果公司针对iPhone、iPod、iTouch和iPad等移动设备所开发的流。HTML5(即超文本标记语言)直接其。

[0009] HTTP Live Streaming基于HTTP的流媒体网络传输协议。是苹果公司QuickTime X和ios软件系统的一部分。它的工作原理是把整个流媒体分成一个个小的基于HTTP的文件来下载,每次只下载一些。当流媒体正在播放时,客户端可以选择从许多不同的备用源中以不同的速率下载同样的资源,允许流媒体会话适应不同的数据速率。在开始一个流媒体会话时,客户端会下载一个包含元数据的extended M3U(m3u8) playlist文件,用于寻找可用的流媒体。

[0010] HLS只请求基本的HTTP报文,与实时传输协议(RTP)不同,HLS可以穿过任何允许HTTP数据通过的防火墙或者代理服务器。它也很容易使用内容分发网络来传输流媒体。HLS的这些优势使得其在各大视频网站、客户端中广泛应用。但是国内很多的视频服务器如CNTV、PPTV、LETV和搜狐等对HLS协议的实现有些不同,原生的Android代码不一定兼容得到,或者有些机型的BSP没有实现HLS的协议。另外,Android的原始播放器或市面上一些播放器并没有针对这些问题进行解决,造成用户观看这些视频是可能中断播放甚至是完全播放不了。

[0011] 那么HTTP Live Streaming是怎么样工作的呢。在以前的视频播放技术中,播放模式中必须等待整个文件下载完才行,而在HLS技术中Web服务器向客户端提供接近实时的音视频流。但在使用的过程中使用的是标准的HTTP协议,所以这时,只要使用HLS的技术,就能在普通的HTTP的应用上直接提供点播和直播。

[0012] 具体原理可参见下面的步骤:

[0013] 视频采集->编码器->流分割->流服务器(下发索引文件和视频文件)->客户端。

[0014] 内容准备的过程大约两种,一是视频采集,编码器首先将摄像机实时采集的音视频数据压缩编码为符合特定标准的音视频基本流或者已经编码后的文件(要使用H.264视频和AAC音频编制的编码文件,因为HLS技术只支持H.264视频和AAC音频),然后将编码后的文件封装成符合MPEG-2(如MPEG2TS、MPEG2PS,主要是因为其声音和视频会交织在一起,也会有关键帧来让视频可以直接播放)的封装格式。

[0015] 针对HLS的流媒体分割方式,与所述RTSP的实时下载或普通点播有较大的不同,HLS会将MPEG-2分割成若干个ts的文件。分割过程大多是按时间来切,根据国外的资料,建议切10s一个的文件。另外,分割还有一点不同,就是所用的流分割器会生成一个含有指向这些小ts文件的索引文件。

[0016] 最后,将这些切分了的小一系列的ts文件,放到普通的web服务器中就行了。在CDN(即内容分发网络,全称是Content Delivery Network,)中也是一样,因为请求这些文件会使用标准的HTTP协议。索引文件后缀是.m3u8,索引文件采用扩展的M3U播放列表格式,其实就一文本。

[0017] 内部的视频的地址会是如下形式:

[0018] `http://media.example.com/segment1.ts、http://media.example.com/segment2.ts、http://media.example.com/segment3.ts`等。

[0019] 所以此时可以直接做Cache(即高速缓冲存储器)或者直接将其放到Web服务器中,简单方便。如果MIME(即多用途互联网邮件扩展类型)的信息输出不符的话,要针对每个ts的后缀.m3u8进行相应修改:`.m3u8application/x-mpegURL,.ts video/MP2T`。

[0020] 最后就是客户端,如果是HTML直接在HTML5中支持这种视频可以使用如下标签。

[0021] `<video tabindex="0"height="480"width="640">`

[0022] `<source src="/content1/content1.m3u8">`

[0023] `</video>`

[0024] 如果是应用客户端(Safari QuickTime之类),就得装软件来支持,客户端会根据选择的流的索引来下载文件,当下载了最少二段后开始播放。直接m3u8的索引结束。另外,HTTP可以设计成自适应比特率流,在不同网络环境,选择下载不同码流的视频。

[0025] 综上,HTTP Live Streaming无论是直播还是点播,都能做到近似实时的方式进行流播放。理论上的最小时延是每个切片的长。

[0026] HLS在互联网视频领域应用广泛,在面对移动互联网的各种复杂环境,也遇到了许多问题,HLS协议虽然有规范,但各家具体实现方案不一,导致播放器无法兼容,存在的具体问题如下:

[0027] 1) HLS协议支持相对地址,但播放器不支持多级相对地址。如图1所示,展示的是一个HLS播放地址,携带了多级的相对地址。

[0028] 2) 由于网络等原因,.ts切片丢失造成播放卡顿、黑屏等问题。HLS协议中,允许.ts片段丢失,播放器一般也是请求失败就跳过本片段,但从用户的角度看,就会有体验不好的感觉问题。

[0029] 3) 有些Android机器播放在线视频能力较差,而播放本地视频时则没问题。常规方式和常规播放器,都是在线渐进式下载。

[0030] 4) HLS按照协议规范,将流媒体按设计的下载切片逐个下载,但在真实的网络传输环境中,由于网络传输状况是动态的,如此按部就班的依次下载流媒体,在网络状况较差时则不能圆满、完整的将流媒体下载下来。由此,可能会在用户所用wifi网络速度较快的情况下仍然不能看到较理想、完整的下载视频。

发明内容

[0031] 本发明要解决的技术问题是提供一种效率高且可提高Android平台的HLS播放器容错的方法。

[0032] 为了解决上述技术问题,本发明采用的技术方案为:

[0033] 本发明的提高Android平台的HLS流播放器容错的方法,包括常规的视频采集,生成视频文件,将该视频文件发送到HLS流服务器,再由HLS流服务器通过互联网对外提供播放地址的方法,其特征在于:还包括以下步骤:

[0034] 1) 在应用Android操作系统的移动智能终端的网络连接端设置HLS Proxy模块;

[0035] 2) 将HLS流服务器提供的播放地址传送到HLS Proxy模块;

[0036] 3) HLS Proxy模块将由网络上下载的流媒体进行缓冲、转码、适配并自动处理流媒体切片的下载、拼接、校验、码流自适应操作并将处理后的流媒体以本地播放地址或文件保存;

[0037] 4) HLS Proxy模块以本地播放器的方式向客户端播放HLS Proxy模块生成的保存在本地播放地址或文件中的流媒体。

[0038] 所述HLS Proxy模块为采用C++语言开发,基于Linux的嵌入式设备编程方式并供Android程序通过JNI方式实现Java调用C++程序;该模块包括执行404处理、死链重试、不规范URL的适配、m3u8文件更新请求和ts片段预加载模块。

[0039] 所述本地播放地址或文件中的流媒体采用mp4、3gp、flv文件的容器格式封装。

[0040] 与现有技术相比,本发明在应用Android操作系统的移动智能终端(如智能手机、平板电脑等)的网络连接端设置HLS Proxy模块,该模块可使其在下载网络传输的流媒体时,使接收到流媒体先经过该HLS Proxy模块进行缓冲、转码、适配并自动处理流媒体切片的下载、拼接、校验、码流自适应等操作,由此将所下载的流媒体合并成为一个本地文件(file://方式本地播放),既提高了容错能力,又降低机器的播放门槛,该模块将下载的文件缓存到本地后,再经Android手机播放,所获得的流媒体文件完整和稳定。

[0041] 本发明能显著的提升用户的播放体验,并且具有以下优点:1)降低Android手机硬件平台要求,通过HLS Proxy模块,可提供本地http url播放或本地文件播放两种方式,提高流媒体播放的兼容性。2)针对各家视频提供商的不同的HLS实现进行优化适配,有效解决死地址、坏地址、循环嵌套地址等多种由于方案实现的不同、网络环境原因、防盗链跳转等带来的问题或副作用。3)智能地预判和预加载机制,极大保障用户体验。如果预判到网络条

件允许,我们将预加载,避免用户网络差时无法下载提高用户体验。4)灵活的404错误重试机制。许多播放器在播放HLS流尤其是HLS直播时,由于服务器响应不过来、或切片未生产、或地址跳转、地址过时失效等,都有可能产生提示“404错误”的问题,从而终止播放。我们通过HLS Proxy模块的接入,有效的优化下载协议和重试机制,更进一步地保障用户能成功的下载。5)Android 4.0及以后的终端,才开始实现HLS流的播放,也就是很多手机不支持HLS流的播放,通过本模块的转化,可以实现终端对HLS的播放。

附图说明

- [0042] 图1是HLS播放地址示意图。
[0043] 图2是现有技术的通用播放过程示意图。
[0044] 图3是根据本发明的提高HLS播放器容错的方法的原理图。
[0045] 图4是本发明的HLS Proxy的结构图。

具体实施方式

[0046] 为使本发明的目的、技术方案和优点更加清楚明白,以下结合具体实施例,并参照附图,对本发明进一步详细说明。

[0047] 首先,对本发明可能用到的术语进行解释。

[0048] 1) HLS (HTTP Live Streaming): Apple的动态码率自适应技术。主要用于PC和Apple终端的音视频服务。包括一个m3u (8)的索引文件,TS媒体分片文件和key加密串文件,具体详见本说明书“背景技术”部分中的描述。

[0049] 2) Android: Android是一种基于Linux的自由及开放源代码的操作系统,主要使用于移动设备,如智能手机和平板电脑,由Google公司和开放手机联盟领导及开发。

[0050] 3) BSP: 板级支持包 (BSP) 是介于主板硬件和操作系统中驱动层程序之间的一层,一般认为它属于操作系统一部分,主要是实现对操作系统的支持,为上层的驱动程序提供访问硬件设备寄存器的函数包,使之能够更好的运行于硬件主板。在嵌入式系统软件的组成中,就有BSP。BSP是相对于操作系统而言的,不同的操作系统对应于不同定义形式的BSP,例如VxWorks的BSP和Linux的BSP相对于某一CPU来说尽管实现的功能一样,可是写法和接口定义是完全不同的,所以写BSP一定要按照该系统BSP的定义形式来写 (BSP的编程过程大多数是在某一个成型的BSP模板上进行修改)。这样才能与上层OS保持正确的接口,良好的支持上层OS。

[0051] 3) QuickTime: QuickTime是一款拥有强大的多媒体技术的内置媒体播放器,可让用户以各式各样的文件格式观看互联网视频、高清电影预告片和个人媒体作品,更可让你以非比寻常的高品质欣赏这些内容。QuickTime不仅仅是一个媒体播放器,而且是一个完整的多媒体架构,可以用来进行多种媒体的创建,生产和分发,并为这一过程提供端到端的支持:包括媒体的实时捕捉,以编程的方式合成媒体,导入和导出现有的媒体,还有编辑和制作,压缩,分发,以及用户回放等多个环节。

[0052] 3) Ios: 苹果iOS是由苹果公司开发的移动操作系统。苹果公司最早于2007年1月9日的Macworld大会上公布这个系统,最初是设计给iPhone使用的,后来陆续套用到iPod touch、iPad以及Apple TV等产品上。iOS与苹果的Mac OS X操作系统一样,它也是以Darwin

为基础的,因此同样属于类Unix的商业操作系统。原本这个系统名为iPhone OS,因为iPad,iPhone,iPod Touch都使用iPhone OS,所以2010WWDC大会上宣布改名为iOS。

[0053] 4) RTP:RTP(Real-time Transport Protocol,实时传输协议)是一个网络传输协议,它是由IETF的多媒体传输工作小组1996年在RFC 1889中公布的,后在RFC3550中进行更新。

[0054] 5) Proxy Server:代理服务器(Proxy Server)是一种重要的服务器安全功能,它的工作主要在开放系统互联模型的会话层,从而起到防火墙的作用。代理服务器大多被用来连接INTERNET(国际互联网)和INTRANET(局域网)。

[0055] 6) MPEG2-TS:MPEG2-TS(Transport Stream“传输流”;又称TS、TP、MPEG-TS或M2T)是用于音效、图像与数据的通信协定,最早应用于DVD的实时传送节目。DVD节目中的MPEG2格式,确切地说是MPEG2-PS,全称是Program Stream(程序流),而TS的全称则是Transport Stream(传输流)。MPEG2-PS主要应用于存储的具有固定时长的节目,如DVD电影,可添加字幕等一些程序操作。而MPEG-TS则主要应用于实时传送的节目,比如实时广播的电视节目。简单地说,将DVD上的VOB文件的前面一截cut掉(或者是数据损坏数据)就会导致整个文件无法解码,而电视节目是任何时候打开电视机都能解码(收看)的。所以MPEG2-TS格式的特点就是从视频流的任一片段开始都是可以独立解码。

[0056] 7) WIFI:Wi-Fi这个术语是指无线保真(Wireless Fidelity),类似历史悠久的音频设备分类:长期高保真(1930年开始采用)或Hi-Fi的(1950年开始采用)。即使Wi-Fi联盟本身也经常在新稿和文件中使用“无线保真”这个词,Wi-Fi还是出现在ITAA的一个论文中。然而,根据菲尔贝朗格的语句,Wi-Fi术语应该是没有任何意义的。IEEE 802.11第一个版本发表于1997年,其中定义了介质访问接入控制层和物理层。物理层定义了工作在2.4GHz的ISM频段上的两种无线调频方式和一种红外传输的方式,总数据传输速率设计为2Mbit/s。两个设备之间的通信可以自由直接(ad hoc)的方式进行,也可以在基站(Base Station,BS)或者访问点(Access Point,AP)的协调下进行。

[0057] 8) HTTP 404错误:是指链接指向的网页不存在,即原始网页的URL失效,这种情况经常会发生,很难避免,比如说:网页URL生成规则改变、网页文件更名或移动位置、导入链接拼写错误等,导致原来的URL地址无法访问;当Web服务器接到类似请求时,会返回一个404状态码,告诉浏览器要请求的资源并不存在。

[0058] 图2是现有的通用的播放过程示意图。根据现有的播放过程,问题主要发生在视频播放的传输过程和在线流媒体解码阶段。如网络不畅通、视频协议的具体实现、Android原生播放器对流媒体的支持度等都有可能导致播放失败,容错度较低,播放体验不好。

[0059] 图3为本发明的方法的原理图。

[0060] 本发明的核心在于终端播放之前增加一个HLS Proxy模块。通过专门研发的HLS Proxy模块,提供给各个Android系统,可供Android终端的内置播放器、第三方应用等进行调研,并且具有极强的拓展性,本发明的方法不仅能够提高HLS的容错处理能力,同时也具备了一个本地局域网流媒体的功能。

[0061] 考虑到方案的便于移植性,本发明采用标准C++进行开发,采用基于Linux的嵌入式设备编程方式,最终该模块的提供方式为.so库。可供Android程序通过JNI方式实现Java调用C++。

[0062] 本发明的方法包括以下步骤：

[0063] 步骤1,采集生成视频文件。

[0064] 在该步骤,通过摄像头等进行视频采集,保存视频的格式为H264/AAC。当然,视频也可以是从网上下载或是合作方提供等等。

[0065] 这一步不是本发明的内容,不过为了阐述明白一个完整HLS流从制作到发布到播放的过程,提及一下容易明白。

[0066] 步骤2,将步骤1采集的视频文件发送到HLS流服务器。

[0067] 在该步骤,HLS服务器会对视频文件进行切片,并按顺序建立m3u索引文件。

[0068] 本步骤同样也属于HLS播放过程的说明,不是本发明内容,各家互联网公司的HLS流,都需要依托于一个HLS流服务器,才能对外提供一个http播放地址,如http://example.com/playlist.m3u8。本文不再具体参数这个过程。

[0069] 步骤3,HLS流服务器通过互联网对外提供播放地址。

[0070] 在该步骤,只要是互联网上的用户,就能通过http协议方式请求流媒体数据(不考虑流服务器的鉴权问题)。一般的播放情况下,如果没有本发明的HLSProxy模块,用户也是基本可以直接播放视频的。

[0071] 本发明,正是通过把这个原先的播放地址,传入下一步骤中的HLS Proxy模块进行转换,从而提供出来新的播放地址,进而达到更为稳定的播放体验。

[0072] 步骤4,把HLS流服务器播放地址传送到HLS Proxy模块,再对客户端提供转化后的流播放地址或文件。

[0073] 该步骤是本发明最关键的步骤,与现有技术的直接播放HLS流相比,该步骤是新增加的。在该步骤,通过HLS Proxy模块的中间缓冲和转化,解决直接播放HLS时会出现的各类问题。通过这个模块,把不稳定或有问题的网络视频流,转化为稳定的本地流,使得播放的容错度、用户体验等就能有较明显的提升。

[0074] 参照图4,该HLS Proxy模块通过以下组成部分,来优化和提高HLS播放器的容错能力:

[0075] 1) HLS协议的解析和下载模块。在该模块中还包括进行404处理、死链重试、不规范URL的适配、m3u8文件更新请求和ts片段预加载模块等。

[0076] 根据现有技术,HLS流媒体是通过不断生成视频切片形式来实现边下载边播放的,每一个视频切片都有其下载地址,在实际使用环境下,经常会遇到索引文件中已经有地址,而实际切片尚未生成、或是切片已经过期等等导致切片不存在的问题,那文件不存在,则在Http服务器中的表现就是404文件不存在或是链接网址无效的错误,一般播放器收到这样的错误,都会误以为视频已经完成播放或是视频播放错误,从而退出播放。那么,本发明通过HLS Proxy模块进行优化、重试,就能实现继续播放,从而解决这个问题。

[0077] 通过HLS Proxy模块,可以实现视频切片文件的顺利下载。

[0078] 2) 视频缓存和组装模块。在HLS协议的解析和下载模块中实现了文件的下载,而视频缓存和组装模块则针对采用不同容器格式封装的mp4、3gp、flv文件,按容器的格式规范对片段进行按顺序组装,最终形成单个文件。

[0079] 视频缓存和组装模块的转码采用的是开源框架ffmpeg。如下载好的ts切片名为a0001.mp4,它的视频格式可能是非h264的,而Android/IOS的默认支持的格式是H264/aac,

那么通过ffmpeg的转码命令就可以实现格式的转化:

[0080] `ffmpeg-i aac-i h264in a0001.mp4out 00001.mp4`

[0081] 如此,本地缓存的00001.mp4就是规范的H264/aac视频了。另外,还可以通过形如以下的ffmpeg命令就能实现视频00001.mp4、00002.mp4合并为video1.mp4,并存放在SD卡根目录下:

[0082] `ffmpeg-i 00001.mp4i-i 00002.mp4-f mp4-acodec libfaac-vcodec libx264-sameq/sdcard/video1.mp4`

[0083] 这样在终端中,就可以通过file文件传输协议进行播放。无需额外的支持,直接播放file://sdcard/video1.mp4即可,通过这一步可大大提高低端机、老机的播放能力。

[0084] 通过视频缓存和组装模块,可以实现视频文件的组装,能支持file://方式本地播放。并对下一部分WEB容器模块提供按顺序的、整理过的切片。

[0085] 3) WEB容器模块。采用简易轻便的mongoose服务器,可对局域网内的所有终端提供服务。可把本地文件,提供出一个局域网在线播放地址。如上一步生成的视频文件切片,都将存放在本服务器的web容器目录,如在/Web/WWW/目录下,有以下文件:

[0086] `Playlist1.m3u8、00001.mp4、00002.mp4、00003.mp4...`

[0087] 那么,原HLS视频的新播放地址就是http://127.0.0.1/Playlist1.m3u8。在这个m3u8描述文件中,存在http://127.0.0.1/00001.mp4、http://127.0.0.1/00002.mp4等具体视频切片的播放地址,这里的协议就不再是相对复杂的HLS协议了,而是普通的单文件http下载协议,这在Android平台的兼容度几乎是100%的。

[0088] 步骤5,本地播放器应用播放HLS Proxy模块所生成的新的播放地址。

[0089] 如果是当前机器的播放器,由于本地mongoose服务器已经在步骤4中生成了本地播放地址,所以播放器只需要播放http://127.0.0.1/Playlist.m3u8就可以了。

[0090] 由于http协议的穿透性很高,所以,若是当前手机是在wifi环境中,如它的ip地址是192.168.2.10,那么同一个wifi局域网内的用户,还可以通过http://192.168.2.10/Playlist.m3u8来实现对播放地址的访问。从而可以看到通过本部分,还能在本地或是局域网,提供一个新的、兼容性更高的播放地址。

[0091] 另外,通过模块设计的框架,后续还支持跟多的扩展,也给本方案的进一步使用推广,保留了更多的可能性。

[0092] 以上所述的具体实施例,对本发明的目的、技术方案和有益效果进行了进一步详细说明,应理解的是,以上所述仅为本发明的具体实施例而已,并不用于限制本发明,凡在本发明的精神和原则之内,所做的任何修改、等同替换、改进等,均应包含在本发明的保护范围之内。

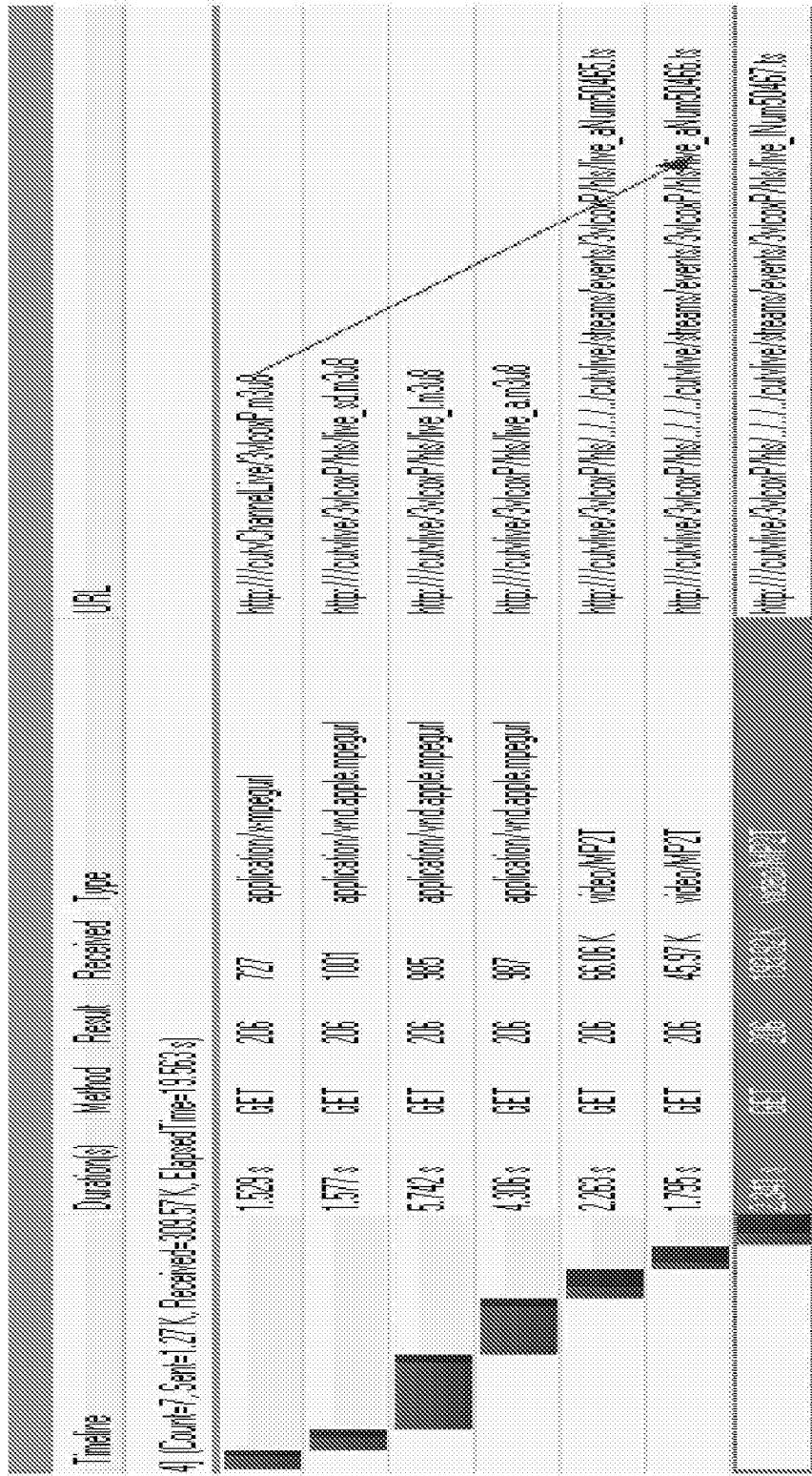


图1

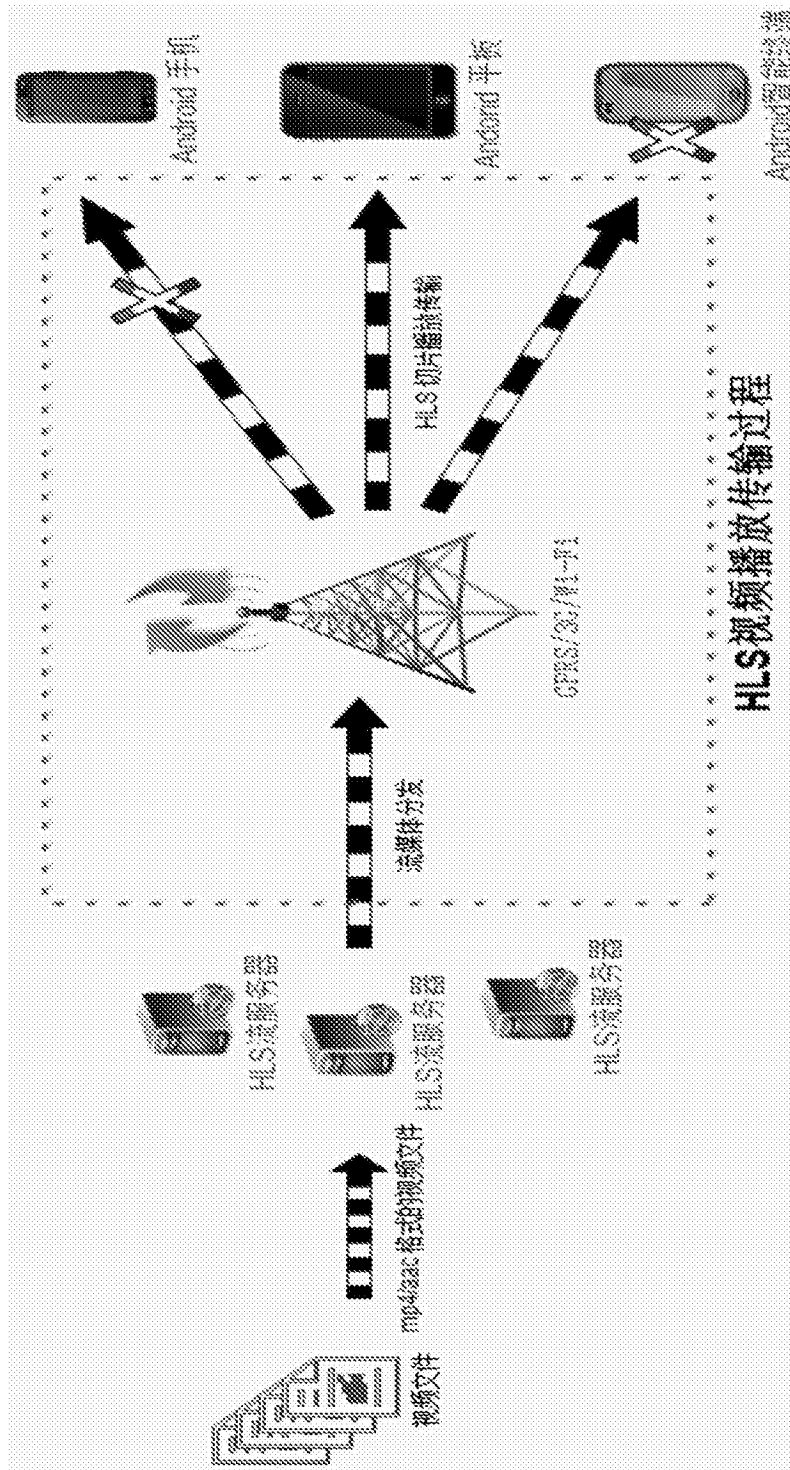


图2

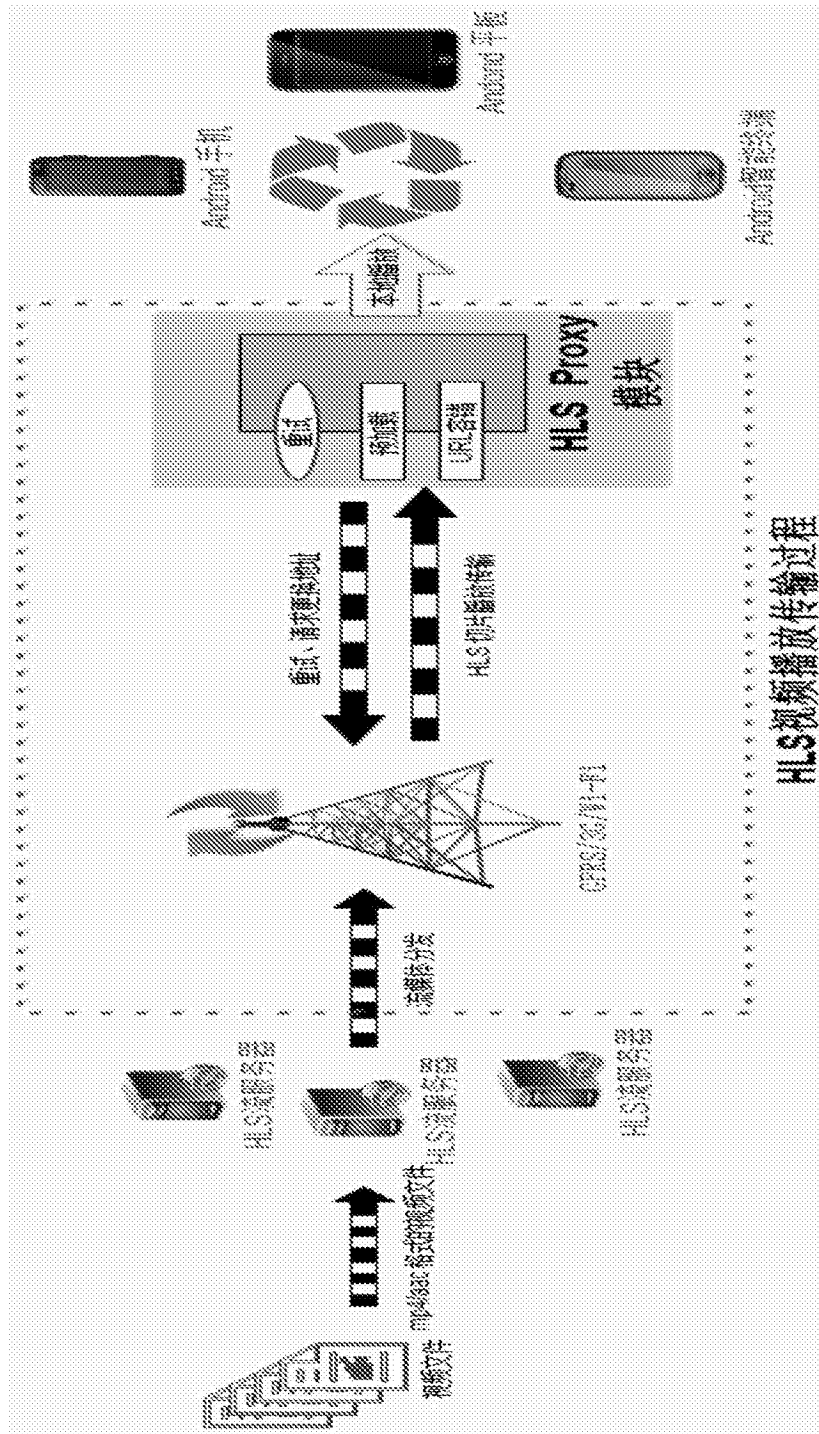


图3

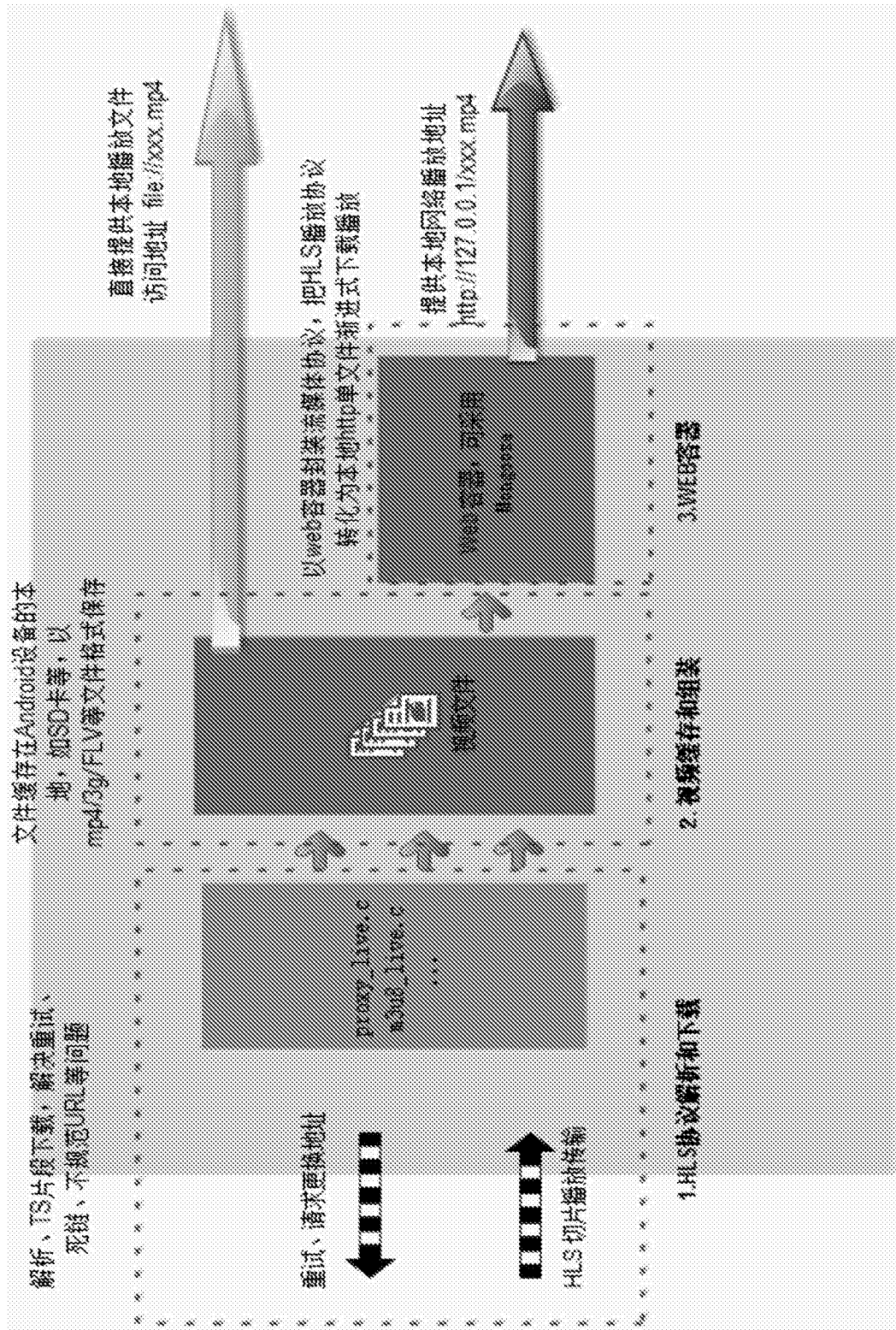


图4