



(12) 发明专利

(10) 授权公告号 CN 109302636 B

(45) 授权公告日 2022.05.27

(21) 申请号 201710608608.6

H04N 21/438 (2011.01)

(22) 申请日 2017.07.24

H04N 21/433 (2011.01)

(65) 同一申请的已公布的文献号

H04N 21/81 (2011.01)

申请公布号 CN 109302636 A

(56) 对比文件

(43) 申请公布日 2019.02.01

CN 103226596 A, 2013.07.31

(73) 专利权人 阿里巴巴集团控股有限公司

CN 106254940 A, 2016.12.21

地址 英属开曼群岛大开曼资本大厦一座四
层847号邮箱

CN 106610721 A, 2017.05.03

CN 106231317 A, 2016.12.14

CN 103838851 A, 2014.06.04

(72) 发明人 宁宇

WO 2017116952 A1, 2017.07.06

US 9473758 B1, 2016.10.18

(74) 专利代理机构 北京润泽恒知识产权代理有
限公司 11319

审查员 曹轶乐

专利代理师 苏培华

(51) Int. Cl.

H04N 21/44 (2011.01)

H04N 21/231 (2011.01)

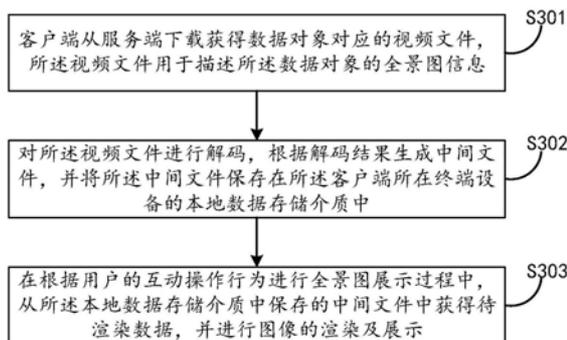
权利要求书3页 说明书12页 附图3页

(54) 发明名称

提供数据对象全景图信息的方法及装置

(57) 摘要

本申请实施例公开了提供数据对象全景图信息的方法及装置,其中,所述方法包括:客户端从服务端下载获得数据对象对应的视频文件,所述视频文件用于描述所述数据对象的全景图信息;对所述视频文件进行解码,根据解码结果生成中间文件,并将所述中间文件保存在所述客户端所在终端设备的本地数据存储介质中;在根据用户的互动操作行为进行全景图展示过程中,从所述本地数据存储介质中保存的中间文件中获得待渲染数据,并进行图像的渲染及展示。通过本申请实施例,可以实现即使配置较低的终端设备,也能够支持高清全景图的流畅展示。



1. 一种提供数据对象全景图信息的方法,其特征在于,包括:

客户端从服务端下载获得数据对象对应的视频文件,所述视频文件用于描述所述数据对象的全景图信息;

对所述视频文件进行解码,根据解码结果生成中间文件,并将所述中间文件保存在所述客户端所在终端设备的本地数据存储介质中;

在根据用户的互动操作行为进行全景图展示过程中,确定当前需要展示的目标帧,采用内存映射文件的方式,从所述本地数据存储介质中保存的中间文件中以帧为单位获得待渲染数据,并基于所述待渲染数据进行图像的渲染及展示;所述待渲染数据为所述目标帧对应的数据;

在切换到下一帧进行渲染时,将内存中当前帧对应的数据清除。

2. 根据权利要求1所述的方法,其特征在于,所述从服务端下载获得数据对象对应的视频文件,包括:

在接收到访问数据对象信息页面的请求时,从服务端下载所述页面的页面数据,以及所述数据对象的视频文件。

3. 根据权利要求1所述的方法,其特征在于,所述从服务端下载获得数据对象对应的视频文件,包括:

在展示数据对象信息页面的过程中,如果接收到查看所述数据对象全景图信息的请求,则从服务端下载获得所述数据对象的视频文件。

4. 根据权利要求1所述的方法,其特征在于,对所述视频文件进行解码,根据解码结果生成中间文件,包括:

创建中间文件;

对所述视频文件进行逐帧解码;

根据每一帧的解码结果生成字节流,并以所述字节流的形式添加到所述中间文件的末尾。

5. 根据权利要求4所述的方法,其特征在于,所述根据每一帧的解码结果生成字节流,包括:

按照预置的压缩格式对每一帧的解码结果进行压缩;

按照压缩后的数据生成待添加到所述中间文件末尾的字节流;

所述进行图像渲染及绘制之前,还包括:

对所述中间文件中的数据进行解压缩处理。

6. 根据权利要求4所述的方法,其特征在于,还包括:

创建元数据文件;

在所述元数据文件中保存每一帧对应的写入所述中间文件的总字节长度,以及相对于所述中间文件起始位置的偏移量,以用于根据所述偏移量以及所述总字节长度,从所述中间文件中确定每一帧对应数据内容。

7. 根据权利要求1至6任一项所述的方法,其特征在于,所述以帧为单位从所述中间文件中获得对应的数据,包括:

根据所述互动操作行为确定待渲染帧的位置;

根据所述待渲染帧的位置从所述中间文件中获得所述待渲染帧对应的数据,并提供给

渲染进程进行渲染处理。

8. 根据权利要求1至6任一项所述的方法,其特征在于,所述进行图像渲染及展示包括:通过OpenGL技术将对应图像绘制到显示屏上。

9. 一种提供数据对象全景图信息的方法,其特征在于,包括:

服务端以视频文件的形式保存数据对象的全景图信息;

接收到客户端的下载请求时,将所述视频文件提供给所述客户端,由所述客户端对所述视频文件进行解码,根据解码结果生成中间文件,并将所述中间文件保存在所述客户端所在终端设备的本地数据存储介质中,在根据用户的互动操作行为进行全景图展示过程中,确定当前需要展示的目标帧,采用内存映射文件的方式,从所述本地数据存储介质中保存的中间文件中以帧为单位获得待渲染数据,并基于所述待渲染数据进行图像的渲染及展示;所述待渲染数据为所述目标帧对应的数据。

10. 一种提供数据对象全景图信息的装置,其特征在于,应用于客户端,包括:

视频文件下载单元,用于从服务端下载获得数据对象对应的视频文件,所述视频文件用于描述所述数据对象的全景图信息;

中间文件生成单元,用于对所述视频文件进行解码,根据解码结果生成中间文件,并将所述中间文件保存在所述客户端所在终端设备的本地数据存储介质中;

数据读取单元,用于在根据用户的互动操作行为进行全景图展示过程中,确定当前需要展示的目标帧,采用内存映射文件的方式,从所述本地数据存储介质中保存的中间文件中以帧为单位获得待渲染数据,并基于所述待渲染数据进行图像的渲染及展示;所述待渲染数据为所述目标帧对应的数据;

内存数据清除单元,用于在切换到下一帧进行渲染时,将内存中当前帧对应的数据清除。

11. 一种提供数据对象全景图信息的装置,其特征在于,应用于服务端,包括:

视频文件保存单元,用于以视频文件的形式保存数据对象的全景图信息;

视频文件提供单元,用于接收到客户端的下载请求时,将所述视频文件提供给所述客户端,由所述客户端对所述视频文件进行解码,根据解码结果生成中间文件,并将所述中间文件保存在所述客户端所在终端设备的本地数据存储介质中,在根据用户的互动操作行为进行全景图展示过程中,确定当前需要展示的目标帧,采用内存映射文件的方式,从所述本地数据存储介质中保存的中间文件中以帧为单位获得待渲染数据,并基于所述待渲染数据进行图像的渲染及展示,所述待渲染数据为所述目标帧对应的数据;在切换到下一帧进行渲染时,将内存中当前帧对应的数据清除。

12. 一种提供数据对象全景图信息的方法,其特征在于,包括:

在本地数据存储介质中保存视频文件对应的中间文件,所述视频文件用于描述数据对象的全景图信息;所述中间文件用于记录所述视频文件的解码结果信息;

在根据用户的互动操作行为进行全景图展示过程中,确定当前需要展示的目标帧,采用内存映射文件的方式,从所述中间文件中以帧为单位获得待渲染数据,以基于所述待渲染数据进行图像的渲染及展示;所述待渲染数据为所述目标帧对应的数据;

在切换到下一帧进行渲染时,将内存中当前帧对应的数据清除。

13. 一种提供数据对象全景图信息的装置,其特征在于,包括:

文件保存单元,用于在本地数据存储介质中保存视频文件对应的中间文件,所述视频文件用于描述数据对象的全景图信息;所述中间文件用于记录所述视频文件的解码结果信息;

数据获得单元,用于在根据用户的互动操作行为进行全景图展示过程中,确定当前需要展示的目标帧,采用内存映射文件的方式,从所述中间文件中以帧为单位获得待渲染数据,以基于所述待渲染数据进行图像的渲染及展示;所述待渲染数据为所述目标帧对应的数据;

内存数据清除单元,用于在切换到下一帧进行渲染时,将内存中当前帧对应的数据清除。

14. 一种电子设备,其特征在于,包括:

一个或多个处理器;以及

与所述一个或多个处理器关联的存储器,所述存储器用于存储程序指令,所述程序指令在被所述一个或多个处理器读取执行时,执行如下操作:

从服务端下载获得数据对象对应的视频文件,所述视频文件用于描述所述数据对象的全景图信息;

对所述视频文件进行解码,根据解码结果生成中间文件,并将所述中间文件保存在客户端所在终端设备的本地数据存储介质中;

在根据用户的互动操作行为进行全景图展示过程中,确定当前需要展示的目标帧,采用内存映射文件的方式,从所述本地数据存储介质中保存的中间文件中以帧为单位获得待渲染数据,并基于所述待渲染数据进行图像的渲染及展示;所述待渲染数据为所述目标帧对应的数据;

在切换到下一帧进行渲染时,将内存中当前帧对应的数据清除。

提供数据对象全景图信息的方法及装置

技术领域

[0001] 本申请涉及全景图信息处理技术领域,特别是涉及提供数据对象全景图信息的方法及装置。

背景技术

[0002] 随着移动互联网的发展,手机等移动终端设备已经支持绝大多数的多媒体形式,例如,gif图片、视频等,在内容的丰富性提高的同时,用户对更具备交互性、更立体化的多媒体形式表现出了更强的诉求。基于该诉求,在一些网络销售等系统中,提供了商品全景展示功能。

[0003] 现有的商品全景展示方案主要有两种,一种是基于3D建模,通过专业设备对目标商品进行多次扫描和大量复杂计算后得到相应的模型文件,建模费用昂贵,门槛较高,模型数据格式多样,很难大规模推广;另一种是基于视频技术的商品全景展示,在制作视频时可以将商品放置在旋转展示台上,采用任意摄影器材(手机亦可)拍摄一段商品自转一周的视频,在展示的时候,根据陀螺仪或手势等用户操作,即时切换展示视频中某一帧,从而还原出一个更全面、更立体的商品的全貌。该方案制作成本低廉,但是在现有技术中,却面临占用用户设备较多的CPU、内存等硬件资源,或者占用较多传输资源等原因,无法支持高清的商品全景。

[0004] 例如,在其中一种现有技术中,在服务端存储视频文件,用户需要查看某商品的全景图像时,从服务端下载视频文件到本地,然后在本地随着陀螺仪或者用户手势的变化进行解码播放。但是,由于当前的视频技术主要通过消除连续图像之间的时域冗余信息来压缩视频。以压缩中常见的I帧、P帧为例,I帧在解码时只需要本帧数据就可以完成,而P帧需要当前帧和前一个P帧或者I帧才能生成最终画面,所以如果直接基于视频技术,从后向前切换帧的时候,需要先找到目标帧在时间轴上的前一个I帧并解码,然后再对目标帧解码,切换时间会大大延长,在移动设备上很容易造成掉帧,引起较明显的卡顿。同时视频文件在播放过程中需要实时解码,这对手机等用户设备的性能有较高的要求,时间长了手机发热会较明显。

[0005] 鉴于上面的原因,为了避免在交互过程中直接进行视频帧的解码,现有技术的处理方式是将视频中所有帧先解码好的图像数据存储在内存中,在用户交互时,直接从内存中获取对应的图像数据展示。该方案可以快速获取视频中某一帧的图像数据,但对设备可用内存有强烈的要求,对于高分辨率、多帧数的视频,手机内存很容易被耗尽而导致程序崩溃。

[0006] 因此,如何在具备制作成本低、易于推广等的优势的同时,保证在大部分用户终端设备上,可以实现对更高清商品全景图的流畅展示,成为需要本领域技术人员解决的技术问题。

发明内容

[0007] 本申请提供了提供数据对象全景图信息的方法及装置,可以实现即使配置较低的终端设备,也能够支持高清全景图的流畅展示。

[0008] 本申请提供了如下方案:

[0009] 一种提供数据对象全景图信息的方法,包括:

[0010] 客户端从服务端下载获得数据对象对应的视频文件,所述视频文件用于描述所述数据对象的全景图信息;

[0011] 对所述视频文件进行解码,根据解码结果生成中间文件,并将所述中间文件保存在所述客户端所在终端设备的本地数据存储介质中;

[0012] 在根据用户的互动操作行为进行全景图展示过程中,从所述本地数据存储介质中保存的中间文件中获得待渲染数据,并进行图像的渲染及展示。

[0013] 一种提供数据对象全景图信息的方法,包括:

[0014] 服务端以视频文件的形式保存数据对象的全景图信息;

[0015] 接收到客户端的下载请求时,将所述视频文件提供给所述客户端,由所述客户端对所述视频文件进行解码,根据解码结果生成中间文件,并将所述中间文件保存在所述客户端所在终端设备的本地数据存储介质中,在根据用户的互动操作行为进行全景图展示过程中,从所述本地数据存储介质中保存的中间文件中获得待渲染数据,并进行图像的渲染及展示。

[0016] 一种提供数据对象全景图信息的装置,应用于客户端,包括:

[0017] 视频文件下载单元,用于从服务端下载获得数据对象对应的视频文件,所述视频文件用于描述所述数据对象的全景图信息;

[0018] 中间文件生成单元,用于对所述视频文件进行解码,根据解码结果生成中间文件,并将所述中间文件保存在所述客户端所在终端设备的本地数据存储介质中;

[0019] 数据读取单元,用于在根据用户的互动操作行为进行全景图展示过程中,从所述本地数据存储介质中保存的中间文件中获得待渲染数据,并进行图像的渲染及展示。

[0020] 一种提供数据对象全景图信息的装置,应用于服务端,包括:

[0021] 视频文件保存单元,用于以视频文件的形式保存数据对象的全景图信息;

[0022] 视频文件提供单元,用于接收到客户端的下载请求时,将所述视频文件提供给所述客户端,由所述客户端对所述视频文件进行解码,根据解码结果生成中间文件,并将所述中间文件保存在所述客户端所在终端设备的本地数据存储介质中,在根据用户的互动操作行为进行全景图展示过程中,从所述本地数据存储介质中保存的中间文件中获得待渲染数据,并进行图像的渲染及展示。

[0023] 一种提供数据对象全景图信息的方法,包括:

[0024] 在本地数据存储介质中保存视频文件对应的中间文件,所述视频文件用于描述数据对象的全景图信息;所述中间文件用于记录所述视频文件的解码结果信息;

[0025] 在根据用户的互动操作行为进行全景图展示过程中,从所述中间文件中获得待渲染数据,以进行图像的渲染及展示。

[0026] 一种提供数据对象全景图信息的装置,包括:

[0027] 文件保存单元,用于在本地数据存储介质中保存视频文件对应的中间文件,所述

视频文件用于描述数据对象的全景图信息;所述中间文件用于记录所述视频文件的解码结果信息;

[0028] 数据获得单元,用于在根据用户的互动操作行为进行全景图展示过程中,从所述中间文件中获得待渲染数据,以进行图像的渲染及展示。

[0029] 一种电子设备,包括:

[0030] 一个或多个处理器;以及

[0031] 与上述一个或多个处理器关联的存储器,所述存储器用于存储程序指令,所述程序指令在被所述一个或多个处理器读取执行时,执行如下操作:

[0032] 从服务端下载获得数据对象对应的视频文件,所述视频文件用于描述所述数据对象的全景图信息;

[0033] 对所述视频文件进行解码,根据解码结果生成中间文件,并将所述中间文件保存在所述客户端所在终端设备的本地数据存储介质中;

[0034] 在根据用户的互动操作行为进行全景图展示过程中,从所述本地数据存储介质中保存的中间文件中获得待渲染数据,并进行图像的渲染及展示。

[0035] 根据本申请提供的具体实施例,本申请公开了以下技术效果:

[0036] 通过本申请实施例,在服务端可以以视频文件的形式保存数据对象的全景图信息,在需要进行全景图信息展示时,客户端可以将数据对象对应的所述视频文件下载到终端设备本地,然后进行解码处理,解码后得到的数据可以以中间文件的形式保存在本地的数据存储介质中,例如,可以是存储卡等非易失性存储介质,之后,就可以根据用户的互动操作行为,从所述中间文件中读入内存,并完成具体图像的渲染及展示。通过这种方式下,可以节省服务端的存储资源,以及服务端到服务端的传输资源,在具体进行全景图展示的过程中,还可以避免对内存的大量占用,使得即使配置较低的终端设备,也能够支持高清全景图的流畅展示。

[0037] 当然,实施本申请的任一产品并不一定需要同时达到以上所述的所有优点。

附图说明

[0038] 为了更清楚地说明本申请实施例或现有技术中的技术方案,下面将对实施例中所需要使用的附图作简单地介绍,显而易见地,下面描述中的附图仅仅是本申请的一些实施例,对于本领域普通技术人员来讲,在不付出创造性劳动的前提下,还可以根据这些附图获得其他的附图。

[0039] 图1是本申请实施例提供的视频文件获得方式示意图;

[0040] 图2是本申请实施例提供的系统架构的示意图;

[0041] 图3是本申请实施例提供的第一方法的流程图;

[0042] 图4是本申请实施例提供的第二方法的流程图;

[0043] 图5是本申请实施例提供的第一装置的示意图;

[0044] 图6是本申请实施例提供的第二装置的示意图;

[0045] 图7是本申请实施例提供的第三方法的流程图;

[0046] 图8是本申请实施例提供的第三装置的流程图;

[0047] 图9是本申请实施例提供的电子设备的示意图。

具体实施方式

[0048] 下面将结合本申请实施例中的附图,对本申请实施例中的技术方案进行清楚、完整地描述,显然,所描述的实施例仅仅是本申请一部分实施例,而不是全部的实施例。基于本申请中的实施例,本领域普通技术人员所获得的所有其他实施例,都属于本申请保护的范围。

[0049] 在本申请实施例中,针对需要提供全景图的数据对象,可以在服务端可以以视频文件的格式保存全景图信息。在用户需要浏览具体数据对象的全景图时,客户端可以将视频文件下载到终端设备本地,并且,对视频文件进行逐帧的解码,解码后得到的数据,并不是直接写入内容,而是以中间文件的形式保存在终端设备本地的数据存储介质中,例如,存储卡等非易失性存储介质。在具体与用户进行互动的过程中,再以帧为单位将本地存储的解码结果读入内存,并进行渲染及图像的绘制。这样,一方面,服务端可以保存视频文件,因此,节省存储空间,下载的过程也会节省流量等网络资源,下载到本地后,还可以逐帧进行解码,也就,在本地保存的是视频解码后的数据,这样,在具体与用户互动的过程中,就不再需要进行实时的视频解码,可以缩短响应时间;另外,视频解码后的数据可以是以中间文件的形式保存在终端设备的存储介质中,而不是直接存入内存,只有在具体互动的过程中,才会将数据读入内存,而且读入的方式可以是以帧为单位进行,每完成一帧的展示需要进入到下一帧进行展示时,就可以将该帧的数据从内存清除,读入下一帧的数据,因此,也不会造成对内存的过多占用,并且,理论上讲可以实现只占用单帧数据的内存空间,对设备内存的要求降低。因此,即使是相对低端的终端设备,也能够实现对高清全景图的流畅展示。

[0050] 具体实现时,数据对象的全景视频文件,可以是由第一用户(商家用户或者卖家用户等)对数据对象对应的实物进行拍摄,或者也可以将样品提供给由网络销售系统后台工作人员,由工作人员进行拍摄,等等。具体的拍摄过程可以如图1所示,可以将被拍摄物品放置于旋转台上,在旋转台旋转的过程中,使用拍摄设备对物品进行拍摄,旋转满一周时,结束拍摄。当然,在实际应用中,还可以有其他的拍摄全景图视频的方式。拍摄完成之后,可以直接以视频文件的形式保存在服务端。之后,服务端在发布对应的数据对象信息时,可以在数据对象信息界面中提供用于查看数据对象全景图的操作选项,用户在进入到数据对象信息界面浏览数据对象信息的过程中,如果需要查看全景图,就可以通过该选项发起相应请求。之后,客户端就可以向服务端请求下载该数据对象的全景图视频文件,当然,也可以在进入数据对象信息界面时,就进行视频文件的下载。

[0051] 完成视频文件的下载后,客户端就可以执行对视频文件的解码操作,解码得到的数据可以保存成中间文件,并存储在终端设备的非易失性存储介质中。其中,具体在解码的过程中,是进行的逐帧的解码,因此,可以是每次完成对一帧的解码后,将解码得到的数据以流的形式附加到中间文件的末尾处。

[0052] 为了使得中间文件中的数据仍然能够具有帧的特性,还可以在每次向文件末尾添加数据流时,记录每一帧写入中间文件的总字节数,以及距离中间文件起始位置的偏移量。在完成所有帧的解码,及数据的写入后,就可以各帧对应的字节数以及偏移量信息写入到一个元数据文件中。这样,就可以根据元数据文件中记录的偏移量以及总字节数信息,实现以帧为单位从中间文件中进行数据的读取。

[0053] 另外,由于对视频帧进行解码得到的解码结果通常是Bitmap格式,其中包含的信

息量通常比较大,如果直接将这种数据写入到中间文件,则可能会导致中间文件过大。因此,为了节省存储空间,避免中间文件过大导致读取速度编码等问题,还可以在完成每一帧的解码后,在写入中间文件之前,对解码结果进行压缩处理,例如,可以使用JPEG等协议进行压缩,使得每一帧的解码结果可以保存为一份JPEG图片格式的数据,在互动过程中,需要对每一帧的图像进行绘制时,再对JPEG格式的数据进行解压缩,还原成Bitmap数据,然后再进行绘制处理。

[0054] 在完成视频文件的解码以及中间文件的保存后,就可以根据用户执行的互动操作,提供对全景图的展示。具体的,在初始状态下,可以在界面中展示全景图的某一帧图像作为初始画面,之后,用户可以通过转动终端设备的方式,或者,左右/上下滑动屏幕等方式来改变观察角度,以观察到数据对象的全貌。在用户操作过程中,就可以根据终端设备的旋转角度,或者用户的滑动距离等,来确定当前需要展示的目标帧。例如,可以根据已旋转的角度与可旋转的最大角度之间的比例,以及视频的帧率等信息,将用户的操作映射为视频帧的切换,例如,每旋转0.1度,切换一帧,等等。在确定出当前需要展示的目标帧之后,就可以从中间文件中取出该目标帧的数据,并写入到内存中,以便在内存中实现对该帧数据的渲染及绘制。具体实现时,如果保存了元数据文件,则可以首先查询元数据文件,确定出目标帧的数据距离中间文件起始位置的偏移量,以及占用的总字节数,这样,就可以根据该偏移量以及总字节数,从中间文件中确定出该目标帧对应的数据。

[0055] 其中,由于本申请实施例中是在全景图播放过程中,将中间文件中的数据逐帧的写入内存,然后在内存中进行渲染以及绘制等处理,因此,如果采用传统的方案,则在每次读取一帧的数据时,都需要一个字节一个字节的进行读取操作,会造成比较大的I/O次数,这对于一些相对低端的终端设备而言,也可能是不能承受,或者造成性能的下降。为此,在优选的实施方式中,还可以采用内存映射文件技术,也就是说,将硬盘等存储介质上文件的位置与进程逻辑地址空间中一块大小相同的区域之间进行一一对应,这种对应关系属于逻辑上的概念,物理上是不存在的。因此,在内存映射的过程中,并没有实际的数据拷贝,文件没有被载入内存,只是逻辑上被放入了内存,具体到代码,就是建立并初始化了相关的数据结构(struct address_space),这个过程由操作系统通过调用mmap()实现,所以建立内存映射的效率很高。当然,虽然建立内存映射没有进行实际的数据拷贝,但是进程仍然能最终直接通过内存操作访问到硬盘上的文件。具体的,mmap()会返回一个指针ptr,它指向进程逻辑地址空间中的一个地址,这样,进程无需再调用read或write对文件进行读写,而只需要通过ptr就能够操作文件。这种通过内存映射的方法访问硬盘上的文件,效率要比read和write系统调用高。

[0056] 在完成数据的读取或者内存映射后,就可以进行具体的图像绘制操作。在Android等操作系统的终端设备上,绘制图像最常见的方式是使用ImageView,借助系统每隔16ms定时刷新的机制来更新图像。但是,由于这种方式只能在主线程被动更新画面,速度较慢,在CPU比较老旧的终端设备可能上会产生明显卡顿。为了解决该问题,本申请实施例可以采用OpenGL技术在单独的渲染线程中对图像进行刷新,同时其具备的主动刷新技术也保证了刷新的即时性。其中,所述OpenGL是一个开放的三维图形软件包,它独立于窗口系统或其他操作系统,以它为基础开发的应用程序可以十分方便地在各种平台间移植,关于该技术的具体实现,这里不再详述。

[0057] 具体实现时,从系统结构而言,参见图2,可以分为服务端以及客户端两部分,例如,具体可以是网络销售系统提供的服务端及客户端,等等。其中,服务端主要用于对各个数据对象的全景图信息对应的视频文件进行保存,客户端则用于在用户需要时,将视频文件下载到终端设备本地,并执行后续的解码、渲染展示的等操作。下面分别从客户端以及服务端的角度,对具体实现方式进行详细介绍。

[0058] 实施例一

[0059] 参见图3,本申请实施例一首先从客户端的角度,提供了一种提供数据对象全景图信息的方法,具体的,该方法可以包括:

[0060] S301:客户端从服务端下载获得数据对象对应的视频文件,所述视频文件用于描述所述数据对象的全景图信息;

[0061] 具体实现时,数据对象可以是网络销售系统中发布的商品对象、店铺对象,或者,还可以是在VR(虚拟现实)等场景下的VR对象等。客户端可以为用户展示用于浏览数据对象信息的页面,例如,可以是某商品对象的详情信息页面,某店铺的首页,等等。在这种页面中可以提供用于查看数据对象全景图信息的操作选项(例如,可以是按钮等形式),如果用户需要查看数据对象的全景图,则可以点击该操作选项,发起相应的请求。在实际应用中,该下载视频文件的步骤,可以是在用户发起具体的查看全景图信息的请求之后,再从服务端进行下载,或者,也可以在加载数据对象信息页面的页面数据时,就一同将全景图信息对应的视频文件下载到本地。

[0062] S302:对所述视频文件进行解码,根据解码结果生成中间文件,并将所述中间文件保存在所述客户端所在终端设备的本地数据存储介质中;

[0063] 在将视频文件下载到终端设备本地后,就可以执行解码操作,并保存为中间文件,存储在终端设备的本地数据存储介质中,例如,通常可以是非易失性的存储器,包括手机等移动终端设备的存储卡等。具体在进行解码的过程中,可以首先创建中间文件,在初始状态下,中间文件可以是空的。之后,每完成一帧的解码,都可以根据每一帧的解码结果生成字节流,并以所述字节流的形式添加到所述中间文件的末尾,这样,随着解码的进行,中间文件中存储的内容在不断增多,直到完成最后一帧的解码操作。

[0064] 具体实现时,由于对视频文件的每一帧进行解码的结果通常是Bitmap格式的数据,这种数据的数据量往往非常大,因此,如果直接保存到中间文件中,则可能会使得最终的中间文件过大。为此,在本申请的优选实施方式中,还可以在解码完成后,首先执行数据压缩,例如,可以按照JPEG格式进行压缩,这样,每一帧的解码结果都相当于压缩为一个JPEG格式的图片数据,之后,再将这种JPEG格式的数据以流的形式添加到中间文件的末尾。

[0065] 另外,在对每一帧进行解码,并将解码结果(或者压缩后的数据)写入到中间文件的过程中,会使得整个视频文件中各个帧的解码数据都保存在同一个中间文件中。因此,为了便于区分中间文件中具体数据内容与各帧之间的对应关系,还可以创建一元数据文件,在对每一帧进行解码并写入中间文件的过程中,可以在所述元数据文件中保存每一帧对应的写入所述中间文件的总字节长度,以及相对于所述中间文件起始位置的偏移量,这样,就可以根据所述偏移量以及所述总字节长度,从所述中间文件中确定每一帧对应数据内容,从而在具体展示全景图的过程中,有利于实现以帧为单位的数据访问,避免一次性将过多的数据读入到内存中,导致对内存的占用。

[0066] S303:在根据用户的互动操作行为进行全景图展示过程中,从所述本地数据存储介质中保存的中间文件中获得待渲染数据,并进行图像的渲染及展示。

[0067] 在完成视频文件的解码,以及中间文件的保存后,就可以根据用户的互动操作行为,提供数据对象的全景图展示。其中,具体的互动操作行为可以是旋转终端设备的操作,此时,客户端可以通过终端设备中的陀螺仪等传感器来感知用户的互动操作,并且还可以获取到旋转的角度等参数信息。另外,互动操作行为也可以是在触摸屏上的滑动等操作,此时,客户端可以通过触摸屏检测用户的操作行为,并且还可以获取到滑动的距离等参数信息。

[0068] 如前文所述,具体在从中间文件中获得待渲染数据时,为了避免一次性读入内存的数据过多,可以以帧为单位从所述中间文件中获得待渲染的数据,并基于每帧对应的数据进行图像渲染及展示。也即,每次将一帧的数据读入内存,并且,每完成一帧的展示需要切换到下一帧时,就可以将已经完成展示的帧对应的数据内容从内存中清除,这样,保证内存中每次只读入一帧的数据,也即,在全景图展示过程中,始终只占用一帧数据对应的内存空间。当然,在实际应用中,还可以通过每次多读入当前待渲染帧之前或者之后一帧或若干帧的数据到内存中,从而实现预加载,提高帧切换之间的流程度。当然,具体是否需要预加载,可以根据终端设备的实际情况而定,例如,在终端设备内存等指标允许的情况下,可以采用预加载。

[0069] 在这种以帧为单位进行数据读取的情况下,还可以首先根据所述互动操作行为确定待渲染帧的位置,然后,根据所述待渲染帧的位置从所述中间文件中获得所述待渲染帧对应的数据,并提供给渲染进程进行渲染处理。其中,具体在确定待渲染帧的位置时,由于在初始状态下,可以显示全景图的某一帧作为起始位置,之后,用户可以通过旋转终端设备、滑动屏幕等方式来改变观察视角,因此,客户端可以根据用户的旋转角度、滑动距离等确定出待渲染帧的位置,所谓待渲染帧的位置具体可以包括待渲染帧的编号等标识信息。这样,由于可以通过元数据文件等方式保存每一帧的标识信息,以及对应的数据内容在中间文件中距离中间文件起始位置的偏移量,以及总占用的字节长度,因此,就可以根据该元数据信息,从中间文件中定位到待渲染帧对应的数据内容。

[0070] 具体实现时,为了避免造成过大的I/O,还可以采用内存映射文件的方式,从所述中间文件中获得待渲染帧对应的数据。另外,具体在进行图像渲染时,可以通过OpenGL技术将对应图像绘制到显示屏上。通过这种方式可以实现图像的主动更新,有利于提高图像渲染速度,从而进一步提高流畅度。

[0071] 总之,通过本申请实施例,在服务端可以以视频文件的形式保存数据对象的全景图信息,在需要进行全景图信息展示时,客户端可以将数据对象对应的所述视频文件下载到终端设备本地,然后进行解码处理,解码后得到的数据可以以中间文件的形式保存在本地的数据存储介质中,例如,可以是存储卡等非易失性存储介质,之后,就可以根据用户的互动操作行为,从所述中间文件中读入内存,并完成具体图像的渲染及展示。通过这种方式下,可以节省服务端的存储资源,以及服务端到服务端的传输资源,在具体进行全景图展示的过程中,还可以避免对内存的大量占用,使得即使配置较低的终端设备,也能够支持高清全景图的流畅展示。

[0072] 实施例二

[0073] 该实施例二是与实施例一相对应的,从服务端的角度,提供了一种提供数据对象全景图信息的方法,参见图4,该方法具体可以包括:

[0074] S401:服务端以视频文件的形式保存数据对象的全景图信息;

[0075] S402:接收到客户端的下载请求时,将所述视频文件提供给所述客户端,由所述客户端对所述视频文件进行解码,根据解码结果生成中间文件,并将所述中间文件保存在所述客户端所在终端设备的本地数据存储介质中,在根据用户的互动操作行为进行全景图展示过程中,从所述本地数据存储介质中保存的中间文件中获得待渲染数据,并进行图像的渲染及展示。

[0076] 关于该实施例二中各步骤的具体实现,可以参见本申请说明书中其他各部分内容的记载,这里不再赘述。

[0077] 与本申请实施例一相对应,本申请实施例还提供了一种提供数据对象全景图信息的装置,参见图5,该装置应用于客户端,包括:

[0078] 视频文件下载单元501,用于从服务端下载获得数据对象对应的视频文件,所述视频文件用于描述所述数据对象的全景图信息;

[0079] 中间文件生成单元502,用于对所述视频文件进行解码,根据解码结果生成中间文件,并将所述中间文件保存在所述客户端所在终端设备的本地数据存储介质中;

[0080] 数据读取单元503,用于在根据用户的互动操作行为进行全景图展示过程中,从所述本地数据存储介质中保存的中间文件中获得待渲染数据,并进行图像的渲染及展示。

[0081] 具体实现时,所述视频文件下载单元501具体可以用于:

[0082] 在接收到访问数据对象信息页面的请求时,从服务端下载所述页面的页面数据,以及所述数据对象的视频文件。

[0083] 或者,所述视频文件下载单元501也可以用于:

[0084] 在展示数据对象信息页面的过程中,如果接收到查看所述数据对象全景图信息的请求,则从服务端下载获得所述数据对象的视频文件。

[0085] 在一种具体实现方式下,所述中间文件生成单元具体可以包括:

[0086] 文件创建子单元,用于创建中间文件;

[0087] 解码子单元,用于对所述视频文件进行逐帧解码;

[0088] 添加子单元,用于根据每一帧的解码结果生成字节流,并以所述字节流的形式添加到所述中间文件的末尾。

[0089] 具体实现时,为了避免中间文件过大,所述添加子单元具体可以包括:

[0090] 压缩子单元,用于按照预置的压缩格式对每一帧的解码结果进行压缩;

[0091] 压缩后添加子单元,用于按照压缩后的数据生成待添加到所述中间文件末尾的字节流;

[0092] 所述数据读取单元在具体进行图像渲染及绘制之前,还可以用于:

[0093] 对所述中间文件中的数据进行解压缩处理。

[0094] 另外,为了便于对中间文件中具体各帧对应的数据内容进行定位,该装置还可以包括:

[0095] 元数据文件创建单元,用于创建元数据文件;

[0096] 元数据保存单元,用于在所述元数据文件中保存每一帧对应的写入所述中间文件

的总字节长度,以及相对于所述中间文件起始位置的偏移量,以用于根据所述偏移量以及所述总字节长度,从所述中间文件中确定每一帧对应数据内容。

[0097] 具体实现时,数据读取单元具体可以用于:

[0098] 以帧为单位从所述中间文件中获得待渲染的数据,并基于每帧对应的数据进行图像渲染及展示。

[0099] 具体的,所述数据读取单元可以包括:

[0100] 帧位置确定子单元,用于根据所述互动操作行为确定待渲染帧的位置;

[0101] 帧数据获得子单元,用于根据所述待渲染帧的位置从所述中间文件中获得所述待渲染帧对应的数据,并提供给渲染进程进行渲染处理。

[0102] 为了避免造成大量的I/O,数据读取单元具体可以用于:

[0103] 采用内存映射文件的方式,从所述中间文件中获得待渲染帧对应的数据。

[0104] 另外,该装置还可以包括:

[0105] 内存数据清除单元,用于在切换到下一帧进行渲染时,将内存中当前帧对应的数据清除。

[0106] 另外,所述数据读取单元具体在进行图像渲染及展示时,具体可以用于:

[0107] 通过OpenGL技术将对应图像绘制到显示屏上。

[0108] 与实施例二相对应,本申请实施例还提供了一种提供数据对象全景图信息的装置,参见图6,该装置应用于服务端,包括:

[0109] 视频文件保存单元601,用于以视频文件的形式保存数据对象的全景图信息;

[0110] 视频文件提供单元602,用于接收到客户端的下载请求时,将所述视频文件提供给所述客户端,由所述客户端对所述视频文件进行解码,根据解码结果生成中间文件,并将所述中间文件保存在所述客户端所在终端设备的本地数据存储介质中,在根据用户的互动操作行为进行全景图展示过程中,从所述本地数据存储介质中保存的中间文件中获得待渲染数据,并进行图像的渲染及展示。

[0111] 实施例三

[0112] 在前述实施例中,是在服务端保存数据对象对应的视频文件,然后,在客户端进行视频解码,并根据解码结果生成为中间文件,保存在客户端所在终端设备的本地数据存储介质中。这样,在具体与用户进行互动展示的过程中,就可以从本地数据存储介质中读取解码后的数据作为待渲染数据,并直接进行渲染展示。而在其他实施方式中,也可以直接在服务端保存视频文件的解码结果,例如,可以由服务端对数据对象对应的视频文件进行逐帧解码,之后还可以对每帧的解码结果(Bitmap格式)按照JPEG等图片格式进行压缩,从而使每一帧的解码结果对应于一张图片,多帧解码结果则可以生成图片序列。这样,可以直接在服务端对图片序列进行保存,使得客户端在需要进行3D展示时,可以从服务端下载获得这种图片序列,然后直接保存在本地数据存储介质中,后续的渲染过程则可以与前述实施例一中的方式相同。为此,本申请实施例三还提供了另一种提供数据对象全景图信息的方法,具体的,参见图7,该方法可以包括:

[0113] S701:在本地数据存储介质中保存视频文件对应的中间文件,所述视频文件用于描述数据对象的全景图信息;所述中间文件用于记录所述视频文件的解码结果信息;

[0114] 该步骤的执行主体可以是客户端,而中间文件的生成操作可以由客户端完成,或

者,也可以由服务端来完成。也就是说,无论是在客户端对视频文件进行解码,还是在服务端进行解码,客户端都可以将根据解码结果生成的中间文件保存在本地数据存储介质中。

[0115] S702:在根据用户的互动操作行为进行全景图展示过程中,从所述中间文件中获得待渲染数据,以进行图像的渲染及展示。

[0116] 该步骤S702可以与前述实施例一中的步骤S303相同,参照执行即可。另外,关于该实施例三中其他的具体实现,包括中间文件的具体生成方式等,可以参照前述实施例一以及实施例二中的记载,这里不再赘述。

[0117] 与实施例三相对应,本申请实施例还提供了一种提供数据对象全景图信息的装置,参见图8,该装置可以包括:

[0118] 文件保存单元801,用于在本地数据存储介质中保存视频文件对应的中间文件,所述视频文件用于描述数据对象的全景图信息;所述中间文件用于记录所述视频文件的解码结果信息;

[0119] 数据获得单元802,用于在根据用户的互动操作行为进行全景图展示过程中,从所述中间文件中获得待渲染数据,以进行图像的渲染及展示。

[0120] 与实施例一相对应,本申请实施例还提供了一种电子设备,该电子设备可以包括:

[0121] 一个或多个处理器;以及

[0122] 与所述一个或多个处理器关联的存储器,所述存储器用于存储程序指令,所述程序指令在被所述一个或多个处理器读取执行时,执行如下操作:

[0123] 从服务端下载获得数据对象对应的视频文件,所述视频文件用于描述所述数据对象的全景图信息;

[0124] 对所述视频文件进行解码,根据解码结果生成中间文件,并将所述中间文件保存在所述客户端所在终端设备的本地数据存储介质中;

[0125] 在根据用户的互动操作行为进行全景图展示过程中,从所述本地数据存储介质中保存的中间文件中获得待渲染数据,并进行图像的渲染及展示。

[0126] 其中,图9示例性的展示出了电子设备的架构,例如,设备900可以是移动电话,计算机,数字广播终端,消息收发设备,游戏控制台,平板设备,医疗设备,健身设备,个人数字助理,飞行器等。

[0127] 参照图9,设备900可以包括以下一个或多个组件:处理组件902,存储器904,电源组件906,多媒体组件908,音频组件910,输入/输出(I/O)的接口912,传感器组件914,以及通信组件916。

[0128] 处理组件902通常控制设备900的整体操作,诸如与显示,电话呼叫,数据通信,相机操作和记录操作相关联的操作。处理元件902可以包括一个或多个处理器920来执行指令,以完成本公开技术方案提供的视频播放方法中的当满足预设条件时,生成流量压缩请求,并发送给服务器,其中所述流量压缩请求中记录有用于触发服务器获取目标关注区域的信息,所述流量压缩请求用于请求服务器优先保证目标关注区域内视频内容的码率;根据服务器返回的码流文件播放所述码流文件对应的视频内容,其中所述码流文件为服务器根据所述流量压缩请求对所述目标关注区域之外的视频内容进行码率压缩处理得到的视频文件的全部或部分步骤。此外,处理组件902可以包括一个或多个模块,便于处理组件902和其他组件之间的交互。例如,处理部件902可以包括多媒体模块,以方便多媒体组件908和

处理组件902之间的交互。

[0129] 存储器904被配置为存储各种类型的数据以支持在设备900的操作。这些数据的示例包括用于在设备900上操作的任何应用程序或方法的指令,联系人数据,电话簿数据,消息,图片,视频等。存储器904可以由任何类型的易失性或非易失性存储设备或者它们的组合实现,如静态随机存取存储器 (SRAM),电可擦除可编程只读存储器 (EEPROM),可擦除可编程只读存储器 (EPROM),可编程只读存储器 (PROM),只读存储器 (ROM),磁存储器,快闪存储器,磁盘或光盘。

[0130] 电源组件906为设备900的各种组件提供电力。电源组件906可以包括电源管理系统,一个或多个电源,及其他与为设备900生成、管理和分配电力相关联的组件。

[0131] 多媒体组件908包括在设备900和用户之间的提供一个输出接口的屏幕。在一些实施例中,屏幕可以包括液晶显示器 (LCD) 和触摸面板 (TP)。如果屏幕包括触摸面板,屏幕可以被实现为触摸屏,以接收来自用户的输入信号。触摸面板包括一个或多个触摸传感器以感测触摸、滑动和触摸面板上的手势。触摸传感器可以不仅感测触摸或滑动动作的边界,而且还检测与触摸或滑动操作相关的持续时间和压力。在一些实施例中,多媒体组件908包括一个前置摄像头和/或后置摄像头。当设备900处于操作模式,如拍摄模式或视频模式时,前置摄像头和/或后置摄像头可以接收外部的多媒体数据。每个前置摄像头和后置摄像头可以是一个固定的光学透镜系统或具有焦距和光学变焦能力。

[0132] 音频组件910被配置为输出和/或输入音频信号。例如,音频组件910包括一个麦克风 (MIC),当设备900处于操作模式,如呼叫模式、记录模式和语音识别模式时,麦克风被配置为接收外部音频信号。所接收的音频信号可以被进一步存储在存储器904或经由通信组件916发送。在一些实施例中,音频组件910还包括一个扬声器,用于输出音频信号。

[0133] I/O接口912为处理组件902和外围接口模块之间提供接口,上述外围接口模块可以是键盘,点击轮,按钮等。这些按钮可包括但不限于:主页按钮、音量按钮、启动按钮和锁定按钮。

[0134] 传感器组件914包括一个或多个传感器,用于为设备900提供各个方面的状态评估。例如,传感器组件914可以检测到设备900的打开/关闭状态,组件的相对定位,例如所述组件为设备900的显示器和小键盘,传感器组件914还可以检测设备900或设备900一个组件的位置改变,用户与设备900接触的存在或不存在,设备900方位或加速/减速和设备900的温度变化。传感器组件914可以包括接近传感器,被配置用来在没有任何的物理接触时检测附近物体的存在。传感器组件914还可以包括光传感器,如CMOS或CCD图像传感器,用于在成像应用中使用。在一些实施例中,该传感器组件914还可以包括加速度传感器,陀螺仪传感器,磁传感器,压力传感器或温度传感器。

[0135] 通信组件916被配置为便于设备900和其他设备之间有线或无线方式的通信。设备900可以接入基于通信标准的无线网络,如WiFi,2G或3G,或它们的组合。在一个示例性实施例中,通信部件916经由广播信道接收来自外部广播管理系统的广播信号或广播相关信息。在一个示例性实施例中,所述通信部件916还包括近场通信 (NFC) 模块,以促进短程通信。例如,在NFC模块可基于射频识别 (RFID) 技术,红外数据协会 (IrDA) 技术,超宽带 (UWB) 技术,蓝牙 (BT) 技术和其他技术来实现。

[0136] 在示例性实施例中,设备900可以被一个或多个应用专用集成电路 (ASIC)、数字信

号处理器 (DSP)、数字信号处理设备 (DSPD)、可编程逻辑器件 (PLD)、现场可编程门阵列 (FPGA)、控制器、微控制器、微处理器或其他电子元件实现,用于执行上述方法。

[0137] 在示例性实施例中,还提供了一种包括指令的非临时性计算机可读存储介质,例如包括指令的存储器904,上述指令可由设备900的处理器920执行以完成本公开技术方案提供的视频播放方法中的当满足预设条件时,生成流量压缩请求,并发送给服务器,其中所述流量压缩请求中记录有用于触发服务器获取目标关注区域的信息,所述流量压缩请求用于请求服务器优先保证目标关注区域内视频内容的码率;根据服务器返回的码流文件播放所述码流文件对应的视频内容,其中所述码流文件为服务器根据所述流量压缩请求对所述目标关注区域之外的视频内容进行码率压缩处理得到的视频文件。例如,所述非临时性计算机可读存储介质可以是ROM、随机存取存储器 (RAM)、CD-ROM、磁带、软盘和光数据存储设备。

[0138] 本说明书中的各个实施例均采用递进的方式描述,各个实施例之间相同相似的部分互相参见即可,每个实施例重点说明的都是与其他实施例的不同之处。尤其,对于系统或系统实施例而言,由于其基本相似于方法实施例,所以描述得比较简单,相关之处参见方法实施例的部分说明即可。以上所描述的系统及系统实施例仅仅是示意性的,其中所述作为分离部件说明的单元可以是或者也可以不是物理上分开的,作为单元显示的部件可以是或者也可以不是物理单元,即可以位于一个地方,或者也可以分布到多个网络单元上。可以根据实际的需要选择其中的部分或者全部模块来实现本实施例方案的目的。本领域普通技术人员在不付出创造性劳动的情况下,即可以理解并实施。

[0139] 以上对本申请所提供的提供数据对象全景图信息的方法及装置,进行了详细介绍,本文中应用了具体个例对本申请的原理及实施方式进行了阐述,以上实施例的说明只是用于帮助理解本申请的方法及其核心思想;同时,对于本领域的一般技术人员,依据本申请的思想,在具体实施方式及应用范围上均会有改变之处。综上所述,本说明书内容不应理解为对本申请的限制。

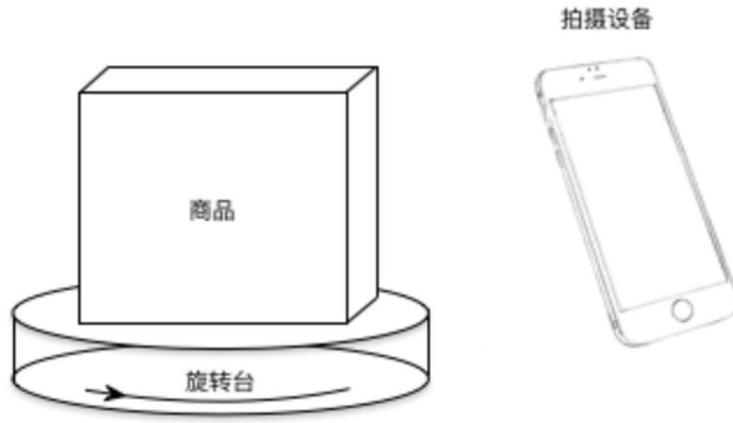


图1

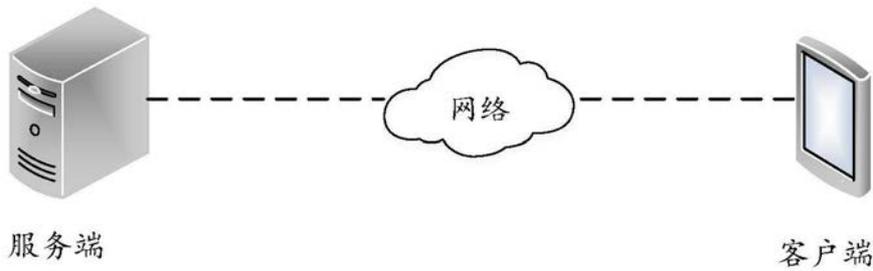


图2

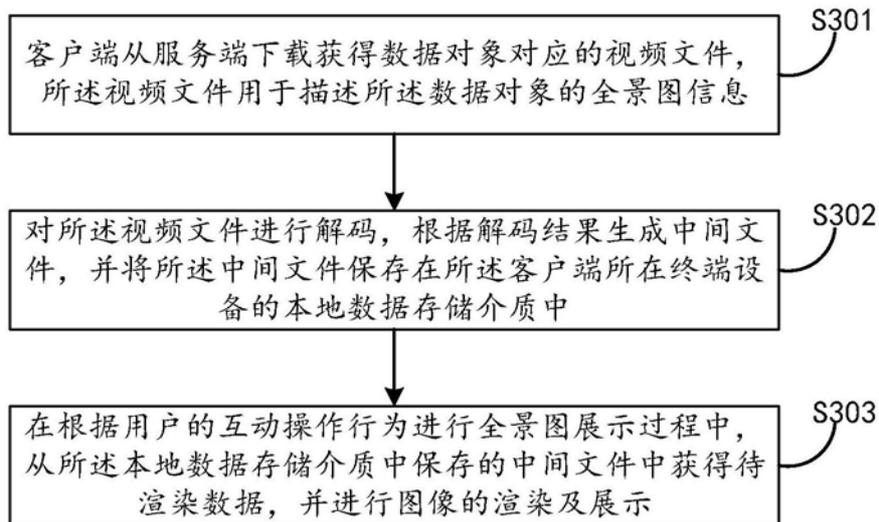


图3

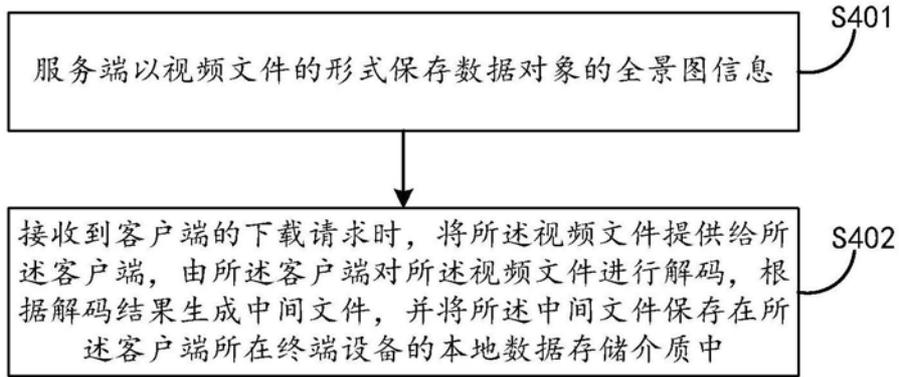


图4

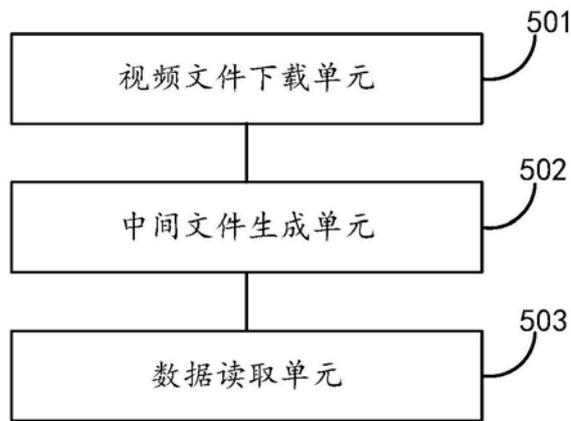


图5



图6

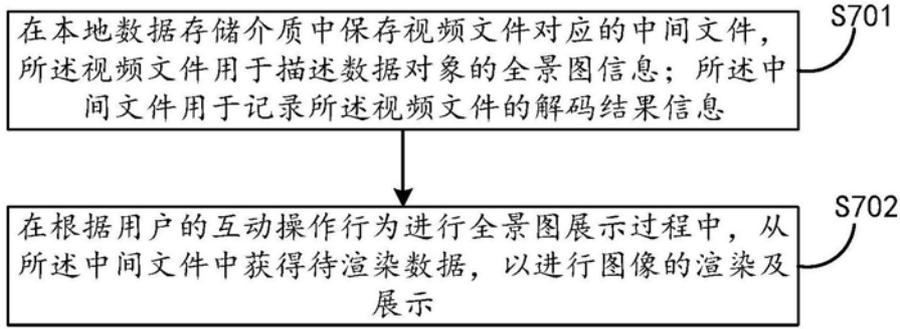


图7



图8

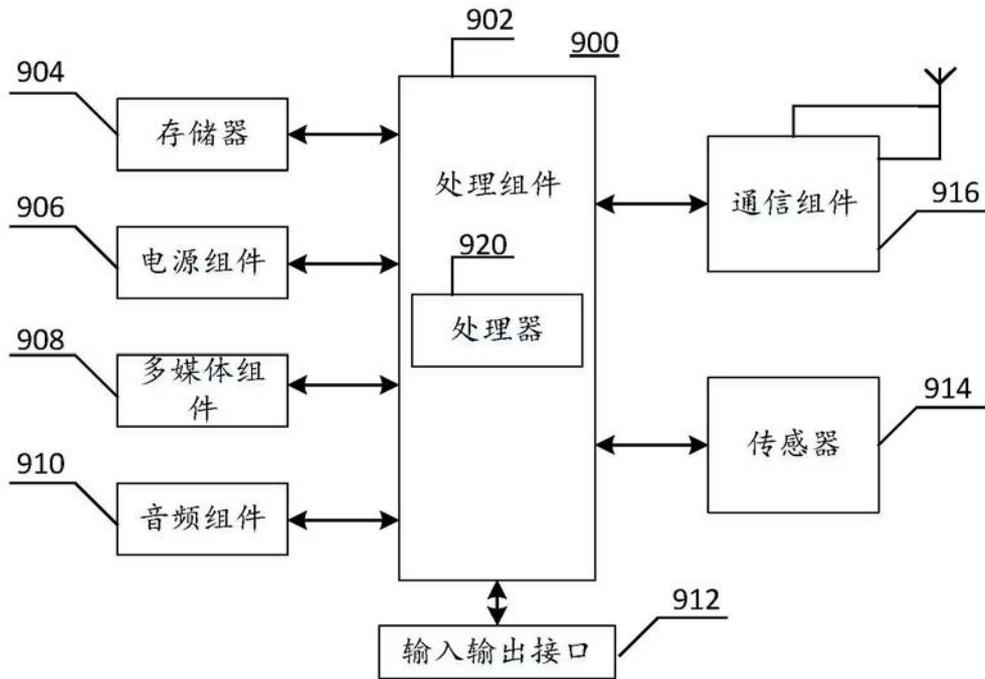


图9