



(12) 发明专利申请

(10) 申请公布号 CN 114867537 A

(43) 申请公布日 2022. 08. 05

(21) 申请号 202080090047.5

(22) 申请日 2020.11.12

(30) 优先权数据

16/688,369 2019.11.19 US

(85) PCT国际申请进入国家阶段日

2022.06.23

(86) PCT国际申请的申请数据

PCT/US2020/060206 2020.11.12

(87) PCT国际申请的公布数据

W02021/101785 EN 2021.05.27

(71) 申请人 索尼互动娱乐股份有限公司

地址 日本东京都

(72) 发明人 R.R.科伦布兰德

(74) 专利代理机构 北京市柳沈律师事务所

11105

专利代理师 张晓明

(51) Int.Cl.

A63F 13/355 (2014.01)

A63F 13/358 (2014.01)

H04N 21/2343 (2011.01)

H04N 21/24 (2011.01)

H04N 21/478 (2011.01)

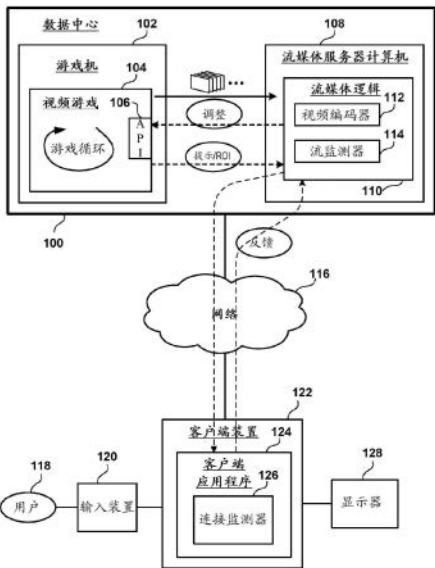
权利要求书2页 说明书19页 附图10页

(54) 发明名称

用于云游戏的自适应图形

(57) 摘要

提供了一种方法,所述方法包括以下操作:由云游戏机执行视频游戏,所述视频游戏的所述执行包括渲染玩游戏视频;由流媒体服务器处理所述玩游戏视频以通过网络流式传输到客户端装置;监测所述流媒体服务器与所述客户端装置之间通过所述网络的连接质量;响应于检测到所述流媒体服务器与所述客户端装置之间的所述连接质量的改变,然后通过所述云游戏机调整对所述玩游戏视频的所述渲染。



1. 一种方法,所述方法包括:
由云游戏机执行视频游戏,所述视频游戏的所述执行包括渲染玩游戏视频;
由流媒体服务器处理所述玩游戏视频以通过网络流式传输到客户端装置;
监测所述流媒体服务器与所述客户端装置之间通过所述网络的连接质量;
响应于检测到所述流媒体服务器与所述客户端装置之间的所述连接质量的改变,然后通过所述云游戏机调整对所述玩游戏视频的所述渲染。
2. 根据权利要求1所述的方法,其中由所述流媒体服务器处理所述玩游戏视频包括以压缩视频格式对所述玩游戏视频进行编码。
3. 根据权利要求1所述的方法,其中监测连接质量包括监测带宽或延迟中的一者或多者。
4. 根据权利要求1所述的方法,其中监测连接质量包括监测在所述玩游戏视频的所述流式传输期间发生的数据包丢失。
5. 根据权利要求1所述的方法,其中通过所述云游戏机调整对所述玩游戏视频的所述渲染包括调整所述玩游戏视频的帧速率或分辨率。
6. 根据权利要求1所述的方法,其中通过所述云游戏机调整对所述玩游戏视频的所述渲染包括调整所述玩游戏视频的所述渲染的细节水平。
7. 根据权利要求1所述的方法,其中通过所述云游戏机调整对所述玩游戏视频的所述渲染包括调整包括在所述玩游戏视频中的内容量。
8. 根据权利要求1所述的方法,
其中所述连接质量的所述改变由所述连接质量的劣化来限定;
其中通过所述云游戏机调整对所述玩游戏视频的所述渲染被配置为降低所述玩游戏视频通过所述网络的所述流式传输的位速率。
9. 根据权利要求1所述的方法,
其中所述连接质量的所述改变由所述连接质量的提高来限定;
其中通过所述云游戏机调整对所述玩游戏视频的所述渲染被配置为提高所述玩游戏视频通过所述网络的所述流式传输的位速率。
10. 根据权利要求1所述的方法,其中检测到所述连接质量的所述改变会触发消息从所述流媒体服务器到所述云游戏机的传输,并且其中通过所述云游戏机对所述玩游戏视频的所述渲染的所述调整是响应于所述消息。
11. 一种方法,所述方法包括:
由云游戏机执行视频游戏,所述视频游戏的所述执行包括渲染玩游戏视频;
由流媒体服务器处理所述玩游戏视频以通过网络流式传输到客户端装置,其中由所述流媒体服务器处理所述玩游戏视频包括以压缩视频格式对所述玩游戏视频进行编码;
监测所述流媒体服务器与所述客户端装置之间通过所述网络的连接质量,其中监测连接质量包括监测在所述玩游戏视频的所述流式传输期间发生的数据包丢失;
响应于检测到所述流媒体服务器与所述客户端装置之间的所述连接质量的改变,然后通过所述云游戏机调整对所述玩游戏视频的所述渲染,其中通过所述云游戏机调整对所述玩游戏视频的所述渲染包括调整所述玩游戏视频的帧速率或分辨率。
12. 根据权利要求11所述的方法,其中通过所述云游戏机调整对所述玩游戏视频的所

述渲染包括调整所述玩游戏视频的所述渲染的细节水平。

13. 根据权利要求11所述的方法，

其中所述连接质量的所述改变由所述连接质量的劣化来限定；

其中通过所述云游戏机调整对所述玩游戏视频的所述渲染被配置为降低所述玩游戏视频通过所述网络的所述流式传输的位速率。

14. 根据权利要求11所述的方法，

其中所述连接质量的所述改变由所述连接质量的提高来限定；

其中通过所述云游戏机调整对所述玩游戏视频的所述渲染被配置为提高所述玩游戏视频通过所述网络的所述流式传输的位速率。

15. 根据权利要求11所述的方法，其中检测到所述连接质量的所述改变会触发消息从所述流媒体服务器到所述云游戏机的传输，并且其中通过所述云游戏机对所述玩游戏视频的所述渲染的所述调整是响应于所述消息。

16. 一种系统，所述系统包括：

云游戏机，所述云游戏机执行视频游戏，所述视频游戏的所述执行包括渲染玩游戏视频；

流媒体服务器，所述流媒体服务器处理所述玩游戏视频以通过网络流式传输到客户端装置，其中所述流媒体服务器监测所述流媒体服务器与所述客户端装置之间通过所述网络的连接质量；

其中响应于检测到所述流媒体服务器与所述客户端装置之间的所述连接质量的改变，然后通过所述云游戏机调整对所述玩游戏视频的所述渲染。

17. 根据权利要求16所述的系统，其中由所述流媒体服务器处理所述玩游戏视频包括以压缩视频格式对所述玩游戏视频进行编码。

18. 根据权利要求16所述的系统，其中监测连接质量包括监测带宽或延迟中的一者或多者。

19. 根据权利要求16所述的系统，其中监测连接质量包括监测在所述玩游戏视频的所述流式传输期间发生的数据包丢失。

20. 根据权利要求16所述的系统，其中通过所述云游戏机调整对所述玩游戏视频的所述渲染包括调整所述玩游戏视频的帧速率或分辨率。

用于云游戏的自适应图形

技术领域

[0001] 本公开涉及用于云游戏的自适应图形的系统和方法。

背景技术

[0002] 相关技术的描述

[0003] 游戏行业当前的趋势是向云游戏迈进。云游戏通过实现在可以保证视频游戏的资源的数据中心中远程地执行视频游戏,而为最终用户提供了优势。远程执行的视频游戏所生成的视频被流式传输到用户的装备,并且来自用户的输入被发送回数据中心。这使最终用户无需拥有特定硬件即可执行游戏本身。相反,最终用户仅需要拥有足够的硬件即可流式传输玩游戏过程,并且仍然可以享受高质量的游戏体验。此外,理论上,云游戏使得能够在任何有网络连接的地方进行游戏。

[0004] 视频游戏行业的一个持续趋势是图形复杂性和计算资源可用性的增加,以满足现代游戏引擎的需求。随着视频游戏的发展,它们的分辨率和帧速率不断提高,从而使得能够渲染非常逼真且详细的虚拟环境。此外,云游戏的受欢迎程度持续增长,并且向云执行视频游戏的转变使得能够有更大机会获得高质量的游戏体验。

[0005] 从广义上讲,当今的云游戏架构依赖于现有控制台或PC游戏名称的流式传输。使用现有的未经修改的游戏,并且在没有它们的“知识”的情况下,它们的视频和音频被捕获并流式传输给用户。在此类流式传输期间,会注意基于用户的互联网连接为用户提供最佳体验。如果连接不佳,可以通过更改视频编码器设置(诸如视频分辨率、帧速率或压缩设置)来降低图像质量。然而,此类调整是在视频游戏已经输出的视频/音频上进行的。

[0006] 正是在这种背景下提出本公开的实施方案。

发明内容

[0007] 本公开的实现方式提供了用于云游戏的自适应图形的系统和方法。虽然当前的云游戏系统通过更改视频编码器设置来解决不同的互联网状况,但此类更改仅影响视频游戏已经生成的现有视频/音频,而不涉及视频游戏自身。在将视频游戏涉及到质量调整的情况下,可以实现更好的图像质量和提高的用户体验。例如,视频游戏可以渲染更容易压缩的低分辨率纹理或根本不渲染不必要的对象。本公开的实现方式在视频游戏与视频流解决方案之间提供了更好的配合,以便为用户获得更高质量的流。

[0008] 在一些实现方式中,提供了一种方法,该方法包括以下操作:由云游戏机执行视频游戏,视频游戏的执行包括渲染玩游戏视频;由流媒体服务器处理玩游戏视频以通过网络流式传输到客户端装置;监测流媒体服务器与客户端装置之间通过网络的连接质量;响应于检测到流媒体服务器与客户端装置之间的连接质量的改变,然后通过云游戏机调整对玩游戏视频的渲染。

[0009] 在一些实现方式中,由流媒体服务器处理玩游戏视频包括以压缩视频格式对玩游戏视频进行编码。

[0010] 在一些实现方式中,监测连接质量包括监测带宽或延迟中的一者或多者。

[0011] 在一些实现方式中,监测连接质量包括监测在玩游戏视频的流式传输期间发生的数据包丢失。

[0012] 在一些实现方式中,通过云游戏机调整对玩游戏视频的渲染包括调整玩游戏视频的帧速率或分辨率。

[0013] 在一些实现方式中,通过云游戏机调整对玩游戏视频的渲染包括调整玩游戏视频的渲染的细节水平。

[0014] 在一些实现方式中,通过云游戏机调整对玩游戏视频的渲染包括调整包括在玩游戏视频中的内容量。

[0015] 在一些实现方式中,连接质量的改变由连接质量的劣化来限定;其中通过云游戏机调整对玩游戏视频的渲染被配置为降低玩游戏视频通过网络的流式传输的位速率。

[0016] 在一些实现方式中,连接质量的改变由连接质量的提升来限定;其中通过云游戏机调整对玩游戏视频的渲染被配置为提高玩游戏视频通过网络的流式传输的位速率。

[0017] 在一些实现方式中,检测到连接质量的改变会触发消息从流媒体服务器到云游戏机的传输,并且其中通过云游戏机对玩游戏视频的渲染的调整是响应于所述消息。

[0018] 在一些实现方式中,提供了一种方法,该方法包括以下方法操作:由云游戏机执行视频游戏,视频游戏的执行包括渲染玩游戏视频;由流媒体服务器处理玩游戏视频以通过网络流式传输到客户端装置,其中由流媒体服务器处理玩游戏视频包括以压缩视频格式编码玩游戏视频;监测流媒体服务器与客户端装置之间通过网络的连接质量,其中监测连接质量包括监测玩游戏视频的流式传输期间发生的数据包丢失;响应于检测到流媒体服务器与客户端装置之间的连接质量的改变,然后通过云游戏机调整对玩游戏视频的渲染,其中通过云游戏机调整对玩游戏视频的渲染包括调整玩游戏视频的帧速率或分辨率。

[0019] 在一些实现方式中,提供了一种系统,该系统包括:云游戏机,该云游戏机执行视频游戏,视频游戏的执行包括渲染玩游戏视频;流媒体服务器,该流媒体服务器处理玩游戏视频以通过网络流式传输到客户端装置,其中流媒体服务器监测该流媒体服务器与客户端装置之间通过网络的连接质量;其中响应于检测到流媒体服务器与客户端装置之间的连接质量的改变,然后通过云游戏机调整对玩游戏视频的渲染。

[0020] 通过结合附图以示例方式示出本公开原理进行的以下详细描述,本公开的其他方面和优点将变得显而易见。

附图说明

[0021] 通过参考结合附图进行的以下描述可以最好地理解本公开及其另外的优点。

[0022] 图1概念性地示出了根据本公开的实现方式的云游戏系统,该云游戏系统被配置为基于网络状况使视频游戏自身涉及视频/音频质量调整。

[0023] 图2概念性地示出了根据本公开的实现方式的游戏流质量反馈循环,其中基于流质量调整视频/音频质量。

[0024] 图3概念性地示出了根据本公开的实现方式的由正在执行的视频游戏本机执行的帧速率相对于在由视频游戏渲染之后由流媒体服务器执行的帧速率的降低。

[0025] 图4概念性地示出了根据本公开的实现方式的用于控制或设置视频游戏的帧速率

的方法。

[0026] 图5概念性地示出了根据本公开的实现方式的虚拟环境中的内容的可变渲染。

[0027] 图6概念性地示出了根据本公开的实现方式的基于网络带宽反馈的若干内容设置。

[0028] 图7概念性地示出了根据本公开的实现方式的从视频游戏到流媒体逻辑的前馈优化。

[0029] 图8A示出了根据本公开的实现方式的用于加载经由云游戏站点可获得的的游戏的游戏文件的示例性系统。

[0030] 图8B是概念性地示出根据本公开的实现方式的经执行用于将云视频游戏流式传输到客户端装置的各种操作的流程图。

[0031] 图9示出了根据本公开的实现方式的信息服务提供商架构的实施方案。

具体实施方式

[0032] 在以下描述中,陈述了许多具体的细节以提供对本公开的深入理解。然而,对于本领域的技术人员将显而易见的是,可以在没有一些或所有这些具体细节的情况下实践本公开。在其他情况下,未详细描述众所周知的过程步骤,以免模糊本公开。

[0033] 本公开的实现方式通过将视频游戏包括到质量调整反馈循环中来提高云游戏会话的音频/视频质量。通常,基于互联网连接在带宽、延迟和数据包丢失方面的波动来调整质量。在现有的云游戏系统中,游戏通常不涉及这些调整。然而,通过使视频游戏涉及到调整视频/音频的质量,可以做出更明智的决策来提高云游戏会话的质量和响应性。

[0034] 从广义上讲,客户端与服务器之间的互联网连接在可用带宽、延迟和数据包丢失方面会有所波动。在当前的云游戏系统中,在会话期间监测连接并基于此调整音频/视频质量。例如,当用户的带宽变得低于为音频和视频编码器服务器侧配置的带宽时,就需要进行调整。如果不进行调整,用户将无法接收所有数据,因为他们的连接无法处理这些数据,这将导致数据包丢失和体验下降。另外,如果可用带宽增加,则可以通过增加编码器位速率来提高质量。

[0035] 目前,质量调整是通过客户端与服务器之间的反馈循环来进行的。有一些设置可以调整,诸如音频或视频编码器的位速率、帧速率和视频分辨率。然而,这些旋钮是次优的,因为流媒体软件接收到要进行编码的完成的音频或视频帧。虽然服务器可以调整分辨率,但其通过缩放已经渲染的图像来做到这一点;并且虽然其可以调整帧速率,但其是通过丢弃视频帧来实现这一点的。这些调整并不理想,因为它们不会影响视频和音频的原始本机渲染,而只会进一步处理已经渲染的视频和音频。

[0036] 但是,通过使视频游戏自身涉及到质量调整,可以获得更好的结果。因此,本公开的实现方式将流媒体质量反馈循环扩展到涉及视频游戏自身。现有的反馈循环可以通过从流媒体软件到视频游戏的通道进行扩展。该通道可用于触发质量调整,也可用于其他目的,例如暂停游戏或游戏提供流媒体软件提示(例如,在场景改变时,这是视频编码器通常难以处理的)。

[0037] 图1概念性地示出了根据本公开的实现方式的云游戏系统,该云游戏系统被配置为基于网络状况使视频游戏自身涉及视频/音频质量调整。如图所示,云游戏基础设施通常

位于数据中心100中,并且由执行视频游戏104的游戏机102和实现流媒体逻辑/流媒体服务器110的流媒体服务器计算机108组成。游戏机102是具有足够硬件来执行视频游戏104的装置。作为示例而非限制,游戏机可以是游戏控制台或其硬件等同物(例如刀片式版本)、一个或多个服务器计算机、或能够为视频游戏104提供适当执行环境的任何其他装置。在一些实现方式中,游戏机实现在其上执行视频游戏的虚拟机(例如使用管理程序)、云计算、或其他虚拟化资源(例如包括操作系统)。

[0038] 广义上讲,正在执行的视频游戏104例如以本机未压缩格式生成反映视频游戏的玩游戏过程的视频和音频(例如,视频游戏的虚拟环境的渲染,诸如虚拟角色的透视图,以及伴随的声音效果和原声带)。流媒体逻辑/流媒体服务器110在通过网络116将视频和音频流式传输到客户端装置122之前以压缩格式对视频和音频进行编码。为此,流媒体逻辑110包括视频编码器112,该视频编码器能够例如使用视频编解码器或根据视频压缩协议以压缩视频格式对视频进行编码。更具体地,流媒体逻辑可以基于网络状况或来自客户端的反馈来调整视频或音频的编码(例如,调整位速率、帧速率、分辨率等)。在一些实现方式中,视频编码器112还包括音频编码器,该音频编码器例如使用音频编解码器或根据音频压缩协议以压缩音频格式编码音频。在一些实现方式中,音频编码器与视频编码器112分离。

[0039] 如所指出,经编码的视频/音频通过网络116流式传输到客户端装置122。作为示例而非限制,客户端装置122可以是游戏控制台、个人计算机、膝上型计算机、平板计算机、移动电话,或如本公开中的能够从云游戏提供商流式传输视频游戏的玩游戏过程的任何其他装置。客户端装置122执行客户端应用程序124,该客户端应用程序接收经编码的视频/音频并解码视频以渲染到用户118观看的显示器128,并且还解码音频以通过一个或多个扬声器播放以被用户118听到。在各种实现方式中,显示器128和/或扬声器可以与客户端装置集成或与客户端装置122分离。在一些实现方式中,显示器128是电视、监测器、屏幕、投影仪或用户118可以通过其观看视频的其他装置。另外,用户118可以操作输入装置120以向视频游戏104提供输入。作为示例而非限制,输入装置120可以是游戏控制器、运动控制器、键盘、鼠标、触摸板、轨迹球、相机、深度相机、传声器或通过其可以提供针对视频游戏的交互式输入的任何其他感测装置。

[0040] 如图所示,流媒体逻辑110包括或调用流监测器114,该流监测器被配置为监测到客户端装置122的流媒体质量。作为示例而非限制,流监测器114可以监测数据包丢失或重传、来自客户端装置的肯定确认(例如,对于客户端装置所接收到的数据/数据包)、来自客户端装置的否定确认(例如,对于客户端装置未接收的数据/数据包)、网络延迟、到客户端装置122的ping时间、网络带宽、或流媒体逻辑110与客户端装置122之间的数据传输质量的任何其他指示符。

[0041] 另外,在一些实现方式中,客户端应用程序124可以包括或调用连接监测器126。连接监测器126可以被配置为监测与上述流监测器114类似但来自客户端侧的网络状况,诸如数据包/数据丢失、网络带宽、网络延迟等。在一些实现方式中,客户端应用程序114将关于此类网络状况的反馈发送到流媒体逻辑110。

[0042] 响应于诸如网络延迟或网络带宽状况等感测到的网络状况,流媒体逻辑110被配置为将流媒体质量反馈循环向后扩展以包括视频游戏本身。即,流媒体逻辑110被配置为与视频游戏104通信以调整视频游戏的执行的一些方面,诸如其视频/音频渲染或玩游戏过

程,从而补偿变化的网络状况。例如,这可以包括以降低游戏流的带宽需求的方式调整视频游戏的视频/音频渲染。在一些实现方式中,除了对视频游戏114进行的那些编码的调整之外,流媒体逻辑110还可以执行对视频编码的调整。另外,视频游戏104可以向流媒体逻辑110传送并且/或者经由流媒体逻辑110向客户端装置122传送与视频/音频编码或解码相关的信息,诸如关于视频中感兴趣区域的信息或关于视频中即将发生的事件(诸如场景改变)的提示。另外,该传送可以建立从视频游戏104经由流媒体逻辑110到客户端应用程序114的路径,以用于其他目的,诸如发送用于客户端侧渲染或界面操作的信息。

[0043] 为了促进视频游戏104与流媒体逻辑110之间的通信,视频游戏104可以(作为示例而非限制)显示API 106或共享存储器或套接字。在一些实现方式中,流媒体逻辑110访问视频游戏104的API 106以便传输请求调整视频游戏104所输出的视频/音频的消息。一般来讲,视频游戏104执行游戏循环,其通常涉及处理输入(例如,用户交互输入,诸如控制器/动作/语音输入、网络请求、或来自多玩家游戏中的其他玩家的输入等)、更新游戏状态以及使用当前游戏状态来渲染视频/音频。因此,在各种实现方式中,可以通过API 106或套接字接收的来自流媒体逻辑110的消息可以构成游戏的游戏循环的另一输入,并且/或者可以被处理以影响视频/音频的基于当前游戏状态的渲染。

[0044] 此外,流媒体逻辑110与视频游戏104之间的通信可以是反馈循环的一部分,其中实现由视频游戏104做出的调整,监测此类调整的结果(例如,通过持续监测流质量),并且在必要时进行额外的调整。例如,如果流媒体逻辑110使视频游戏104响应于网络状况的劣化而降低视频/音频质量,则可能直到通过流媒体逻辑的视频编码器112执行视频编码之后才完全知道该效果是否足够显著以克服网络状况的劣化。并且因此,如果效果不够大,则可能需要流媒体逻辑110向视频游戏104发出附加指令以进一步降低视频质量。在某种意义上,可能需要若干视频帧来评估情况,并且视频游戏104在达到适当的视频/音频质量水平之前需进行多轮调整。应当理解,可以以类似方式处理其中网络状况改善的相反情形,但是在流媒体逻辑指示视频游戏提高视频/音频质量的情况下。

[0045] 尽管本公开的实现方式通常涉及其中在与视频游戏104分离的计算装置上实现流媒体逻辑110的配置,但是应当理解,在其他实现方式中,如本文所述的流媒体逻辑110的功能可以在与视频游戏104(诸如游戏机102)相同的装置上实现的,或者在一些实现方式中,甚至可以在逻辑上并入作为视频游戏104的一部分。

[0046] 图2概念性地示出了根据本公开的实现方式的游戏流质量反馈循环,其中基于流质量调整视频/音频质量。在方法操作200处,起始云视频游戏,这可以包括将用户分配到数据中心和云游戏机、在云游戏机上执行云视频游戏、建立流媒体服务器与用户的客户端装置之间的连接等。在方法操作202处,正在执行的视频游戏的视频/音频被流式传输到客户端装置,并且随着用户与流式传输的视频/音频交互,开始云视频游戏的交互式玩游戏过程,例如使用控制器装置或其他输入装置。

[0047] 在方法操作204处,监测游戏流质量或连接质量或网络状况。作为示例而非限制,这可能需要监测从其流式传输游戏的数据中心与客户端装置之间的状况,诸如数据包丢失、延迟、带宽等。

[0048] 在方法操作206处,确定游戏流质量是否可接受,并且进一步确定是否应该增大或减小视频游戏的质量设置。一般来讲,如果游戏流质量低于给定阈值,则在方法操作208处,

指示视频游戏降低游戏质量设置。作为示例而非限制,低于给定阈值的游戏流质量可以通过各种度量(诸如数据包丢失超过预定义阈值、带宽下降到低于预定义阈值、延迟超过预定义阈值等)来定义。所得的游戏质量设置的降低可以包括本文进一步详细描述的各种措施,诸如降低分辨率或帧速率、降低纹理质量、降低细节设置、减小所渲染的虚拟环境的范围等。

[0049] 与上文类似,如果游戏流质量上升超过给定阈值,则在一些实现方式中可以不采取任何动作,而在其他实现方式中,则在方法操作210处,任选地指示视频游戏提高游戏质量设置。作为示例而非限制,游戏流质量上升超过给定阈值可以通过各种度量(诸如数据包丢失下降到低于预定义阈值、带宽上升超过预定义阈值、延迟下降到低于预定义阈值等)来定义。在执行时,所得的游戏质量设置的提高可以包括本文进一步详细描述的各种措施,诸如提高分辨率或帧速率、提高纹理质量、提高细节设置、增大所渲染的虚拟环境的范围等。

[0050] 综上所述,可以理解,在一些实现方式中,游戏流质量可以具有一个目标范围,其中分别在游戏流质量高于目标范围(的上限)或低于目标范围(的下限)时,向上或向下调整游戏质量设置。

[0051] 可以理解,在游戏质量设置提高或降低之后,该方法返回到方法操作204,其中进一步监测流质量,并根据需要进行附加调整,从而形成针对可能变化的网络状况而优化流质量的反馈循环,并且通过将视频游戏涉及到调整视频游戏的渲染、逻辑或其他设置中的各方面来实现这一点。

[0052] 图3概念性地示出了根据本公开的实现方式的由正在执行的视频游戏本机执行的帧速率相对于在由视频游戏渲染之后由流媒体服务器执行的帧速率的降低。应当理解,根据本公开的实现方式,减少流式传输视频所需的数据量的一种方式降低视频的帧速率。现有的视频编码器可降低帧速率;然而,这在偶数倍数之间进行时效果最好,例如接收60帧每秒(FPS)的视频并以30FPS的速率对其进行编码(并且可能在客户端执行插值),使得帧之间的间隔保持不变。

[0053] 但是,针对可用带宽,从60FPS下降到30FPS可能比必要的情形更多地减少数据量。但是,如果尝试将视频从60FPS降低到例如50FPS,则视频编码器可能会每六帧进行丢弃;但是帧之间的间隔不一致,因此运动可能看起来不流畅,因为帧平滑性不平稳。该情形由附图标记300处所示的已由视频游戏104以60FPS生成的一系列图像帧概念性地说明,并且在每个帧之间具有16.67毫秒(ms)的间隔。但是,如果视频编码器每六帧丢弃一次,则每个丢弃的帧的前一帧和后续帧之间的间隔加倍至33.33ms,并且不再匹配其他帧之间的间隔。虽然可以执行客户端侧插值,但此类插值只是对丢失帧的内容的最佳猜测,并且可能与实际丢弃的帧不匹配,可能会产生伪影或人为外观效果。

[0054] 如果视频游戏自身在本机将帧速率降低到50FPS,则可以获得更好的效果。这实现视频帧(如附图标记302处所示)具有恒定的20ms间隔,使得帧之间不存在不一致。通过使视频游戏自身调整视频帧速率,可以本机生成任何自定义帧速率,使得帧之间的间隔甚至保持一致。如所述,可以响应于感测到的网络状况或流质量来调整帧速率。例如,当流质量或网络状况恶化时,可以使视频游戏104降低其帧速率;并且当流质量或网络状况改善时,可以使视频游戏104提供其帧速率。通过使视频游戏涉及帧速率调整,这允许游戏调整其内部时序和帧被传送的速率,从而解除当前方法的帧平滑限制,使得帧速率降低(或提高)到其

他速率成为可能。

[0055] 图4概念性地示出了根据本公开的实现方式的用于控制或设置视频游戏的帧速率的方法。一些现有的PC/控制台游戏被配置为等待监测器/显示器的垂直同步(Vsync)。即,当游戏将一个视频帧传递给GPU驱动程序时,GPU驱动程序会一直等待,直到显示器开始下一行新图像,在此期间游戏循环被阻断,直到GPU驱动程序接受该帧,从而限制游戏循环并将其同步到显示器的刷新率(例如60Hz、120Hz等)。

[0056] 但是在云视频游戏的情况下,不存在可以与云视频游戏同步的实际显示器。因此,在一些实现方式中,模拟如上所述的实际Vsync的效果的虚拟Vsync被创建。例如,在一些实现方式中,以给定帧速率生成人工Vsync时序信号,所述给定帧速率被设置或调整为实现对视频流的期望效果,诸如减少或增加数据使用。在一些实现方式中,虚拟Vsync的效果由GPU驱动程序创建。

[0057] 在一些实现方式中,与使用如上所述的Vsync时序信号类似的效果是使用定时器/时钟来实现的,如所示实现方式中所示。示出了游戏循环,其中在方法操作400处,处理输入,诸如通过操作诸如游戏控制器或其他装置等输入装置从用户交互输入生成的输入。在方法操作402处,基于游戏的逻辑(诸如实时物理引擎或用于确定游戏的虚拟环境中对象的状态的其他过程)来更新游戏状态。

[0058] 在方法操作404处,然后根据更新后的游戏状态渲染下一视频帧。在方法操作406处,暂停游戏循环,并等待定时器/时钟408之后再继续游戏循环。定时器/时钟408设置帧之间的间隔并且因此被配置为基于从如上所述的流媒体逻辑接收的反馈410将游戏循环限制到期望的帧速率。在一些实现方式中,定时器由操作系统或视频游戏逻辑或游戏引擎提供。虽然在所示实现方式中,游戏循环被示出为在渲染最后一帧之后通过定时器暂停,但在其他实现方式中,游戏循环可以在处理输入和更新游戏状态的其他步骤中的任一个步骤之后暂停。通过实现游戏循环的暂停,可以将游戏循环的视频输出的帧速率设置为任何自定义速率,使得视频帧之间的间隔保持均匀、一致,从而使运动看起来流畅。

[0059] 可以由视频游戏自身调整的玩游戏视频的另一方面是视频的分辨率。在当前的云游戏系统中,流媒体逻辑/流媒体服务器从视频游戏接收完成的视频帧,并且只能在事后对视频帧进行尺寸和颜色深度的缩放。但是,可以通过指示视频游戏自身以较低的分辨率渲染视频而不是在事后进行缩放来提高质量。例如,如果生成的视频帧大小为1280×720像素,则将生成的视频帧缩放到不是1280×720的良好片段的其他分辨率可能会出现问题,并且文本质量可能会受到影响。

[0060] 然而,不是在已经生成此类视频帧之后对此类视频帧进行缩放,而是可以指示(或致使)视频游戏自身以不同的分辨率渲染视频帧,这使得视频游戏能够优化针对所选分辨率的渲染,例如,使得文本质量得以优化。这样,可以将分辨率动态地更改为任意分辨率,而不会因事后缩放而受到损失。例如,当流质量或网络状况恶化时,可以使视频游戏104降低视频分辨率;并且当流质量或网络状况改善时,可以使视频游戏104提供视频分辨率。

[0061] 在一些实现方式中,可以指示视频游戏将视频的纵横比从现有的第一纵横比改变为(不同的)第二纵横比(例如,16:9到4:3等)。改变纵横比对于具有其取向可改变的显示器的装置(诸如移动装置(例如手机、平板计算机等)或用户可以在典型操作期间旋转显示器的其他装置)可以是有用的。因此,在一些实现方式中,感测到的显示器旋转(或取向改变)

可以触发流媒体逻辑来指示视频游戏改变纵横比。这样,视频是从视频游戏本身以适合观看显示器的方式的纵横比来传送的,而不是在视频已经生成之后依赖流媒体逻辑来改变纵横比。应当理解,通过视频游戏改变纵横比可能需要调整通过视频渲染的视野,诸如调整水平/竖直角度视野。在一些实现方式中,改变纵横比以使显示器的全屏区域与供用户观看的正确或优选取向相匹配。

[0062] 在一些实现方式中,检测到装置旋转(或取向改变,例如纵向到横向或反过来)可以触发由本公开其他地方描述的视频游戏影响的任何其他改变。

[0063] 在一些实现方式中,检测到从一个装置切换到另一个装置可以触发由视频游戏执行的调整,诸如视频分辨率、帧速率或纵横比的上述改变。例如,如果用户将玩游戏过程从电视切换到手机,则系统可能会检测到这一点并改变视频分辨率、帧速率或纵横比以适应手机的显示屏。

[0064] 图5概念性地示出了根据本公开的实现方式的虚拟环境中的内容的可变渲染。在一些实现方式中,可以由视频游戏自身(例如响应于关于流质量或网络状况的反馈)调整的另一方面是在视频帧中渲染或示出的虚拟环境中的内容量。实现这一点的一种方式调整虚拟环境中的观看距离。

[0065] 例如,在所示实现方式中,在虚拟环境中概念性地示出了虚拟角色500。被渲染以供视频游戏的用户观看的虚拟环境的视图可以是落在从虚拟角色500的角度来看的视野502内的一部分(并且在虚拟环境中具有可由用户控制的观看方向)。通常,视野502由要渲染的水平角度量和竖直角度量来限定。在给定时间,所渲染的虚拟环境的视图可以从虚拟角色500的位置延伸到虚拟环境内的深度D2(或半径或距离)。即,虚拟环境中落入视野502和深度D2内(即,在所示实现方式中的区域504和506内)的对象将在视频游戏所输出的视频帧中渲染。然而,超出深度D2的对象(诸如落入区域508内的对象)将不会被渲染。这有效地限制了虚拟环境中的对象将被渲染以供观看的距离。

[0066] 在一些实现方式中,例如响应于流质量或网络状况恶化,可以降低渲染深度,以便从虚拟环境中渲染更少的对象。例如,深度可以从D2减小到深度D1,如所示实现方式中所示。并且因此在这种情况下,区域504中的对象被渲染,所述对象落入视野502和深度D1内。而超出深度D1的对象(诸如区域506和508中的对象)没有被渲染以供用户观看。

[0067] 在一些实现方式中,例如响应于流质量或网络状况改善,可以增大渲染深度,以便从虚拟环境中渲染更多数量的对象。例如,深度可以从D2增大到深度D3,如所示实现方式中所示。并且因此在这种情况下,区域504、506和508中的对象都被渲染,因为所述对象落入视野502和深度D3内。应当理解,超出深度D3的对象不会被渲染以供用户观看。

[0068] 可以理解,通过调整视频游戏在视频帧中所渲染的场景中的虚拟对象的量,可以提高或降低视频帧的复杂度,从而改变忠实地传输视频所需的数据带宽量。虽然在上述实现方式中,包括在视图中的虚拟对象的量已基于距离或深度来确定,但是应当理解,可以采用调整要渲染的虚拟对象的量的其他方式。例如,在一些实现方式中,某些类型或类别的对象可以基于流质量和/或网络状况被包括在渲染中/排除在渲染外。在一些实现方式中,对象被分配优先级排名或优先级值,其定义它们包括在虚拟环境的渲染视图中的优先次序。即,当减少要包括在渲染的视频帧中的对象的量时,对象被丢弃的顺序为从较低优先级对象到较高优先级对象。或者相反,当增大要渲染的对象的量时,对象被添加的顺序为从较高

优先级对象到较低优先级对象。

[0069] 因此,可以相对于虚拟对象包括在虚拟环境的渲染图像帧中来打开或关闭虚拟对象。并且当网络状况恶化或流质量劣化时,具有最低渲染优先级的对象将首先从帧渲染中被丢弃。应当理解,一般来讲,预期被排除在渲染之外的对象将是其移除不会显著影响玩游戏过程的对象。

[0070] 视频游戏自身可以调整的另一方面是图像质量。例如,在网络状况劣化的情况下,调整图像质量的目标将是调整场景复杂度以获得更容易编码的视频帧。可以指示视频游戏响应于给定的网络状况而调整图像质量。这可以包括调整应用到虚拟环境中表面的纹理的质量。高质量纹理的编码可能非常复杂,并且因此通过降低纹理的质量,这会减少虚拟环境中表面上的细节量,从而使它们更易于编码并减少流式传输视频的带宽。在一些实现方式中,可以响应于所监测的网络状况来调整纹理的分辨率。例如,可以响应于网络带宽的减小来降低纹理分辨率(例如,4K×4K纹理到1K×1K纹理)。这将导致纹理看起来不那么清晰,但更易于编码器编码,并且可能具有更少的编码伪影,从而产生比在以编码器级别简化较高分辨率的纹理的情形下更好的整体用户体验。

[0071] 在一些实现方式中,调整用于纹理的颜色深度或颜色空间或色域或颜色数量。例如,此类颜色参数可以响应于带宽的减小而减小或向下调整,或者响应于带宽的增加而按比例放大。

[0072] 在一些实现方式中,可以调整(例如,响应于网络带宽的减小而减少)用于对虚拟环境中的给定表面或对象进行建模的多边形或点的数量。

[0073] 在一些实现方式中,照明效果、粒子效果、雾效果、反射或任何其他渲染参数可以响应于网络带宽的减小或流质量的降低而被简化或降低。在一些实现方式中,视频游戏可能不会渲染某些几何形状(例如,降低户外场景中植被的量或细节水平)。

[0074] 在一些实现方式中,根据游戏情景或功能,游戏可以基于诸如带宽等网络状况来对要提供什么视觉内容/情景做出决策。例如,对于在有雾的森林中行走的角色,游戏可以被配置为增加雾量,使得用户在虚拟环境中不能够看到很远处。

[0075] 在一些实现方式中,视频游戏可以实现确定要包括的内容或细节的量的不同级别的内容设置。图6概念性地示出了根据本公开的实现方式的基于网络带宽反馈的若干内容设置。在操作600处,检测可用网络带宽。在操作602处,从几个选项中为视频游戏设置细节水平。例如,在附图标记604处示出了低带宽选项,其中两个角色被示出为带有简单的背景,该背景由表示角色所站立在的草地的简单纹理和没有示出云或其他细节的天空组成。

[0076] 对于附图标记606处所示的中等带宽选项,草地由多边形表示以使得能够看见各个草叶。另外,还示出了一些云,但绘制简单且细节量低。

[0077] 然而,对于附图标记608处所示的高带宽选项,各个草叶数量更多且更密集。此外,草叶可以被配置为摇摆或晃动。此外,云的绘制细节量非常丰富。而且,在高带宽背景中可以看到树木,这些树木先前在较低的细节设置中是不可见的。可以示出额外的植被,诸如花卉、灌木或其他类型的植物。

[0078] 因此,视频游戏可以被配置为响应于网络状况或流质量的改变来调整内容量或所述内容的细节水平。可以进行各种调整,诸如调整观看距离、纹理、细节、诸如植被、角色、车辆等虚拟元素的量。因此,可以动态地修改虚拟环境的质量水平,这与先前的方法不同,因

为本公开的实现方式提供了对内容本身的修改,而不是在它们已经被渲染之后仅仅修改帧速率。在一些实现方式中,可以将视频游戏的内容修改设置为根据需要动态地切换的各种级别,诸如高/中/低带宽备选方案。在一些实现方式中,可以提供用户偏好设置,以使用户能够设置关于内容调整的偏好,诸如设置其相对重要级别。例如,用户可以指示诸如纹理质量、对象数量、分辨率、帧速率等方面的相对偏好级别,并且视频游戏将相应地优先考虑各个方面。

[0079] 在一些实现方式中,应用机器学习方法来确定用户的偏好。例如,在一些实现方式中,系统被配置为分析用户的游戏历史以确定其偏好。可以考虑诸如玩家的水平、他/她死亡的频率、经验级别、选择的难度设置、玩家使用的游戏输入的组合等因素来确定用户偏好。

[0080] 除了进行影响视频图像的调整之外,视频游戏还可以调整视频游戏正在渲染的音频。在现有的云游戏系统中,视频游戏传送以例如立体声、5.1环绕声或其他高保真音频设置渲染的音频,其中此类设置在游戏启动时是固定的。流媒体服务器的音频编码器对音频流进行编码,并且可用于调整音频质量的选项有限。例如,如果网络状况下降(例如,以5.1进行流式传输的用户现在仅具有用于立体声的带宽),则音频编码器将需要丢弃通道或缩混为立体声。此外,音频编码器可以降低音频位速率。但是,此类调整是在音频已经渲染之后进行的。通过调整视频游戏本身的音频,而不依赖下游的编码调整,可以获得更好的结果。

[0081] 在一些实现方式中,用户的音频能力可能发生改变。例如,用户可能正在使用5.1音频设置,但在游戏进行到一半时切换到使用一副头戴式耳机。在这种音频能力改变的情况下,系统可以触发视频游戏改变其音频设置,并且例如从5.1切换到立体声。应当理解,在各种实现方式中,响应于检测到的用户的音频能力或用户的收听装置的改变,可以触发视频游戏将音频设置改变为适合新音频能力或收听装置的音频设置/配置。

[0082] 通过使视频游戏涉及音频调整,可以做出更智能的音频渲染决策,从而提供更好的用户体验。考虑到上面的示例,5.1环绕声的视频游戏渲染可以切换到立体声渲染,这提供比事后进行缩混更好的体验。可以响应于带宽减小而应用的其他选项包括限制音频/声音效果或禁用背景音频,诸如背景音乐。在一些实现方式中,为了响应于用户连接质量而动态地关闭/开启的目的,声音效果可以被优先考虑。一般来讲,在上述示例中,视频游戏响应于诸如带宽减小等网络状况劣化而降低音频质量。同样,可以理解,在相反的情形下,当网络状况改善时,视频可以响应性地提高音频质量。

[0083] 在一些实现方式中,视频游戏可以基于网络连接或流质量(例如延迟和带宽)来调整游戏难度,以便确保视频游戏保持对用户的可玩性和乐趣性。例如,视频游戏可以被配置为响应客户端和流媒体服务器之间的延迟。在一些实现方式中,延迟是基于ping时间确定的。在一些实现方式中,视频游戏可以调整其内部时序,诸如何时其截止时间针对给定输入执行给定动作(例如,按下按钮以进行跳跃或杀死敌人的截止时间)。例如,在一些实现方式中,当延迟增加时,内部时序放宽。即,对于给定动作,允许更多时间或延长截止时间。相反,当延迟减少时,内部时序收紧,即对于给定动作,允许的时间减少或截止时间缩短。例如,这可以实现的效果是,当延迟增加时,使执行诸如攻击或杀死敌人等动作变得更容易,而当延迟减少时,反之则使这些动作更困难。

[0084] 在一些实现方式中,当连接质量下降时,视频游戏会提升游戏内玩家辅助特征(例如,自动瞄准、转向辅助等)的效果,例如,因为当延迟相对较高时,游戏可能变得更难。相反,在一些实现方式中,当连接质量提高时,视频游戏降低游戏内玩家辅助特征的效果。另外,对游戏内辅助特征的上述动态调整具有当网络状况下降时使游戏更容易而当网络状况改善时使游戏更困难的效果。这有助于为用户保持给定水平的公平性。

[0085] 在一些实现方式中,可以响应于用户的装置类型来调整游戏难度。例如,使用触摸控制在手机上玩视频游戏可能比使用控制器在游戏控制台上玩视频游戏更难。因此,当用户从使用一种类型的装置切换到不同类型的装置时,这有助于调整(降低)游戏难度。

[0086] 在一些实现方式中,视频游戏可以被配置为在用户连接质量或流质量特别差或下降到低于最小阈值水平时(例如,用户在其手机上使用4G/5G蜂窝数据网络并且暂时正处于接收不良的地方,诸如在穿过隧道的火车上)暂时暂停游戏。同样,视频游戏可以被配置为在连接/流质量恢复到可接受的水平时取消暂停游戏。在一些实现方式中,可以在游戏暂停的同时示出广告,其中在遇到较差的连接/流质量情况之前,该广告已经被传输到用户装置并存储在该处。

[0087] 在其他实现方式中,视频游戏被系统有意地暂停以便能够呈现广告(而不是响应于较差的连接质量)。

[0088] 在各种实现方式中,流媒体服务器和视频游戏可以以不同方式交互以基于连接/流质量的改变来调整视频或音频。作为示例而非限制,可以存在基于监测到的数据包丢失而用于视频编码器或音频编码器的目标位速率。在一些实现方式中,首先使视频游戏进行调整,然后如果视频游戏进行的调整不充分,则使视频编码器或音频编码器进行进一步调整(例如,缩放、丢弃帧、调整编码器设置以降低位速率、QP设置等)。在某种意义上,流媒体服务器可以在游戏自身进行的调整不足以满足当前连接质量的情况下,回退到自己的机制。

[0089] 在一些实现方式中,流媒体服务器首先执行某些调整,然后如果流媒体服务器进行的调整不充分,则使视频游戏执行调整。

[0090] 在一些实现方式中,视频游戏和流媒体服务器两者都被配置为响应于连接质量改变而基本上同时起始调整。

[0091] 在一些实现方式中,可以存在针对使视频游戏和流媒体服务器中的任一者或两者响应于连接质量来执行调整的阈值。例如,在一些实现方式中,如果延迟超过给定阈值,则这触发视频游戏开始执行调整。

[0092] 在一些实现方式中,流媒体服务器可以根据游戏进行逻辑设置。即,每个游戏都可以定义控制视频游戏将相对于流媒体服务器进行的调整的类型和数量的设置。在一些实现方式中,游戏提供包含针对特定游戏的预设的配置文件。配置文件可以为视频游戏的游戏引擎和流媒体服务器中的任一者或两者提供某些设置的优先级。例如,第一人称射击游戏可优先考虑帧速率而不是分辨率,而益智游戏可优先考虑分辨率而不是帧速率。

[0093] 在一些实现方式中,在流媒体服务器监测与客户端的连接质量时,流媒体服务器确定何时调用视频游戏进行的调整,并且在一些实现方式中,进一步确定视频游戏要进行哪些类型的调整。

[0094] 尽管本文描述的实现方式基于从客户端或流媒体软件返回到视频游戏的反馈来

提供流媒体优化,但也可以存在视频游戏向流媒体服务器或客户端提供信息以实现流媒体优化的前馈路径。图7概念性地示出了根据本公开的实现方式的从视频游戏到流媒体逻辑的前馈优化。概念性地示出了视频游戏的场景700,包括虚拟角色702和门/电梯706。

[0095] 在一些实现方式中,视频游戏可以向视频编码器通知感兴趣区域。视频游戏可以被配置为确定视频帧的哪些部分正在改变,并且可以通知流媒体服务器,使得视频编码器可以专注于正在改变的部分。在现有的游戏流媒体系统中,视频编码器为视频帧的所有不同部分分配一定量的位,因为其对哪一部分可能会改变不知悉。然而,如果画面没有变化,比如四分之三帧,则视频游戏将不会为那些未改变的部分发送数据,并且为那些未改变的部分分配的位是不必要的。因此,与现有系统相比,在一些实现方式中,视频编码器可以将位(其否则将被不必要地分配给未改变的部分)分配给正在经历改变的画面区域,并且因此这些区域可以具有更好的图像质量。这种位分配是基于流媒体服务器从视频游戏本身接收标识感兴趣区域或正在改变的区域的信息。此外,即使整个场景可能正在改变(例如由于视图的平移),标识感兴趣区域对于使视频编码器能够将某些区域优先于其他区域(诸如可能被用户关注的区域,例如,他们的角色所在的区域、他们的目标区域、活动发生的区域等)也可以是有用的。

[0096] 继续参考图7,在所示实现方式中,感兴趣区域704由视频游戏标识(附图标记708),该感兴趣区域是虚拟角色702所在的区域。该区域可以由视频游戏基于虚拟角色702的位置并进一步基于虚拟角色的移动来标识。例如,感兴趣区域可以被标识为包括虚拟角色正在走向的区域或者可能发生交互的区域。所标识的感兴趣区域704被传送到流媒体逻辑110,并且由视频编码器112用来优先考虑感兴趣区域的编码质量。

[0097] 现有的流媒体系统中的视频编码器的另一个挑战发生在场景改变时,因为编码器不知晓并且只“看到”帧中的大量改变(例如,接收到的p帧的大小要大得多)。因此,场景改变具有挑战性,因为帧可能变得太大而无法方便地处理,并且由于改变太多,图像可能需要一些时间才能稳定。结果是,可能需要很多帧才能将图像质量恢复到场景改变之前的状态,在此期间用户体验下降。然而,根据本公开的实现方式,视频游戏可以向视频编码器通知场景改变即将到来或者当前帧是场景改变的一部分。该信息可以由视频编码器使用例如来确定如何分配其位预算。

[0098] 在所示实现方式中,虚拟角色702正朝门/电梯706移动。当虚拟角色702到达门706时,场景将发生改变。视频游戏确定场景改变很可能很快发生,并将该信息传送到流媒体逻辑110(附图标记710)。视频编码器112利用该信息来调整其编码设置(例如分配位预算以处理即将发生的场景改变)。

[0099] 在一些实现方式中,游戏机/游戏系统的操作系统也可以导出某些“前馈”信息。这对于未修改的游戏提供前馈信息特别有用并且/或者可以向流媒体逻辑提供额外线索。例如,过程可以监测游戏正在访问磁盘上的哪些文件(例如,可以指示新关卡或其他事件的加载)或硬件/ssd的吞吐量(例如,吞吐量的突然改变可指示场景改变或其他事件)。在一些实现方式中,该过程可以监测视频游戏正在访问哪些存储器区域。

[0100] 在一些实现方式中,可以(例如,由QA操作者)生成基于这些系统级指示符的游戏配置文件。系统级指示符可以用作标记来确定用户在游戏的位置。因此,视频游戏的适当参考设置可以基于指示符来挑选,并存储到配置文件中。例如,配置文件可以将指示符中的

各种指示符映射到特定参考设置,使得当指示符具有特定值时,则指示视频游戏根据参考设置调整其视频/音频渲染。游戏的不同区域可以具有不同的设置。例如,如果用户处于战斗中(例如根据存储访问、视频图像的机器学习、对音频数据的机器学习、用户按下按钮/摇杆的速率来推断),则可以优先考虑帧速率。而如果用户处于相当静态的区域或表现出低活动性,则分辨率或其他视频属性可能会被优先考虑。

[0101] 在一些实现方式中,数据通道可以从视频游戏延伸到流媒体服务器再到客户端,使得视频游戏可以向客户端发送内容,例如以便绘制覆盖图或发送代码/脚本以提供一些客户端逻辑。视频游戏将前馈到流媒体服务器,该流媒体服务器然后前馈到客户端。例如,视频游戏可以发送共享资产,例如以便为诸如状态信息、玩家健康状况、如击败头目的成就等项目绘制覆盖图。在一些实现方式中,可以发送数据通道的数据以实现触摸界面的客户端侧绘图。在一些实现方式中,通过数据通道发送到客户端的代码可以被配置为渲染定时器,例如,对该游戏将在十秒内完成进行倒计时的时钟。在一些实现方式中,数据通道可用于外围设备(诸如传声器、相机或游戏专用外围设备),以提供视频游戏控制外围设备的处理的方式。

[0102] 此类数据通道将消耗一定量的可用带宽,因此在一些实现方式中,流媒体逻辑可以将可用于数据通道的带宽量传送给视频游戏。在一些实现方式中,视频游戏可以设置要传送给客户端的信息的优先级设置或截止时间。在一些实现方式中,如果需要,可以调整针对数据通道的数据包被传播的速率,例如以便保持一定量的可用于视频/音频的流式传输的带宽。

[0103] 本公开的实现方式可被包括作为游戏引擎的一部分。广义地讲,游戏引擎是提供实现视频游戏的有效开发的特征的软件开发框架。游戏引擎可以包括具有可复用模块的软件库,以处理游戏功能的各个方面,包括(作为示例但不限制)图形渲染(例如,包括顶点处理、多边形处理、着色、照明、纹理化等)、声音、物理效果(包括碰撞处理)、动画、脚本、人工智能、联网、流式传输、内存管理、线程处理、本地化支持、场景图、过场动画等。

[0104] 游戏引擎可以针对不同的硬件平台(诸如游戏控制台、移动装置、个人计算机等)进行优化。作为示例而非限制,游戏引擎可以根据平台优化内存使用(例如,如何在图像流水线中对各种任务进行优先排序等)。在一些实现方式中,硬件可以是一些特定处理实体的刀片版本,诸如游戏控制台。因此,用户可以被分配给特定的刀片,该刀片给出与已针对其优化控制台游戏的相同硬件。

[0105] 可以理解,还可存在游戏服务器逻辑以提供流式传输和/或其他服务(分包、编码、服务质量(QoS)监测、带宽测试、访问社交网络/好友等)。

[0106] 在一些实现方式中,云基础设施可以运行管理程序,其对硬件进行抽象并提供虚拟机框架,操作系统(OS)可以在该虚拟机框架上加载。因此,堆栈可以包括在OS上运行的应用程序/视频游戏,其被加载在由管理程序实例化的虚拟机(VM)上,该虚拟机被加载在底层硬件上。这样,应用程序的执行不必与特定硬件耦合。

[0107] 在一些实现方式中,应用程序/视频游戏可以在容器上执行,该容器在应用层进行抽象,将代码和依赖项打包在一起,从而实现与OS或硬件平台无关的软件开发,并促进跨平台的软件可移植性。

[0108] 在一些实现方式中,采用分布式游戏引擎,其中游戏引擎的不同部分可以由不同

的计算实体处理。例如,游戏引擎的诸如物理引擎、渲染引擎(2D/3D图形)、声音、脚本、动画、AI、联网、流式传输(编码)、内存管理、线程处理等功能可以分为分布在许多不同计算中的不同的功能处理块和/或服务。应当理解,对于分布式游戏引擎,需要低延迟通信以避免延迟问题。为了保持期望的帧速率,计算和通信的总时间应满足某些约束。因此,根据是否可能在更短的时间内完成该过程来划分某些任务可能是高效的,也可能不是高效的。

[0109] 使用分布式游戏引擎的一个优点是可以利用弹性计算,其中计算资源可以根据需要放大或缩小。例如,在传统上在单个硬件服务器上执行的大型多玩家游戏中,在例如大约100名玩家之后,硬件资源变得有限,因此无法添加更多的玩家。游戏可能会让额外的玩家排队,这意味着玩家必须等待加入游戏。然而,使用分布式游戏引擎,通过使用弹性云计算资源,可以添加更多的计算节点来满足需求,从而支持例如数千名玩家。游戏不再受特定硬件服务器的限制。

[0110] 因此,云游戏引擎可以将功能分布到不同的处理实体。可以理解,不同的功能可以在不同的框架中执行。例如,一些功能(例如社交)可更容易地在容器中运行,而使用连接到GPU的VM可更好地运行图形。

[0111] 为了促进云游戏引擎功能的分布,分布/同步层可以管理作业的分布,例如发出作业、收回数据、标识执行哪些任务以及何时执行、处理排队(在一个作业完成的速度比需要的快的情况下)。在一些实现方式中,如果需要,可以动态地细分给定任务。例如,动画可以具有照明,并且如果照明特别复杂,则照明可以细分为三个照明作业,所述照明作业被发出以进行计算并在返回时重组。因此,如果游戏引擎功能需要更多工作,则可以对其进行细分。

[0112] 云服务提供商提供指定性能级别的计算,例如以每秒输入/输出操作数(“IOPS”)。因此,游戏提供商可以从云服务提供商指定VM、专用处理能力、内存量等,并使用云服务提供商的系统来实例化分布式云游戏引擎。

[0113] 在一些实现方式中,库模块和更新处理程序可以是游戏引擎的一个或多个组件或模块。在一些实现方式中,库模块和更新处理程序可以是独立组件,或者是集成的。在一些实现方式中,库模块和更新处理程序可以作为游戏引擎的补充来操作。在一些实现方式中,游戏引擎可以是分布式游戏引擎,如上所述。

[0114] 如所述,本公开的实现方式可应用于云游戏系统。云游戏系统的一个示例是Playstation®Now云游戏系统。在此类系统中,客户端装置可以是游戏控制台,诸如Playstation®4游戏控制台,或者可以是另一个装置,诸如个人计算机、膝上型计算机、平板电脑、手机、移动装置等。

[0115] 广义地讲,为了实现云游戏,当接收到针对游戏名称的用户请求时,由与云游戏站点相关联的数据中心内的一个或多个服务器执行若干操作。当云游戏站点接收到用户请求时,标识托管与所选择的名称相关联的游戏的数据中心,并且将该请求发送到所标识的数据中心,以针对所选择的名称实例化游戏。响应于该请求,数据中心处的服务器标识游戏代码,加载所标识的游戏代码并且初始化与游戏代码相关的文件,以准备将游戏内容呈现给用户。与游戏相关联的游戏数据可包括通用游戏数据和用户特定的游戏数据。因此,初始化文件可以包括标识、加载和初始化通用游戏数据和用户特定的游戏数据。初始化

通用游戏数据可包括初始化图形引擎、安装图形数据、初始化声音文件、安装原图等。初始化用户特定数据可包括定位、传输和安装用户数据、用户历史、游戏历史等。

[0116] 在加载和初始化通用游戏数据时,可提供“启动”画面以在客户端装置处进行渲染。可设计一种启动画面来提供正在加载的游戏的代表性图像,以允许用户预览正在加载的游戏的类型。一旦加载了通用游戏数据,就可以渲染某些初始内容,并且可呈现选择/导航屏幕以供用户选择和定制。在选择/导航屏幕处提供的用户选择输入可包括游戏等级选择、一个或多个游戏图标选择、游戏模式选择、游戏奖励、以及可能需要上传额外游戏内容的其他用户相关数据。在一些实施方案中,通过将游戏内容从游戏云系统流式传输到用户的计算装置来使游戏内容可用于查看和交互。在一些实现方式中,在加载用户特定数据之后,游戏内容可用于进行游戏。

[0117] 图8A示出了用于为可通过云游戏站点获得的游戏加载游戏文件的示例性系统。该系统包括多个客户端装置800,所述多个客户端装置通过网络802通信地连接到云游戏站点804,所述网络可以包括:LAN,有线、无线、蜂窝(例如4G、5G等)网络,或者任何其他类型的数据网络,包括互联网。当从客户端装置800接收到访问云游戏站点804的请求时,云游戏站点804访问存储在用户数据存储区808中的用户账户信息806,以标识与发起请求的客户端装置相关联的用户。在一些实施方案中,云游戏站点还可以验证所标识的用户,以便确定该用户被授权查看/玩的所有游戏。在用户账户标识/验证之后,云游戏站点访问游戏名称数据存储区810以针对发起请求的用户账户标识在游戏云站点处可用的游戏名称。游戏名称数据存储区810继而与游戏数据库812进行交互以获得针对云游戏站点可用的所有游戏的游戏名称。当推出新的游戏时,将用游戏代码更新游戏数据库812,并且将为游戏名称数据存储区810提供新推出的游戏的游戏名称信息。当发起请求时,发起请求的客户端装置可以向或可以不向云游戏站点注册。如果发起请求的客户端装置的用户不是注册用户,则云游戏站点可将该用户标识为新用户,并且选择适合新用户的游戏名称(例如,默认的一组游戏名称)。将所标识的游戏名称返回给客户端装置以呈现在显示屏800-a上,如图8A所示。

[0118] 检测在客户端装置上呈现的游戏名称中的一个游戏名称上的用户交互,并且将信号发送到云游戏站点。该信号包括检测到用户交互的游戏名称信息以及在游戏名称处注册的用户交互。响应于从客户端装置接收的信号,云游戏站点主动确定托管游戏的数据中心,并且向所标识的数据中心发送信号以加载与检测到用户交互的游戏名称相关联的游戏。在一些实施方案中,可能有一个以上的数据中心正在托管游戏。在此类实施方案中,云游戏站点可以确定发起请求的客户端装置的地理位置,并且标识在地理上靠近客户端装置的数据中心,并且发信号通知该数据中心来预加载游戏。可以使用客户端装置内的全球定位系统(GPS)机制、客户端的IP地址、客户端的ping信息等来确定用户的地理位置。当然,检测用户的地理位置的前述方式可以是示例性的,并且可以使用其他类型的机制或工具来确定用户的地理位置。标识靠近客户端装置的数据中心可以使用户与游戏进行交互期间的延迟最小化。在一些实施方案中,所标识的数据中心可能不具有托管游戏所需的带宽/容量或者可能被过度使用。在这些实施方案中,云游戏站点可以标识在地理上靠近客户端装置的第二数据中心。游戏的加载包括加载游戏代码并且执行游戏的实例。

[0119] 响应于从云游戏站点接收到信号,所标识的数据中心可以选择数据中心处的服务器来实例化服务器上的游戏。基于可用的硬件/软件能力和游戏要求来选择服务器。服务器

可以包括多个游戏控制台,并且服务器可以确定使用多个游戏控制台中的哪一个游戏控制台来加载游戏。游戏控制台可以类似于独立的游戏控制台,或者可以是机架式服务器或刀片服务器。刀片服务器继而可以包括多个服务器刀片,其中每个刀片具有用于实例化单个专用应用程序(诸如游戏)所需的电路。当然,上述游戏控制台是示例性的,且不应被视为是限制性的。其他类型的游戏控制台(包括游戏站等)以及其他形式的刀片服务器也可以用于托管所标识的游戏。

[0120] 一旦标识游戏控制台,游戏的通用游戏相关代码就被加载到游戏控制台上,并且通过网络经由云游戏站点向客户端装置返回信号,从而标识在其上实例化游戏的游戏控制台。因此,加载的游戏可供用户使用。

[0121] 图8B是概念性地示出根据本公开的实现方式的经执行用于将云视频游戏流式传输到客户端装置的各种操作的流程图。游戏系统818执行视频游戏并且生成原始(未压缩的)视频820和音频822。视频820和音频822被捕获和编码以用于流式传输目的,如在所示的图中的附图标记824处所指示。编码可用于视频和音频流的压缩,以减少带宽使用并且优化游戏体验。编码格式的示例包括H.265/MPEG-H、H.264/MPEG-4、H.263/MPEG-4、H.262/MPEG-2、WMV、VP6/7/8/9等。

[0122] 经编码的音频826和经编码的视频828被进一步分包为网络数据包,如附图标记832所指示,以用于通过网络(诸如互联网)进行传输。网络数据包编码过程也可以采用数据加密过程,从而提供增强的数据安全性。在所示的实现方式中,如附图标记840所指示,生成音频数据包834和视频数据包836以用于通过网络进行传输。

[0123] 游戏系统818另外生成触觉反馈数据830,所述触觉反馈数据也被分包为网络数据包以用于网络传输。在所示的实现方式中,生成触觉反馈数据包838以用于通过网络进行传输,如在附图标记840处进一步指示。

[0124] 在共同定义云游戏服务/系统的一个或多个服务器上执行上述生成原始视频和音频以及触觉反馈数据、对视频和音频进行编码并且对经编码的音频/视频和触觉反馈数据进行分包以进行传输的操作。如附图标记840所指示,音频、视频和触觉反馈数据包通过网络(诸如和/或包括互联网)进行传输。如附图标记842所指示,音频数据包834、视频数据包836和触觉反馈数据包838由客户端装置解码/重组,以在客户端装置处定义经编码的音频846、经编码的视频848和触觉反馈数据850。如果数据已经加密,则也将对网路数据包解密。然后,如附图标记844所指示,由客户端装置对经编码的音频846和经编码的视频848进行解码,以生成客户端侧原始音频和视频数据以在显示装置852上渲染。可以处理/传送触觉反馈数据850以在控制器装置856或可以呈现触觉效果的其他接口装置处产生触觉反馈效果。触觉效果的一个示例是控制器装置856的振动或隆隆声。

[0125] 应理解,视频游戏对用户输入进行响应,并且因此可以执行与上述用于用户输入的传输和处理的过程类似但是在从客户端装置到服务器的相反方向上的程序流程。如图所示,用户操作控制器装置856可生成输入数据858。该输入数据858在客户端装置处进行分包,以通过网络传输到云游戏系统。输入数据包860被云游戏服务器拆包并且重组以在服务器侧定义输入数据862。输入数据862被馈送到游戏系统818,所述游戏系统处理输入数据862以更新视频游戏的游戏状态。

[0126] 在音频数据包834、视频数据包836和触觉反馈数据包838的传输(附图标记840)期

间,可以监测通过网络的数据传输以确保云游戏流的服务质量。例如,可以如附图标记864所指示监测网络状况,包括上游和下游网络带宽两者,并且可以响应于可用带宽的变化来调整游戏流式传输。即,可以基于当前网络状况来控制网络数据包的编码和解码,如附图标记866所指示。

[0127] 图9示出了信息服务提供商架构的实施方案。信息服务提供商 (ISP) 970向地理上分散且经由网络986连接的用户1082提供大量信息服务。ISP可以只提供一种类型的服务,诸如股票价格更新,或可以提供多种服务,诸如广播媒体、新闻、体育、游戏等。另外,每个ISP所提供的服务都是动态的,即,可以在任何时间点添加或移除服务。因此,向特定个人提供特定类型的服务的ISP可能会随时间变化。例如,当用户在她的家乡时,可以由靠近该用户的ISP为用户提供服务,并且当用户前往不同城市时,可以由不同的ISP为用户提供服务。当地ISP将把所需的信息和数据传输到新的ISP,使得用户信息“跟随”用户到新的城市,从而使数据更接近用户并且更易于访问。在另一个实施方案中,可以在为用户管理信息的主ISP和在主ISP的控制下直接与用户对接的服务器ISP之间建立主-从关系。在另一个实施方案中,当客户端在世界范围内移动时,数据从一个ISP传输到另一个ISP,以使处在更好位置为用户提供服务的ISP成为提供这些服务的ISP。

[0128] ISP 970包括应用服务提供商 (ASP) 972,其通过网络向客户提供基于计算机的服务(例如,包括但不限于任何有线或无线网络、LAN、WAN、WiFi、宽带、电缆、光纤、卫星、蜂窝(例如4G、5G等)、互联网等)。使用ASP模型提供的软件有时也称为按需软件或软件即服务(SaaS)。提供对特定应用程序(诸如客户关系管理)的访问的简单形式是使用标准协议(诸如HTTP)。该应用程序软件驻留在供应商的系统上,并且由用户使用HTML通过web浏览器进行访问、通过由供应商提供的专用客户端软件进行访问或通过其他远程接口(诸如瘦客户端)进行访问。

[0129] 在广泛的地理区域内提供的服务通常使用云计算。云计算是一种计算方式,其中动态可扩展和通常虚拟化的资源通过互联网作为服务提供。用户不需要成为支持他们的“云”中技术基础设施方面的专家。云计算可以分为不同的服务,例如基础设施即服务(IaaS)、平台即服务(PaaS)和软件即服务(SaaS)。云计算服务通常提供在线的从web浏览器访问的通用业务应用,而软件和数据则存储在服务器上。基于计算机网络图中互联网的描绘方式,术语云被用作互联网的隐喻(例如,使用服务器、存储装置和逻辑),并且是对它所隐藏的复杂基础设施的抽象概念。

[0130] 此外,ISP 970包括游戏处理服务器 (GPS) 974,该GPS被游戏客户端用来玩单玩家和多玩家视频游戏。在互联网上玩的大多数视频游戏都是通过与游戏服务器的连接来运行的。通常,游戏使用专用的服务器应用程序,所述服务器应用程序从玩家收集数据并将其分发给其他玩家。这比对等布置更有效且高效,但它需要单独的服务器来托管服务器应用。在另一个实施方案中,GPS在玩家之间建立通信,并且玩家的相应的玩游戏装置在不依赖于集中式GPS的情况下交换信息。

[0131] 专用GPS是独立于客户端运行的服务器。这样的服务器通常在位于数据中心内的专用硬件上运行,从而提供更多的带宽和专用的处理能力。对于大多数基于PC的多玩家游戏,专用服务器是托管游戏服务器的优选方法。大型多玩家在线游戏在专用服务器上运行,所述服务器通常由拥有游戏名称的软件公司托管,从而允许它们控制和更新内容。

[0132] 广播处理服务器(BPS) 976将音频或视频信号分发给观众。向很小范围的观众广播有时称为窄播。广播分发的最后一站是信号如何到达听众或观看者,并且它可能像广播电台或电视台一样从空中传播到天线和接收器,或可能通过有线电视或有线广播(或“无线电缆”)通过工作站或直接从网络传播。互联网也可以将收音机或电视中的任一者带给接收者,尤其是利用允许共享信号和带宽多播。历史上,广播已经由地理区域界定,诸如国家广播或区域广播。但是,随着快速互联网的普及,不按地理条件来定义广播,因为内容可以到达世界上的几乎任何国家。

[0133] 存储服务提供商(SSP) 978提供计算机存储空间和相关的管理服务。SSP还提供周期性备份和存档。通过提供存储即服务,用户可以根据需要订购更多存储。另一个主要优点是SSP包括备份服务,并且如果计算机的硬盘驱动器发生故障,用户将不会丢失其所有数据。此外,多个SSP可以具有用户数据的全部或部分副本,从而允许用户以有效的方式访问数据,而与用户所在的位置或用于访问数据的装置无关。例如,用户可以在家用计算机以及移动电话中(当用户移动时)访问个人文件。

[0134] 通信提供商980向用户提供连接。一种通信提供商是互联网服务提供商(ISP),其提供对互联网的访问。ISP使用适合于传递互联网协议数据报的数据传输技术来连接其客户,诸如拨号、DSL、电缆调制解调器、光纤、无线或专用高速互连。通信提供商还可以提供消息传递服务,诸如电子邮件、即时消息传递和SMS短信。另一种类型的通信提供商是网络服务提供商(NSP),其通过提供对互联网的直接主干访问来出售带宽或网络接入。网络服务提供商可以包括电信公司、数据运营商、无线通信提供商、互联网服务提供商、提供高速互联网接入的有线电视运营商等。

[0135] 数据交换988将ISP 970内部的若干模块互连,并经由网络986将这些模块连接到用户982。数据交换988可以覆盖ISP 970的所有模块都非常靠近的小区域,或者当不同的模块在地理上分散时可以覆盖大的地理区域。例如,数据交换988可以包括数据中心的机柜内的快速千兆以太网(或更快的以太网),或洲际虚拟区域网络(VLAN)。

[0136] 用户982使用客户端装置984访问远程服务,该客户端装置至少包括CPU、存储器、显示器和I/O。客户端装置可以是PC、移动电话、笔记本计算机、平板计算机、游戏系统、PDA等。在一个实施方案中,ISP 970辨识由客户端使用的装置类型并且调整所采用的通信方法。在其他情况下,客户端装置使用标准通信方法(诸如html)来访问ISP 970。

[0137] 本公开的实施方案可以用包括手持式装置、微处理器系统、基于微处理器的或可编程的消费电子产品、小型计算机、大型计算机等的各种计算机系统配置来实践。本公开也可以在分布式计算环境中实践,其中由通过有线或无线网络链接的远程处理装置执行任务。

[0138] 在一些实施方案中,可使用无线技术来促进通信。此类技术可以包括例如5G无线通信技术。5G是第五代蜂窝网络技术。5G网络是数字蜂窝网络,其中提供商覆盖的服务区域被划分为称为小区的小地理区域。代表声音和图像的模拟信号在电话中被数字化、由模数转换器转换并作为位流传输。通过从其他小区中重复使用的频率池中由收发器分配的频率通道,小区中的所有5G无线装置通过无线电波与小区中的本地天线阵列和小功率自动收发器(发射器和接收器)通信。本地天线通过高带宽光纤或无线回程连接与电话网络和互联网连接。与其他小区网络一样,移动装置从一个小区跨越到另一个小区会自动转移到新小区。

应当理解,5G网络只是通信网络的示例类型,并且本发明的实施方案可以利用更早一代的无线或有线通信,以及5G之后的更新一代的有线或无线技术。

[0139] 考虑到上述实施方案,应理解,本公开可以采用涉及存储在计算机系统中的数据的各种计算机实现的操作。这些操作是需要物理地操纵物理量的那些操作。形成本公开的一部分的本文描述的任何操作都是有用的机器操作。本公开还涉及用于执行这些操作的装置或设备。可以针对所需目的专门构造所述设备,或者所述设备可以是由存储在计算机中的计算机程序选择性地激活或配置的通用计算机。具体地,各种通用机器可与根据本文的教导编写的计算机程序一起使用,或者构造更专门的设备来执行所需操作可能更方便。

[0140] 本公开还可以实施为计算机可读介质上的计算机可读代码。可替代地,可以使用上述数据交换互连从服务器下载计算机可读代码。计算机可读介质是可以存储数据的任何数据存储装置,该数据随后可以由计算机系统读取。计算机可读介质的示例包括硬盘驱动器、网络附接存储装置(NAS)、只读存储器、随机存取存储器、CD-ROM、CD-R、CD-RW、磁带以及其他光学和非光学数据存储装置。所述计算机可读介质可以包括分布在网络耦合的计算机系统上的计算机可读有形介质,使得以分布式方式存储和执行计算机可读代码。

[0141] 尽管以具体的顺序描述了方法操作,但应理解,可在操作之间执行其他内务操作,或者可以调整操作,使得它们在略微不同的时间发生,或者可以分布在允许以与处理关联的各种间隔发生处理操作的系统中,只要覆盖操作的处理以期望的方式执行即可。

[0142] 虽然出于清楚理解的目的相当详细地描述了前述公开内容,但显而易见的是,可以在所附权利要求的范围内实践某些改变和修改。因此,本实施方案将被认为是说明性的而非限制性的,并且本公开不限于本文给出的细节,而是可在所描述的实施方案的范围和等效范围内进行修改。

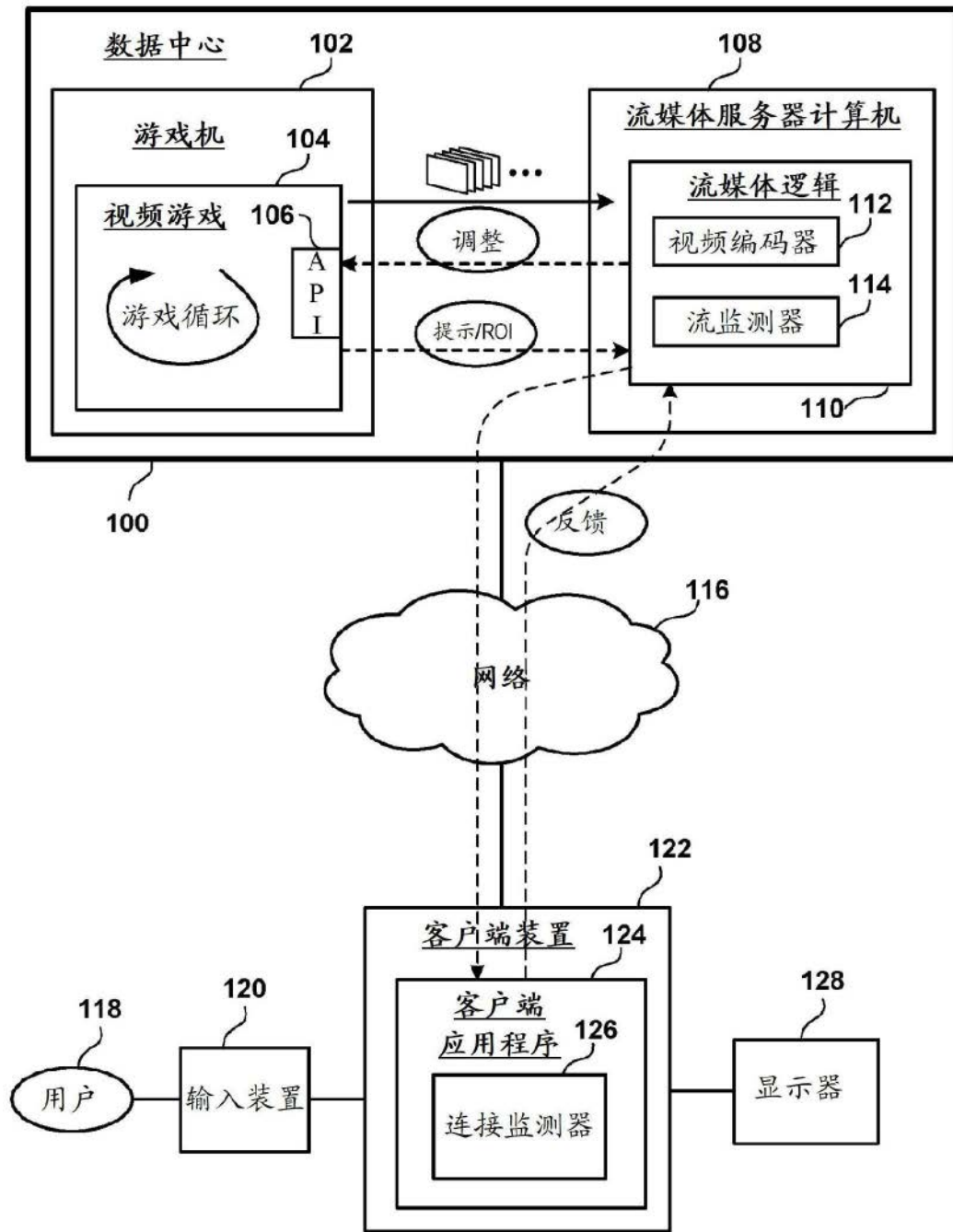


图1

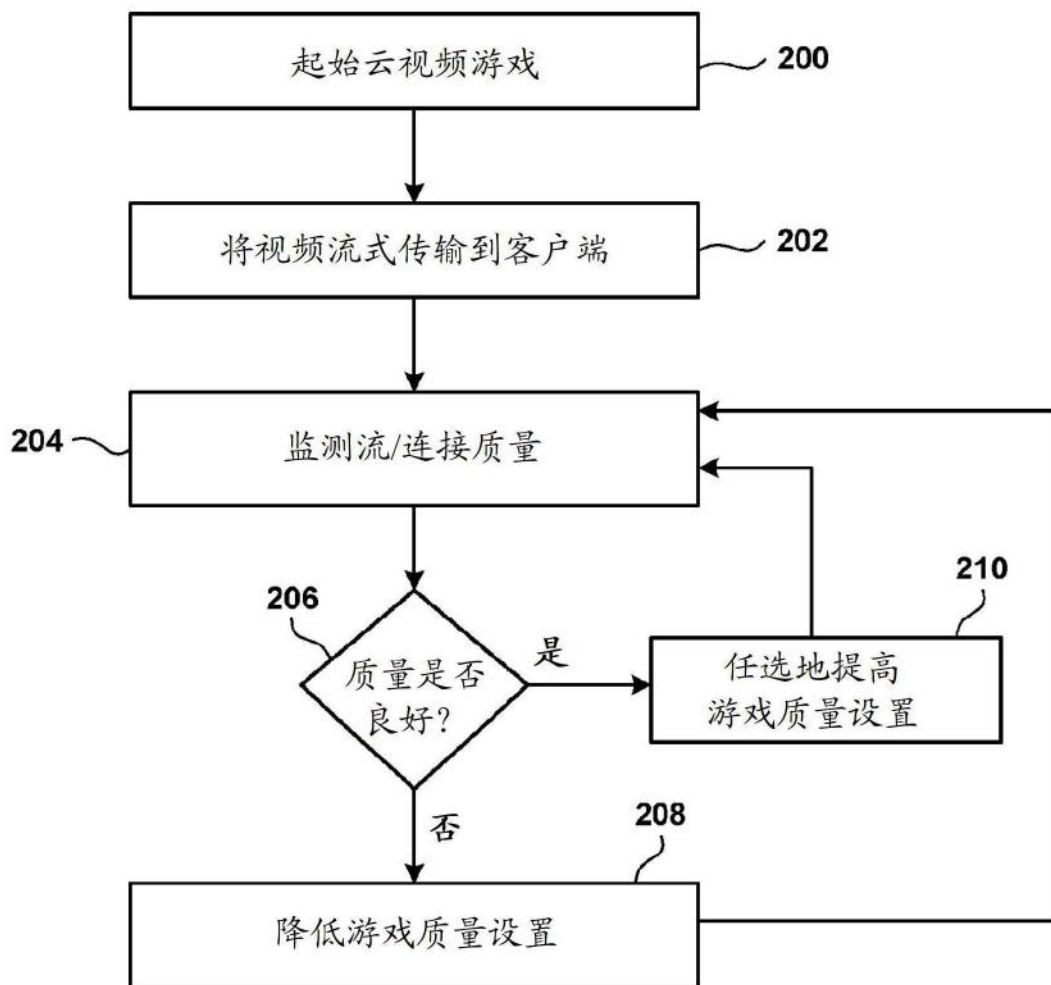


图2

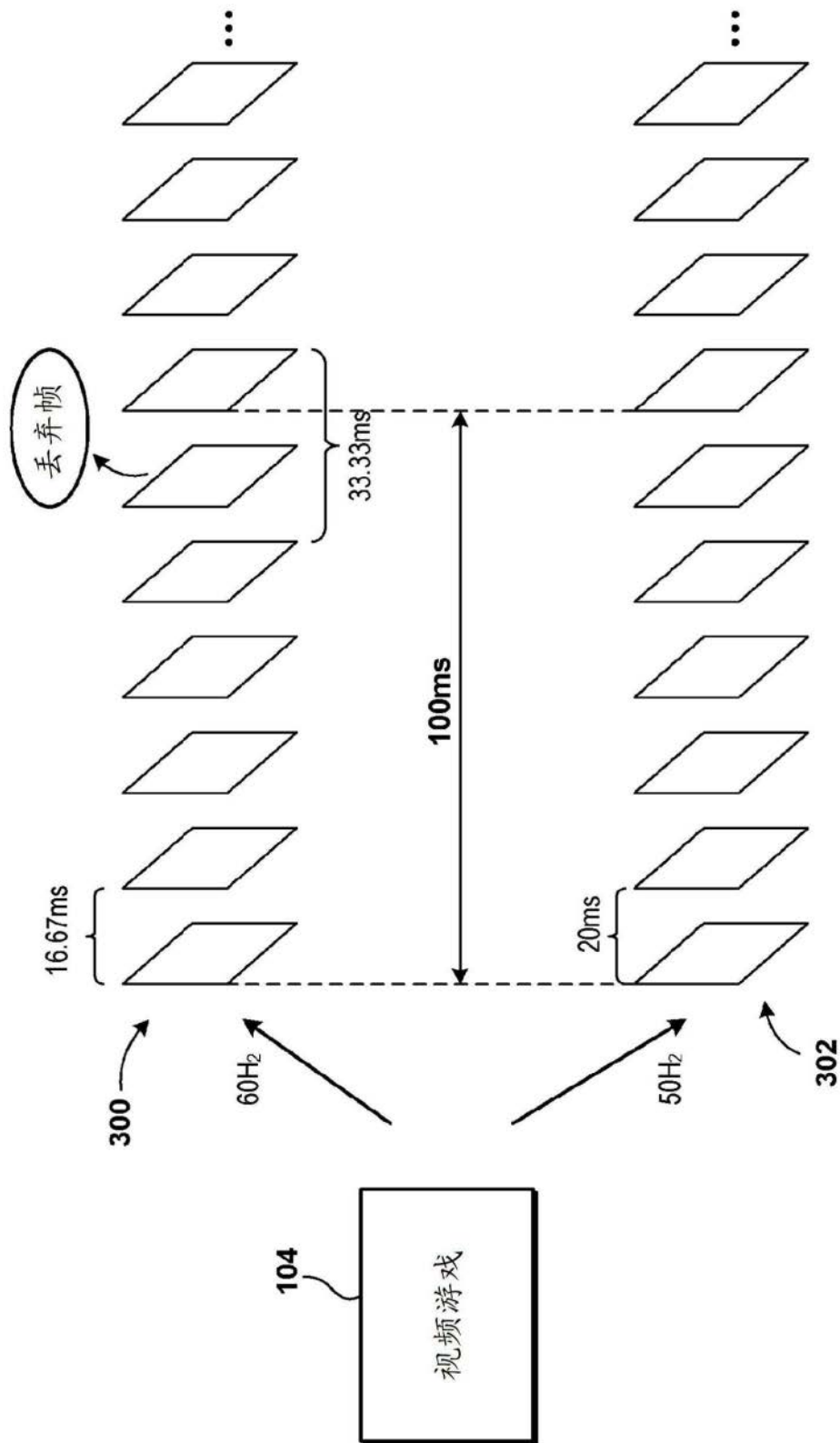


图3

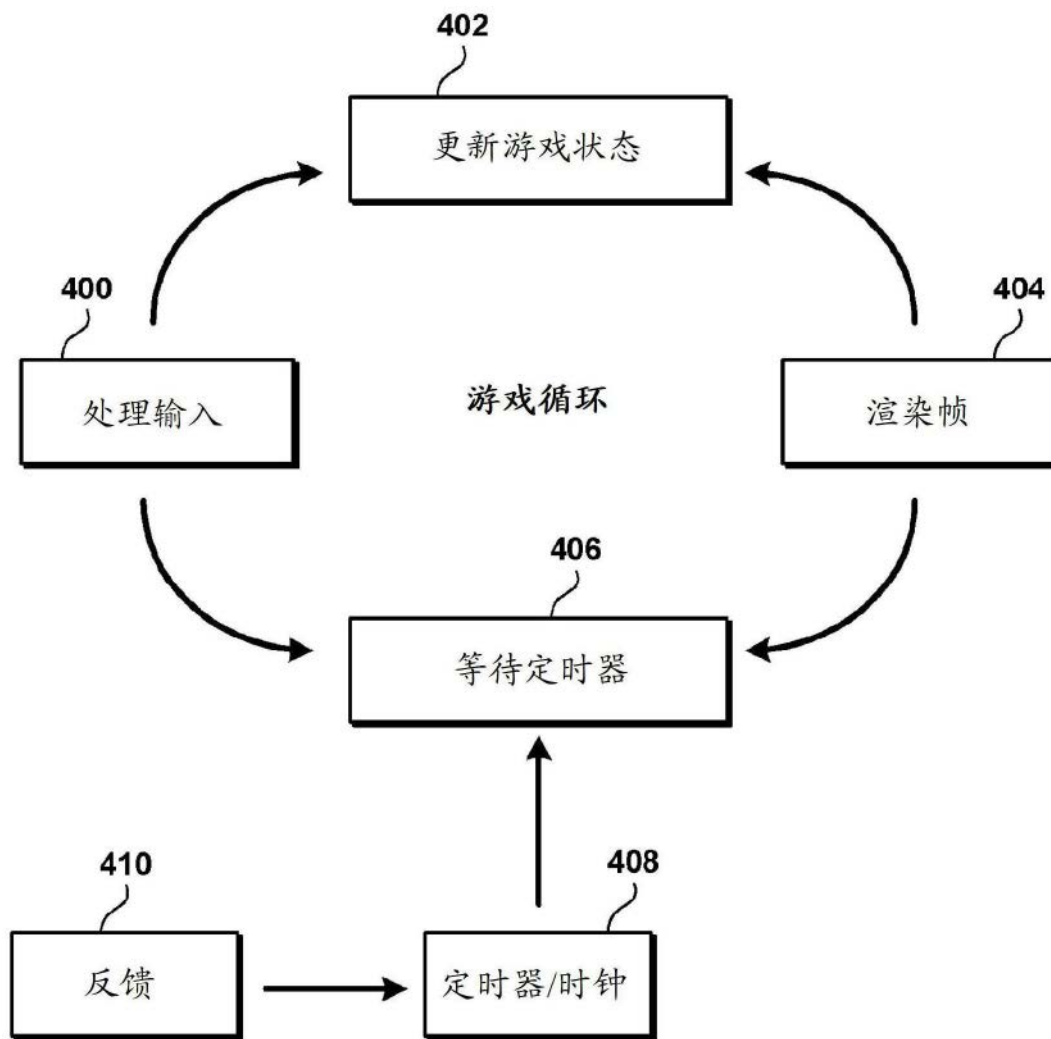


图4

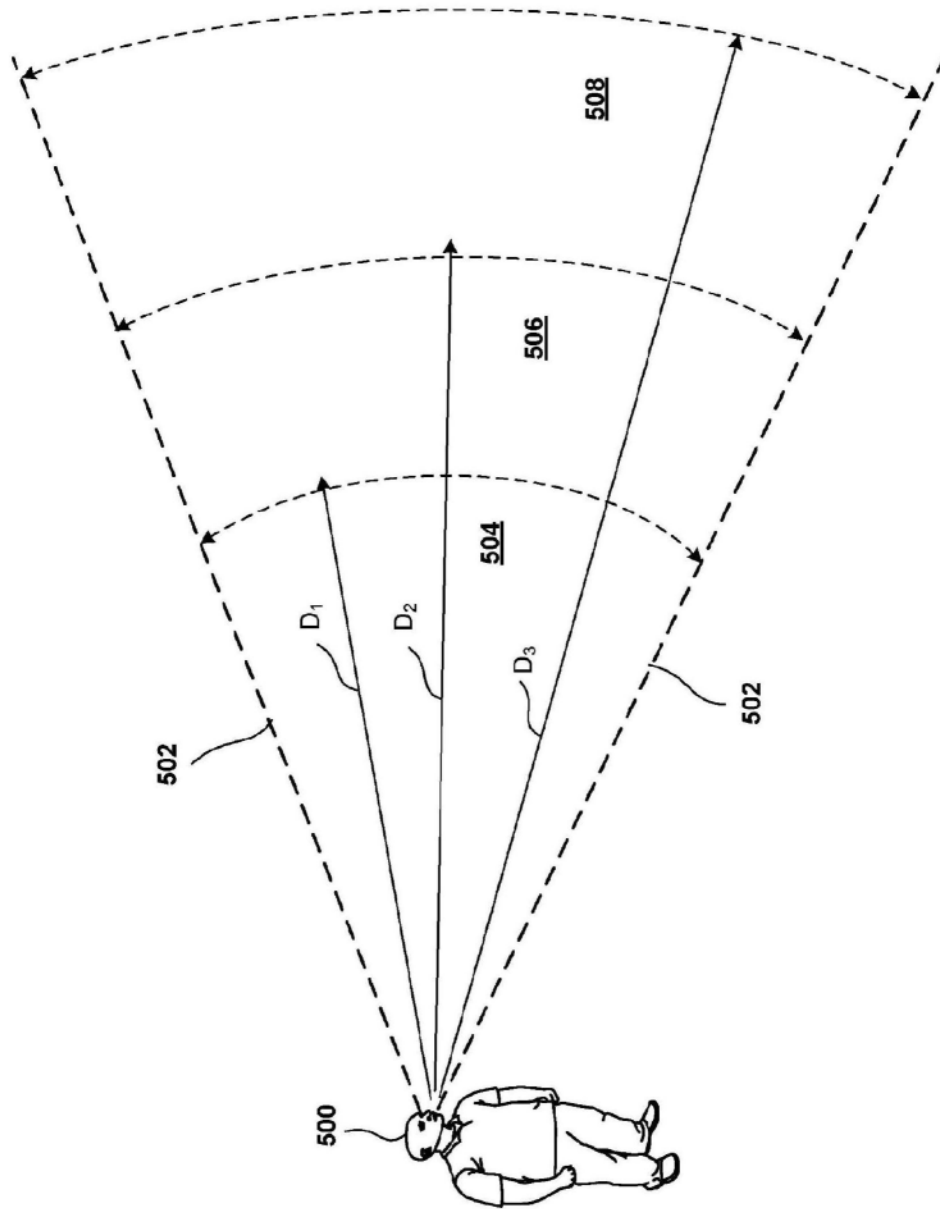


图5

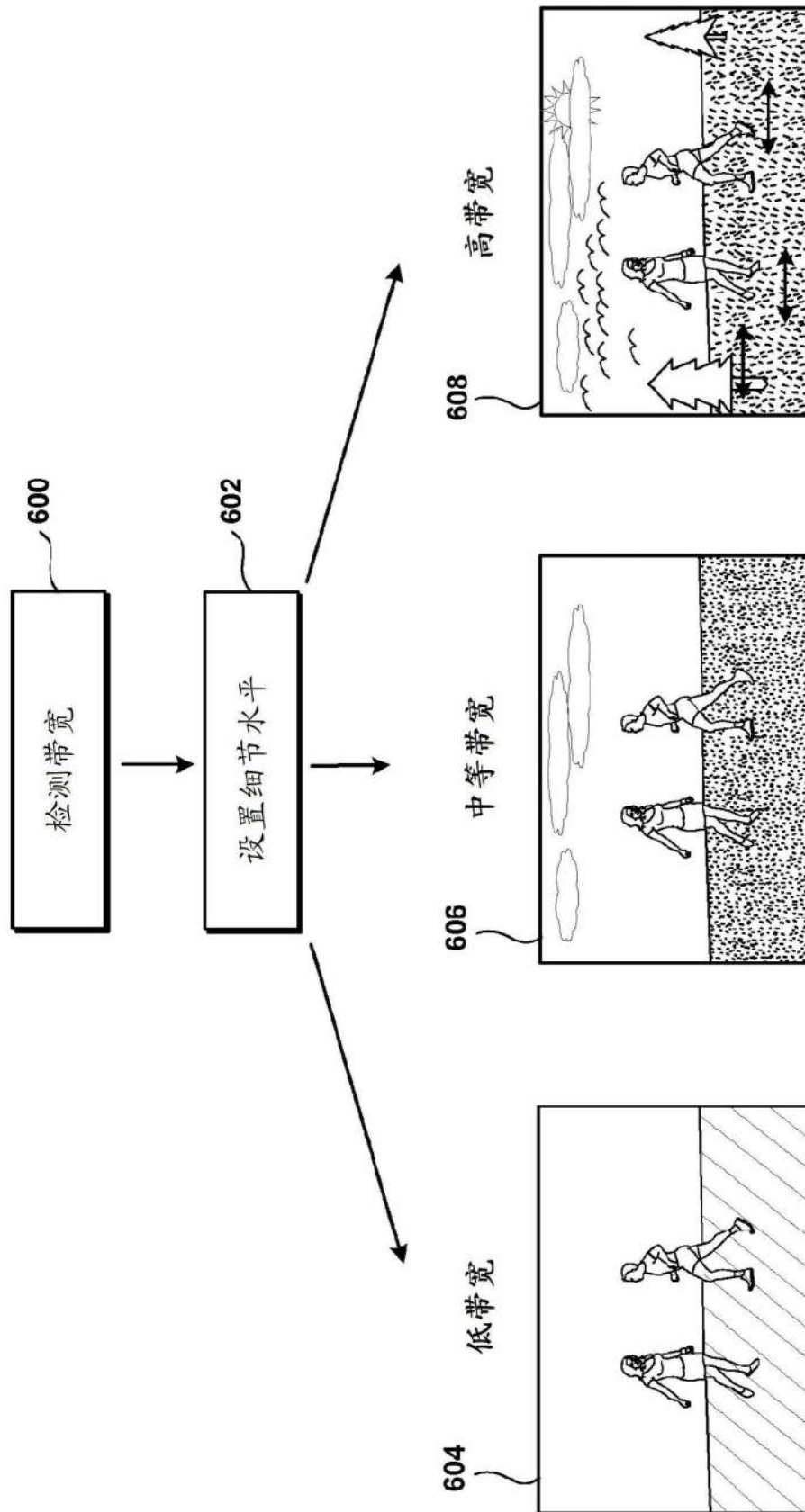


图6

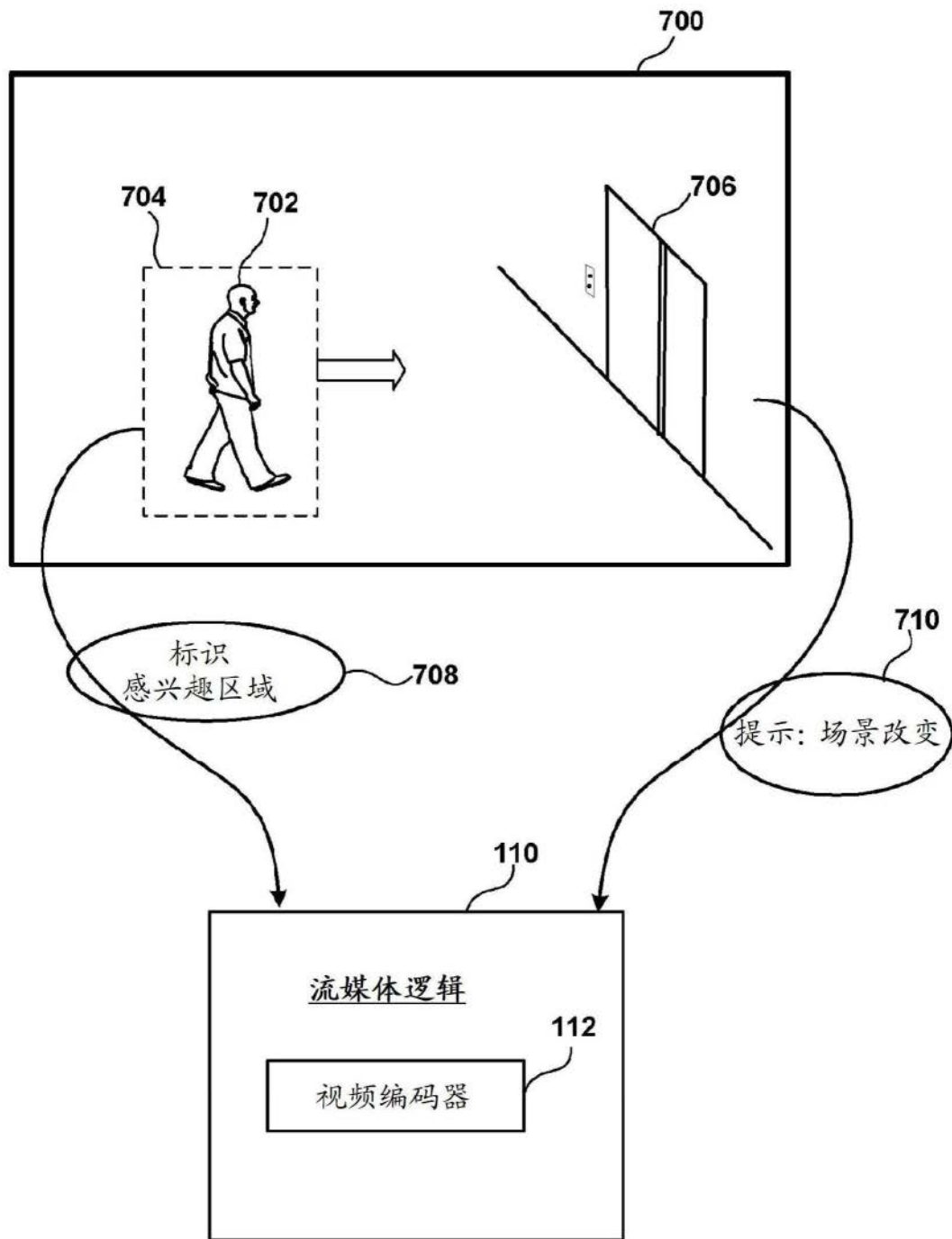


图7

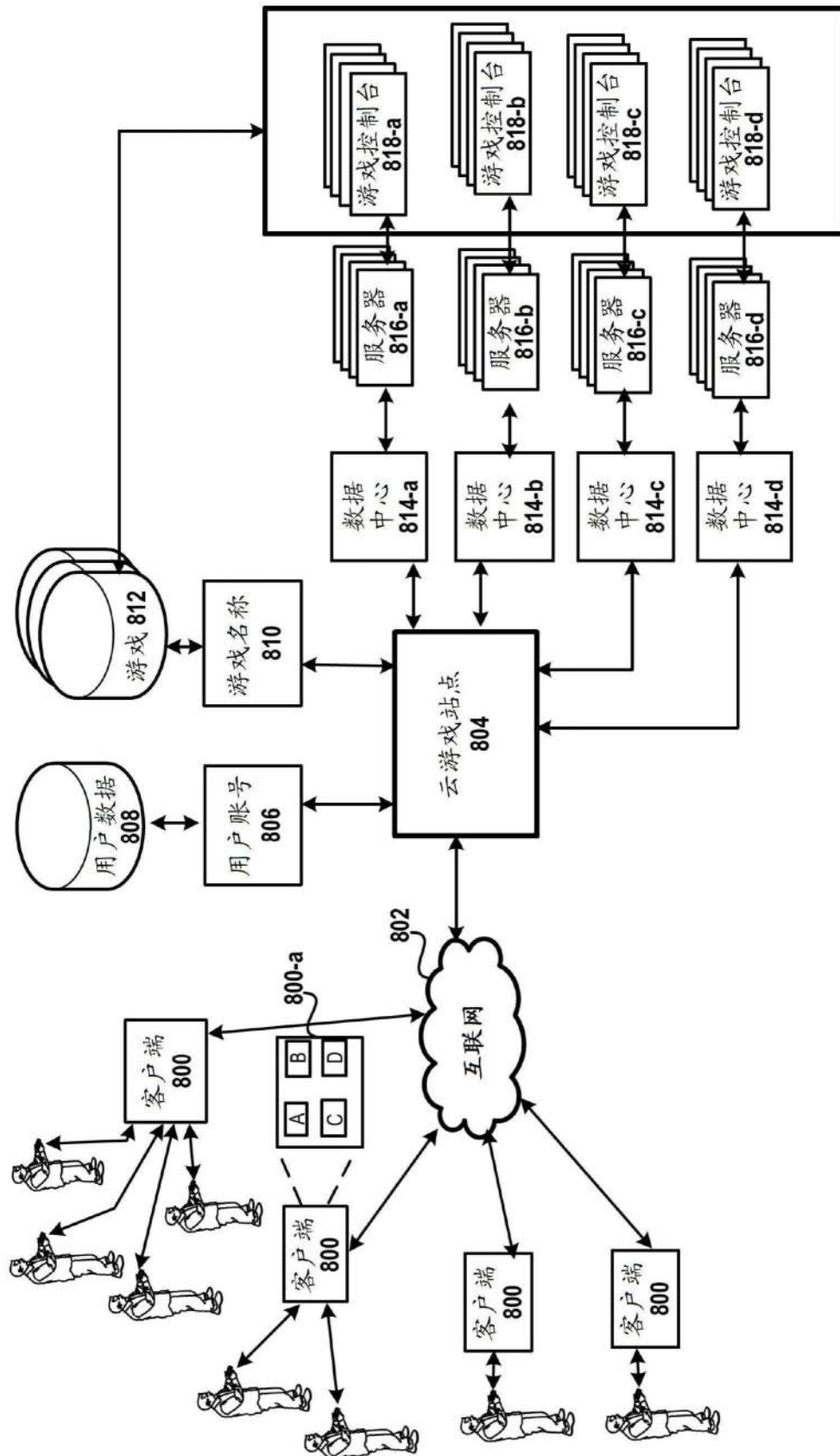


图8A

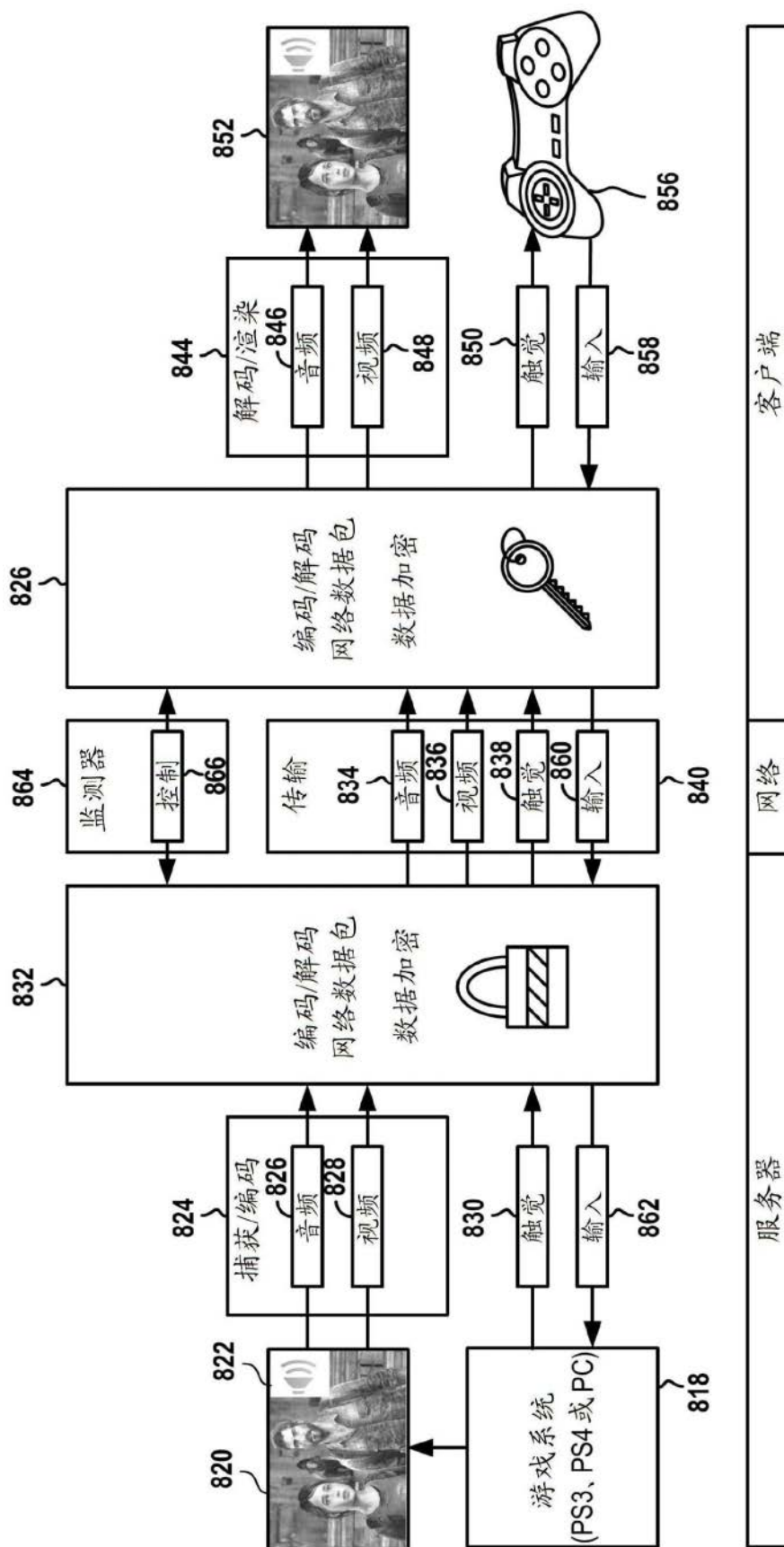


图8B

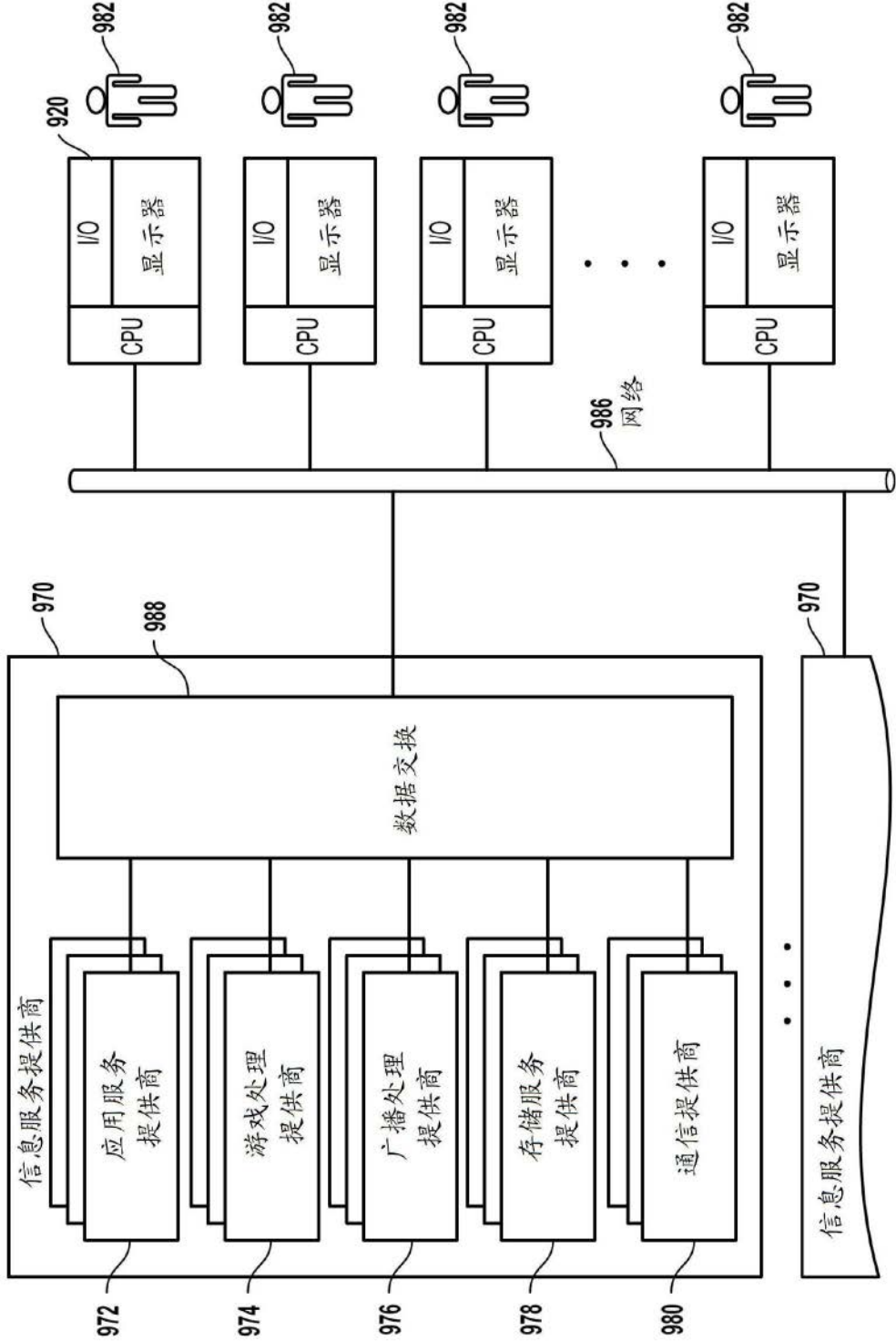


图9