



(12) 发明专利申请

(10) 申请公布号 CN 113661471 A

(43) 申请公布日 2021. 11. 16

(21) 申请号 202080026304.9

(74) 专利代理机构 北京市金杜律师事务所  
11256

(22) 申请日 2020.03.09

代理人 马明月

(30) 优先权数据

16/379,693 2019.04.09 US

(51) Int.Cl.

G06F 3/01 (2006.01)

(85) PCT国际申请进入国家阶段日

G06T 19/00 (2006.01)

2021.09.29

G06F 9/451 (2006.01)

(86) PCT国际申请的申请数据

PCT/US2020/021616 2020.03.09

(87) PCT国际申请的公布数据

WO2020/209962 EN 2020.10.15

(71) 申请人 微软技术许可有限责任公司

地址 美国华盛顿州

(72) 发明人 D·B·弗罗姆霍尔德

J·M·莱昂斯 B·M·陶特

A·A·米查伊尔

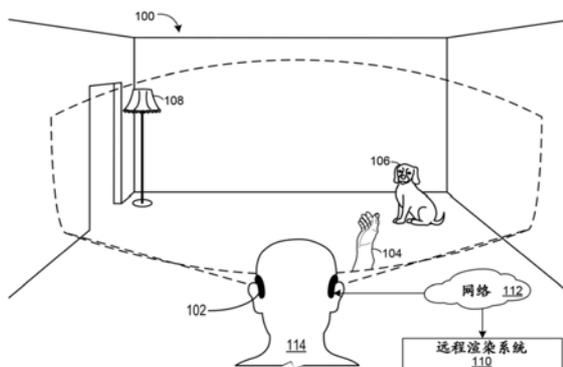
权利要求书2页 说明书13页 附图6页

(54) 发明名称

混合渲染

(57) 摘要

一个公开的示例提供了一种计算设备,该计算设备包括处理设备和存储设备,该存储设备存储执行,该指令由处理设备可执行以在第一本地过程中执行输出用于渲染和显示的数字内容的应用。在应用的执行期间,该指令可执行以向第二本地或远程过程提供关于要由第二本地或远程过程渲染的对象的对象信息,从第二本地或远程过程接收对象的渲染,输出对象的渲染以显示对象,接收对对象进行的操纵,基于对对象进行的操纵向第二本地或远程过程提供已更新的对象信息,从第二本地或远程过程接收对象的已更新渲染,并且输出对象的已更新渲染以显示对象。



1. 一种计算设备,包括:  
处理设备;以及  
存储设备,存储指令,所述指令由所述处理设备可执行以:  
在第一本地过程中执行输出用于渲染和显示的数字内容的应用;  
在所述应用的执行期间,向第二本地或远程计算系统过程提供关于要由所述第二本地或远程计算系统过程渲染的对象的对象信息;  
从所述第二本地或远程计算系统过程接收所述对象的渲染;  
向显示器输出所述对象的所述渲染以显示所述对象;  
接收对所述对象进行的操纵;  
基于对所述对象进行的所述操纵,向所述第二本地或远程计算系统过程提供已更新的对象信息;  
从所述第二本地或远程计算系统过程接收所述对象的已更新渲染;以及  
向所述显示器输出所述对象的所述已更新渲染以显示所述对象。
2. 根据权利要求1所述的计算设备,其中所述对象是第一对象,并且其中所述指令还可执行以:  
本地地渲染第二对象以生成所述第二对象的渲染;以及  
合成所述第一对象的所述渲染和所述第二对象的所述渲染以用于作为合成帧显示。
3. 根据权利要求2所述的计算设备,其中从所述第二本地或远程过程所接收的所述对象的所述渲染包括第一颜色缓冲,并且其中所述指令还可执行以在合成所述第一对象的所述渲染和所述第二对象的所述渲染之前,执行所述第一颜色缓冲的重投影。
4. 根据权利要求2所述的计算设备,其中所述指令还可执行以在合成所述第一对象的所述渲染和所述第二对象的所述渲染之后,执行所述合成帧的颜色缓冲的重投影。
5. 根据权利要求2所述的计算设备,其中从所述第二本地或远程过程所接收的所述第一对象的所述渲染包括第一深度数据,并且其中所述第二对象的所述渲染包括第二深度数据,并且其中所述指令可执行以通过如下操作来合成所述第一对象的所述渲染和所述第二对象的所述渲染:对所述第一深度数据和所述第二深度数据执行z测试,并且基于z测试来合成所述第一对象和所述第二对象。
6. 根据权利要求1所述的计算设备,其中所述指令可执行以通过提供包括所述对象的几何形状的三维模型来提供所述对象信息。
7. 根据权利要求1所述的计算设备,其中所述指令可执行以通过提供用于更新被数字渲染的所述内容的数据结构的请求来提供所述对象信息。
8. 根据权利要求1所述的计算设备,其中所述对象的所述渲染包括视频数据,所述视频数据包括颜色缓冲。
9. 根据权利要求8所述的计算设备,其中所述视频数据还包括深度的表示。
10. 根据权利要求1所述的计算设备,其中所述指令可执行以:通过接收指示头部姿势和/或眼睛注视方向的变化了的传感器数据、和/或通过接收用于操纵所述对象的用户输入,来接收对所述对象进行的所述操纵。
11. 一种在计算设备上实施的方法,所述方法包括:  
执行输出用于渲染和显示的数字内容的应用;

在所述应用的执行期间：

标识要远程地被渲染的所述数字内容的第一对象，并且还标识要本地地被渲染的所述数字内容的第二对象；

向远程计算系统提供关于要由所述远程计算系统渲染的所述第一对象的对象信息；

从所述远程计算系统接收所述第一对象的渲染；

本地地渲染所述第二对象以生成所述第二对象的渲染；并且

合成所述第一对象的所述渲染和所述第二对象的所述渲染以用于作为合成帧显示。

12. 根据权利要求11所述的方法，其中所述第一对象的所述渲染包括第一颜色缓冲和第一深度缓冲，所述方法还包括，在合成所述第一对象的所述渲染和所述第二对象的所述渲染之前，执行至少所述第一颜色缓冲的重投影。

13. 根据权利要求11所述的方法，还包括：在合成所述第一对象的所述渲染和所述第二对象的所述渲染之后，执行重投影。

14. 根据权利要求12所述的方法，其中执行所述重投影包括：基于用户的头部姿势来执行所述重投影。

15. 根据权利要求11所述的方法，其中所述第一对象的所述渲染包括第一深度数据，并且其中所述第二对象的所述渲染包括第二深度数据，所述方法还包括：

对所述第一深度数据和所述第二深度数据执行深度测试；以及

基于所述深度测试来合成所述第一对象的所述渲染和所述第二对象的所述渲染。

## 混合渲染

### 背景技术

[0001] 计算设备通常包括图形渲染管线以渲染用于在二维显示器上显示的三维模型。

### 发明内容

[0002] 本发明内容被提供来以简化的形式介绍以下在详细描述中进一步描述的概念的选择。本发明内容不旨在标识要求保护的的主题的关键特征或必要特征,也不旨在被用来限制要求保护的的主题的范围。此外,要求保护的的主题不限于解决在本公开的任何部分中指出的任何或所有缺点的实现。

[0003] 公开了涉及混合渲染的示例。一个公开的示例提供了一种计算设备,该计算设备包括处理设备和存储设备,该存储设备存储指令,该指令由处理设备可执行以在第一本地过程中执行输出用于渲染和显示的数字内容的应用。该指令可执行以向第二本地或远程过程提供关于要由第二本地或远程过程渲染的对象的对象信息,从第二本地或远程过程接收对象的渲染,向显示器输出对象的渲染以显示对象,接收对对象进行的操纵,基于对对象进行的操纵向第二本地或远程过程提供已更新的对象信息,从第二本地或远程过程接收对象的已更新渲染,并且向显示器输出对象的已更新渲染以显示对象。

[0004] 另一示例提供了一种计算系统,该计算系统包括处理设备和存储设备,该存储设备存储指令,该指令由处理设备可执行以:从计算设备接收关于要被渲染的对象的对象信息;基于所接收的对象信息,渲染对象;向计算设备提供对象的渲染;从计算设备接收关于对对象进行的操纵的已更新的对象信息;基于所接收的关于对对象进行的操纵的已更新的对象信息来渲染对象;并且向计算设备提供对象的已更新渲染。

[0005] 另一示例提供了一种在计算设备上实施的方法,该方法包括:执行输出用于渲染和显示的数字内容的应用,在该应用的执行期间,标识远程地被渲染的数字内容的第一对象,并且还标识要本地地被渲染的数字内容的第二对象,向远程计算系统提供关于要由远程计算系统渲染的第一对象的对象信息,从远程计算系统接收第一对象的渲染,本地地渲染第二对象以生成第二对象的渲染,并且合成第一对象的渲染和第二对象的渲染,以用于作为合成帧显示。

### 附图说明

[0006] 图1A和图1B描绘了示例使用环境,其中计算设备合成本地地被渲染的和远程地被渲染的数字内容,以用于作为合成帧显示。

[0007] 图2是图示示例混合渲染系统的框图。

[0008] 图3是图示准备用于渲染的内容的示例方法的流程图。

[0009] 图4是图示基于为操纵远程地被渲染的对象而做出的输入来更新远程地被渲染的对象的显示的示例方法的流程图。

[0010] 图5是图示基于从计算设备所接收的信息来渲染对象的示例方法的流程图。

[0011] 图6是图示示例计算系统的框图。

## 具体实施方式

[0012] 计算设备上的处理硬件可能会限制计算设备的渲染能力。例如,复杂模型的渲染可能受到时钟速度、可用存储器的量、存储器带宽和/或计算设备本地的图形处理单元(GPU)或中央处理单元(CPU)的其他特性的约束。特别地,高度移动的设备,诸如手机和不受束缚的头戴式显示设备(例如,其未经由有线连接而连接到另一计算设备)可能没有足够的GPU和/或CPU能力来渲染具有可接受性能的复杂场景。

[0013] 可以采取各种方法来缓解这样的问题。例如,计算设备可以被配置为对复杂模型进行抽取(decimate)以简化用于渲染的模型。然而,抽取从模型中移除细节,这可能会损害模型的期望视觉准确性。作为另一种方法,代替于对模型进行抽取,计算设备可以被配置为远程地运行应用,以使得高能的远程计算系统渲染用于应用的所有数字内容并将所渲染的图像(例如作为视频流)传送给计算设备。所渲染的图像可以包括对应模型的完整细节。然而,这样的解决方案将所有应用逻辑转移到远程计算系统。结果,如果计算设备失去网络连接性,则计算设备可能无法显示远程地被渲染的内容,直到重新建立网络连接为止。此外,远程地被渲染的内容的网络传送对于时延敏感的内容来说可能太慢,诸如当所显示的模型被操纵时。

[0014] 除了上述问题之外,远程地渲染用于应用的所有内容还会造成各种其他限制。例如,远程地渲染所有内容的系统可能不向客户端设备提供本地渲染能力、输入能力、硬件访问等,并且因此不允许应用开发者指定用于本地渲染而不是远程渲染的内容。例如,对于希望使用特定游戏引擎和/或用户接口框架来编写应用代码的开发人员,对于时延敏感的用例(例如与真实世界对象相关的墨迹书写、动画或其他内容等),以及对于被配置为离线运行的针对特定用例按需集成远程地被渲染的内容的应用,执行本地渲染的能力可能是相关的。

[0015] 因此,公开了可以经由混合渲染系统向客户端计算设备提供这样的能力的示例,该混合渲染系统允许应用开发者包括用于生成远程地被渲染的内容和本地地被渲染的内容两者以合成为本地地所显示的图像的的逻辑。例如,诸如计算机辅助设计模型的更复杂的内容可以在远程计算系统上被渲染并且被流式传输给执行生成和/或显示模型的应用的客户端设备。客户端设备可以利用诸如轻量级、低时延的内容(例如用户接口/用户体验元素、墨迹书写、与真实世界对象(诸如关节手)相关的内容等)的本地地被渲染的内容来增强远程地被渲染的内容。客户端设备还可以执行深度校正融合以提供远程地被渲染的和本地地被渲染的内容之间的适当遮挡关系。此外,远程计算系统可以暴露远程场景图API,客户端设备可以使用该API来与远程地被渲染的内容交互—例如通过更新材料、操纵对象转换、将新内容注入到远程场景中以及执行光线投射查询。这可能有助于利用远程计算系统的处理能力,而无需在云中运行整个应用。通过使用远程过程调用(RPC)机制来与远程地被渲染的内容交互,混合渲染系统向客户端设备提供对远程地被渲染的场景元素(取向、定位、缩放、材料属性等)的控制,并且因此准许所有应用逻辑本地地运行在客户端设备上。此外,在网络连接性丢失或不可用的情况下,指定用于远程地渲染的模型也可以本地地被渲染(例如,使用替代的或抽取的内容)。

[0016] 图1A描绘了示例使用环境100,其中计算设备102提供与远程渲染的内容的交互能力,并且还利用本地渲染的内容来增强远程渲染的内容。在图1A的示例中,计算设备102包

括混合现实显示设备,该混合现实显示设备被配置为经由透视显示器将虚拟现实图像(例如全息图像)与真实世界图像相混合。在其他一些示例中,计算设备102可以包括不同的移动计算设备(虚拟现实显示设备、蜂窝电话、平板电脑等)、膝上型计算机、个人计算机、游戏控制台或任何其他合适的计算设备。虽然在该示例中被描绘为居住环境100,但是本文描述的示例可以在任何其他合适的使用环境中被实现。其他使用环境的示例包括其他室内空间,诸如商业空间(例如办公室、学校、仓库等)和室外空间(例如建筑工地、公园等)。

[0017] 返回图1A,计算设备102显示包括本地地被渲染的手104、远程地被渲染的狗106和远程地被渲染的灯108的合成帧。在渲染和显示这些对象之前,在计算设备102上执行的应用请求加载场景的模型并且在模型内标识要本地地被渲染的内容和要远程地被渲染的内容。例如,计算设备102请求为其加载模型的所选应用可以包括针对每个对象的条件逻辑(例如,基于网络带宽、对象的复杂性等)或者关于如何渲染对象的绝对逻辑。

[0018] 针对被标识用于远程渲染的每个对象106、108,计算设备102经由网络112向远程渲染系统110提供对象信息。对象信息可以包括关于要远程地被渲染的(多个)对象106、108的任何合适的信息。在一些示例中,对象信息包括三维模型(例如,包括每个所显示像素的多个多边形的高多边形网格、基于体素的模型等),其包括每个对象106、108的几何形状。对象信息还可以包括用于每个对象106、108的材料(例如纹理)。此外,对象信息还可以包括描述场景内的每个对象106、108的位置和/或取向的场景信息,其可以是绝对的或者是相对于场景中的另一对象的。在其他一些示例中,对象信息包括加载和/或更新由远程渲染系统110存储的现有模型的请求,例如,在应用的开发期间所存储的模型。

[0019] 远程渲染系统110可以在对象106、108的要针对其而被渲染和被显示的应用的执行之前或期间(从计算设备102或另一设备或过程)接收对象信息。当远程渲染系统110在运行时之前接收到对象信息时,远程渲染系统110可以存储对象信息以供在运行时期间检索,例如,在接收到加载和/或更新包含对象信息的模型的请求后检索。在任何实例中,远程渲染系统110将从计算设备102接收到的信息存储在由计算设备102可访问的场景图或其他合适的的数据结构中。

[0020] 计算设备102还可以向远程渲染系统提供关于用户114的感测或预测的姿势(手、头等)和/或眼睛注视方向的信息。在一些示例中,计算设备102可以基于感测到的姿势/注视来确定虚拟相机位置,并且可以向远程渲染系统110提供所确定的虚拟相机位置以用于输入到渲染管线中。在其他一些示例中,计算设备102可以向远程渲染系统110提供关于感测到的姿势/注视方向的传感器数据(例如,原始的或经处理的传感器数据),远程渲染系统110可以使用这些传感器数据来确定用于渲染的虚拟相机变换。远程渲染系统110将对象信息和虚拟相机位置输入到渲染管线中,并且渲染每个对象106、108。针对每个对象106、108,渲染管线输出对象的渲染,其可以采取每个被渲染对象的经渲染的像素或任何其他合适表示的形式。对象的渲染包括颜色缓冲和深度缓冲。远程渲染系统110然后将颜色缓冲和可选的深度缓冲(或深度数据的其他表示)作为视频数据(例如经压缩的视频流)提供给计算设备102。

[0021] 例如,基于真实的手在计算设备102的图像传感器的视场内的感测定位,并且可选地,基于从远程渲染系统110所接收的深度数据,计算设备102可以本地渲染手104。以这种方式,与远程地被渲染的对象相比,本地地被渲染的手104的表示可以更接近实时地被渲

染。为了帮助补偿网络时延,计算设备102可以重投影从远程渲染系统110接收的渲染。这可以包括:例如,基于最近所获取的姿势/注视数据,重投影从远程渲染系统110接收的渲染的颜色缓冲以及可选地深度缓冲。除了重投影从远程渲染系统110接收的渲染之外或者备选地,任何其他合适的时延缓解和图像稳定过程可以被执行。

[0022] 如下文更详细描述,计算设备102将狗106和灯108的渲染与手104的渲染合成以用于作为合成帧显示。为了帮助稳定场景并补偿本地地被渲染的对象的渲染/远程地被渲染的对象的渲染的重投影与已合成图像的显示之间的时间期间的任何移动,计算设备可以在输出用于显示的合成帧之前执行后期阶段重投影(LSR)过程,如下面更详细的描述的那样。

[0023] 一旦场景被渲染和被显示,用户可能希望与所显示的内容交互。在图1A的示例中,用户114可以与所显示的灯108交互(例如移动灯、开灯/关灯等),并且还可以与所显示的狗106交互。当用户114与远程地被渲染的对象交互时,计算设备102执行关于对对象进行的操纵的远程API的调用,并且从远程渲染系统110接收已更新的渲染,这在一些示例中可以在逐帧的基础上发生。在图1B中,计算设备102向远程渲染系统110提供关于为操纵狗106的显示而做出的挥手手势116的输入。作为响应,计算设备从远程渲染系统110接收狗106的已更新的渲染,并且输出已更新的渲染以供显示。如图1B中所示,响应于挥手手势输入116,狗106被渲染和被显示为站立。

[0024] 图2描绘了包括计算设备202和远程渲染系统204的示例混合渲染系统200。适合作为计算设备202的示例计算设备包括移动计算设备(增强现实显示设备、虚拟现实显示设备、蜂窝电话、平板电脑等)、膝上型计算设备、台式计算设备、游戏控制台等。远程渲染系统204可以包括第二计算系统,第二计算系统具有任何数目和类型的具有足够渲染能力来实现本文所描述的混合渲染过程的(多个)计算设备。在一些示例中,远程渲染系统204包括一个或多个服务器计算机,诸如被托管在云计算环境中的那些服务器计算机。在其他一些示例中,远程渲染系统204包括一个或多个服务器计算机、台式计算机和/或在计算设备202的使用环境本地的(多个)其他计算设备。

[0025] 计算设备202包括客户端应用206,客户端应用206输出用于渲染和显示的数字内容。客户端应用的示例包括用于游戏、计算机辅助设计/可视化、以及指导/教育的应用。客户端应用206被配置为与游戏引擎208或其他渲染引擎进行接口。任何合适的渲染引擎都可以被用作游戏引擎208,包括独立于远程渲染服务而存在的第一方和第三方软件。本文中所使用的术语“远程渲染服务”是指使用远程地被托管的渲染引擎来渲染像素以供传输给客户端计算设备(例如计算设备202)用于本地显示并且支持如本文所描述的混合渲染的服务。

[0026] 关于如何在客户端应用206内渲染数字内容的决策是对应用开发者可用的开发时间决策。例如,应用开发者可以在场景图或其他数据结构的节点内包括关于如何渲染由场景图的节点表示的对象的逻辑。在一些示例中,客户端应用206可以包括用于渲染对象的条件逻辑(例如,如果网络时延大于或等于阈值则远程地渲染对象)。客户端应用206还可以包括用于渲染对象的绝对逻辑(例如本地地渲染用户接口、远程地渲染高多边形模型等)。在任何实例中,客户端应用206的逻辑可以被用来控制本地地被渲染的场景和远程地被渲染的场景。

[0027] 继续图2,客户端应用206还被配置为与远程渲染(RR)组件库210进行接口,RR组件

库210可以针对由计算设备202使用的游戏引擎208而被专门开发。RR组件库210被配置为与RR客户端库212进行接口,RR客户端库212用作远程渲染服务的客户端侧端点。如下文更详细描述,RR客户端库212可以被实现为轻量级客户端侧场景图API,其将API调用转换为被发送给远程渲染系统204的消息。

[0028] 远程渲染系统204包括RR服务器库214,RR服务器库214用作远程渲染服务的远程端点。RR服务器库214被配置为经由网络从计算设备202的RR客户端库212接收对象信息和已更新的对象信息(例如,场景图更新)。远程渲染系统204可以经由RR服务器库214暴露轻量级场景图API,用于在计算设备202与远程渲染系统204之间提供并且更新场景图数据。场景图数据可以包括例如网格、材料属性(例如,用于基于物理的着色和/或Phong着色)、三维文本、光照、色调映射、节点变换和/或特定于使用的特征(例如切割平面)。在一些示例中,场景图数据也可以被高速缓存在计算设备202上,这可以帮助计算设备202在运行时期间快速访问场景图数据,例如当到远程渲染系统204的网络连接不可用时。

[0029] 场景图可以被用于例如指导远程渲染系统204加载特定场景、向计算设备202通知远程渲染系统204所加载的帧场景、允许计算设备202操纵远程场景图中的数据(例如,指定对象的动画/移动)、允许计算设备202在远程场景中创建新节点并且操纵节点、和/或允许计算设备202执行异步查询,诸如到要由远程渲染系统204执行的远程场景图中的光线投射。

[0030] 在一些实例中,场景图可以基于计算设备202所使用的工具/引擎(例如游戏引擎208)而被包装在各种类中。此外,场景图API可以被投影到诸如C、C#和JavaScript的目标语言中。这可以向任意第三方应用暴露远程渲染系统204的能力,而独立于这样的应用所使用的游戏/渲染引擎。除了混合现实和虚拟现实应用之外或者备选地,远程渲染系统204还可以在web和移动应用中被利用。

[0031] 远程渲染系统204还包括应用远程处理服务216,应用远程处理服务216被配置为向渲染引擎218提供输入,诸如从计算设备202所接收的已更新的对象信息(例如,场景图数据),并且向计算设备202提供对象的渲染。在一些示例中,应用远程处理服务216可以采取独立于远程渲染服务而存在的第一方或第三方软件的形式。基于所接收的已更新的对象信息,渲染引擎218将已更新的对象信息输入到渲染管线中并且渲染对象以生成对象的渲染。在一些实例中,渲染管线可以支持自定义着色器。

[0032] 渲染引擎218可以经由一个或多个处理设备来被实现。在图2中所示的示例中,渲染引擎218经由被配置为渲染高多边形内容的GPU218a来被实现。渲染引擎218也可以经由向GPU提供渲染任务的CPU(未示出)来被实现。此外,在一些示例中,渲染引擎可以包括其他处理设备,诸如其他GPU 218b、218c。在其他一些示例中,可以使用任何其他合适数目和/或类型的处理设备来实现渲染引擎218,包括以下参考图6描述的示例。应当理解,在一些示例中,现成的商业引擎可以被用作渲染引擎218。

[0033] 混合渲染系统200可以实现各种策略以处理大场景图并且绘制调用计数。例如,远程渲染系统204可以构建网格或其他模型,以使得材料和转换数据可以从GPU侧缓冲中被拉取,并且还可以利用面向图形的API套件(诸如DirectX12(DX12),可从华盛顿州雷德蒙德的微软公司获得)和间接绘制,以帮助减少CPU侧绘制调用开销。对于与大场景图相关联的客户端应用206,远程渲染系统204可以参考场景图的紧凑表示或者可以修改场景图的架构以

便以面向数据的方式来操作(例如实体组件系统)。此外,为了帮助在运行时期间更新场景图的潜在重要子集,混合渲染系统200可以采用转换压缩来在计算设备202与远程渲染系统204之间进行数据的网络传送。

[0034] 应用远程处理服务216将来自渲染引擎218的渲染作为包括颜色/帧缓冲的视频数据(例如经压缩的视频流)提供给RR客户端库212。在一些实例中,有限的Wi-Fi或其他网络带宽可能会限制可以被传送到计算设备202的颜色/帧缓冲数据的量。因此,应用远程处理服务216可以使用诸如H.265的视频压缩标准来向计算设备202流式传输颜色缓冲。例如经由高效视频编码或被配置为准确压缩深度数据的自定义编解码器,应用远程处理服务216还可以向计算设备202流式传输深度缓冲或深度数据的其他表示。

[0035] 在发送/接收经压缩的深度缓冲的实现中,RR客户端库212接收并且重构用于远程地被渲染的内容的深度缓冲。在一些示例中,计算设备202可以在利用本地地被渲染的内容来增强远程地被渲染的内容之前实现一个或多个时延缓解策略。例如,当在计算设备202上消耗和使用关于手和头部/眼睛跟踪的最近获取的信息时,计算设备202可以使用用于渲染远程地被渲染的图像的相机变换来渲染图像。作为另一示例,计算设备202可以利用最近获取的姿势/注视数据来重投影远程地被渲染的内容的颜色缓冲和深度缓冲。

[0036] 混合渲染系统200还可以实现各种其他时延缓解策略。作为一个示例,远程渲染系统204可以通过对颜色缓冲执行色度子采样以减少被发送的数据量来减少时延。远程渲染系统204还可以执行多GPU渲染,诸如基于图块和/或基于几何形状/集群的多GPU渲染。在示例基于几何形状/集群的多GPU渲染过程中,模型的几何体被划分为集群并且跨多个GPU被分配,并且每个GPU渲染其相应被指派的几何形状并且通过PCIe总线向主GPU(例如图2中的218a)提供颜色和深度缓冲,主GPU执行深度感知的缓冲合并和视频编码。多GPU渲染可以帮助支持以足够的帧速率(例如60Hz)来渲染大型网格(例如100M+多边形),并且还可以通过减少总体渲染时间来减少时延。作为其他示例时延减少和带宽减少策略,应用远程处理服务216可以以120Hz来对立体图像的编码进行交织,或者可以不是针对完整的第二图像而是针对第二只眼睛的经重投影的帧和原始帧之间的差量进行编码。此外,与远程渲染所有内容的系统相比,混合渲染本身可以帮助减少时延,因为至少一些内容可以在计算设备202上本地地被渲染而不是通过网络被传送。

[0037] 一旦计算设备202将从远程计算系统204接收的深度缓冲解压缩(在发送/接收经压缩的深度缓冲的实现中)或者以其他方式对所接收的深度信息执行任何必要的处理,计算设备202就可以本地地渲染具有深度缓冲或绑定到渲染管道的其他深度信息的对象。计算设备202经由任何合适的帧缓冲合并过程合成远程渲染和本地渲染的内容以用于作为合成帧显示。在一些示例中,计算设备执行深度校正融合以提供场景的深度校正合成。深度校正融合可以包括例如:对远程地被渲染的内容的深度缓冲和本地地被渲染的内容的深度缓冲执行深度测试(例如,z测试),并基于深度测试来进行合成。这样的融合可以在多种场景中是有用的,诸如用于墨迹特征、客户端平板、针对特定深度坐标而本地地被渲染的文本注释(例如,以保持文本易读性),以及用于直接操纵和用户接口特征—其中远程地被渲染的内容展现出相对于本地地被渲染的内容的深度校正关系。在其他一些示例中,本地地被渲染的内容相对于远程地被渲染的内容驻留于固定定位(例如,驻留在远程地被渲染的背景之上的用户接口),并且计算设备202可以在没有深度测试的情况下合并相应的帧。

[0038] 在合成本地地被渲染和远程地被渲染的内容以用于作为合成帧显示之后,计算设备202可以执行后期阶段重投影过程以进一步稳定合成帧。在一些示例中,计算设备202可以仅重投影合成帧的颜色缓冲。在其他一些示例中,重投影也可以基于合成帧的深度缓冲来被执行。重投影可以基于任何合适的信息而被执行,诸如被编码在视频流中的运动矢量,或者来自驻留在计算设备202上的一个或多个运动传感器的运动传感器数据。计算设备202被配置为经由显示器220来输出合成帧,显示器220可以与计算设备202集成在计算设备202的共同外壳或在外围设备中。任何合适的显示器可以被使用,包括透视显示器(例如基于波导或棱镜的显示器)和不透明显示器。

[0039] 在图1A-图1B和图2的示例中,远程渲染系统(分别为110和204)从单个计算设备(分别为102和202)接收对象信息和已更新的对象信息。在其他一些示例中,远程渲染系统可以服务于对同一场景有贡献的多个计算设备。例如,远程渲染系统可以针对单独的增强现实显示设备的用户渲染场景图,其中每个用户可以贡献对同一场景图的更新。在更具体的示例中,远程渲染系统可以基于场景内每个用户的定位而针对每个用户从不同视角渲染场景内的同一对象。此外,在一些示例中,多个网络可访问的云服务器可以协同渲染场景。

[0040] 本文所描述的混合渲染系统200可以帮助以适当高的帧速率来渲染包括多个个体对象的高多边形场景(例如,计算机辅助设计模型、构造/制造文档等)。相比之下,不支持混合渲染的本地渲染引擎可能无法以适当高的帧速率针对这样的应用来进行渲染。虽然在图2中被示为单独的计算系统,但是在一些实例中,计算设备202本身也可以执行远程渲染系统204(例如,在同一系统上在不同过程中运行)。在一些这样的实例中,本文所描述的混合渲染示例可以被用于在计算设备202上所实施的本地场景,诸如合成以独立相位和帧速率运行的许多3D应用。

[0041] 在一些实例中,数字内容可以在进入渲染管线之前被准备好用于渲染。图3描绘了可选地由远程渲染服务执行的示例方法300,其可以帮助准备用于混合渲染的数字内容。在一些示例中,方法300可以被实现为由混合渲染系统(诸如系统200)的处理设备可执行的被存储指令。

[0042] 在302处,方法300包括发起文件处理。在一些示例中,客户端设备(例如计算设备102和/或计算设备202)可以通过如上所述的场景图API将资产(例如作为包括诸如模型、材料等的个体文件在内的压缩档案)上传到云存储位置。在一些示例中,这可以经由建立在场景图API上的网站而被执行。在其他一些示例中,计算设备可以本地地发起文件处理。

[0043] 在304处,方法300包括通过确定在步骤302被上传的文件的高速缓存的版本是否对远程渲染服务可用来确定该文件是否是新文件。如果文件的高速缓存的版本可用,则方法300进行到320。如果文件的高速缓存的版本不可用,则方法300进行到306,在306处,远程渲染服务确定文件是否包括渲染引擎(例如,图2中的渲染引擎218)可读的本机文件格式。如果文件的格式是本机文件格式,则方法300进行到314,在314处,远程渲染服务将文件上传到云存储位置。

[0044] 如果远程渲染服务在306处确定文件的格式不同于本机文件格式,则方法300进行到308,在308处,文件被上传到云存储位置。在310处,方法300包括将所存储的文件的格式转换为本机文件格式。

[0045] 在312处,远程渲染服务将文件的本机文件格式版本放置在网络可访问的存储位

置。在316处,远程渲染服务处理本机文件格式的文件,以将本机文件格式转换为远程渲染文件格式,诸如二进制格式。在318处,远程渲染服务将文件的远程渲染文件格式版本放置在存储装置中,并且还返回被上传的内容的标识符(ID)(例如唯一模型ID),其可以通知客户端设备该文件被正确上传。

[0046] 在320处,客户端应用发送API调用(例如,经由图2中的RR客户端库212)以加载用于渲染的特定模型。在322处,远程渲染API(图2中的RR服务器库214)使用针对特定模型所返回的ID来加载该模型并且启动渲染管线。

[0047] 图4图示了用于实现混合渲染的示例方法400。方法400可以被实现为由客户端计算设备(诸如计算设备102和/或计算设备202)的处理设备可执行的被存储指令。

[0048] 在402处,方法400包括在第一本地过程中执行输出用于渲染和显示的数字内容的应用。在应用的执行期间,方法400可以包括标识要远程地被渲染的第一对象以及标识要本地地被渲染的第二对象。例如,计算设备可以基于由应用的应用开发者创作的条件逻辑和/或绝对逻辑来标识是本地地还是远程地渲染对象。

[0049] 在404处,方法400包括向第二本地或远程过程提供关于要由第二本地或远程过程渲染的对象的对象信息。提供对象信息可以包括提供包括对象的几何形状和材料的三维模型(例如,高多边形网格、基于体素的模型等),如406所指示。例如,计算设备可以经由场景图API提供用于远程渲染的模型,如图3中所述。提供对象信息还可以包括提供用于更新被数字渲染的内容的数据结构的请求,如408所指示。例如,计算设备可以发送API调用以与针对之前被上传的特定模型的远程场景图交互。客户端计算设备还可以提供远程计算系统可使用来渲染对象的任何其他合适的信息,诸如姿势/注视数据和/或已确定的虚拟相机定位。在一些示例中,提供对象信息可以包括向远程计算系统(例如,远程渲染系统110、204)提供对象信息,如409所指示。在其他一些示例中,提供对象信息可以包括向不同于第一本地过程的第二本地过程提供信息。

[0050] 在410处,方法400包括从第二本地或远程过程接收对象的渲染。在一些示例中,接收对象的渲染包括接收包括颜色缓冲的视频数据,如412所指示。这样的视频数据还可以包括深度的表示(例如深度缓冲或其他深度数据),如414所指示。此外,计算设备可以接收经压缩的视频数据,其可以帮助缓解时延。在其他一些示例中,接收对象的渲染可以包括从第二本地或远程过程接收任何其他合适的的数据。

[0051] 在416处,方法400可以包括:例如,通过重投影(分别在412和414处)所接收的颜色缓冲和/或深度缓冲,重投影对象的渲染。重投影对象的渲染可以帮助缓解由时延引起的图像不稳定性(例如抖动)。

[0052] 在一些示例中,远程地被渲染的对象是第一对象,并且计算设备被配置为本地地渲染一个或多个其他对象。在更具体的示例中,计算设备可以被配置为本地地渲染图形用户接口元素、关节手和/或任何其他对时延敏感的对象,并且利用本地地被渲染的内容来增强第一对象。因此,在418处,方法400可以包括本地地渲染第二对象以生成第二对象的渲染。在一些示例中,本地地渲染第二对象可以包括基于感测或预测的头部姿势、眼睛注视方向和/或手姿势进行渲染。

[0053] 在420处,方法400包括合成第一对象的渲染和第二对象的渲染以用于作为合成帧显示。在一些示例中,合成包括对第一对象的深度缓冲和第二对象的深度缓冲执行深度测

试,并且基于深度测试来合成第一对象的渲染和第二对象的渲染,如422所指示。在其他一些示例中,合成包括将第二对象的渲染叠加在第一对象的渲染之上,从而利用本地地被渲染的内容来增强远程地被渲染的内容。在更具体的示例中,第二对象可以包括用户接口元素,并且合成可以包括将用户接口元素叠加在第一对象的渲染之上。

[0054] 在424处,方法400可以包括重投影合成帧的颜色缓冲以及可选地深度缓冲,这可以帮助进一步稳定合成帧的显示。计算设备可以基于任何合适的信息来执行这样的重投影,诸如深度信息、被编码在从远程计算系统所接收的视频数据中的运动矢量、或者来自驻留在计算设备202上的一个或多个运动传感器的运动传感器数据。在426处,方法400包括向显示器输出对象的渲染以显示该对象。

[0055] 在428处,方法400包括接收对远程地被渲染的所显示对象进行的操纵。在一些示例中,接收操纵可以包括接收(原始的和/或经处理的)传感器数据的输入,该传感器数据的输入指示计算设备的用户的头部姿势和/或眼睛注视方向的变化或预测变化,如430所指示。在其他一些示例中,接收操纵可以包括接收用户输入,如432所指示。适合于操纵对象的用户输入的示例包括手势输入、语音输入和按钮按压。在另一些示例中,对所显示对象进行的操纵的接收可以来自用户输入或传感器以外的其他源,诸如来自与所显示对象相关联的动画。

[0056] 在434处,方法400包括基于对对象进行的操纵向第二本地或远程过程提供已更新的对象信息。作为响应,第二本地或远程过程基于所接收到已更新的对象信息来渲染对象并且向计算设备提供对象的已更新渲染。因此,方法400包括,在436处,从第二本地或远程过程接收对象的已更新渲染。在438处,方法400包括向显示器输出对象的已更新渲染以显示对象。虽然在第一对象远程地被渲染和第二对象本地地被渲染的上下文中进行描述,但是应当理解,被渲染的场景可以包括多个远程地被渲染的对象和/或多个本地地被渲染的对象,这些对象根据本文公开的示例被处理和被合成。

[0057] 图5图示了基于从客户端计算设备所接收的信息来渲染对象的示例方法500。方法500可以被实现为由计算系统(诸如远程渲染系统110和/或远程渲染系统204)的处理设备可执行的被存储指令,或者被实现为独立于提供要被渲染的对象的过程而运行的本地过程。

[0058] 在502处,方法500包括从计算设备接收关于要被渲染的对象的对象信息。在一些示例中,接收对象信息可以包括接收请求以加载包括对象的数字模型,如504所指示。在其他一些示例中,接收对象信息可以包括接收远程API调用以更新针对对象的先前被上传的数据结构。

[0059] 基于所接收的对象信息,方法500包括在506处渲染对象。渲染对象可以使用任何合适的渲染管线来被执行。如508所指示,渲染对象可以包括基于被存储在针对数字模型的场景图中的信息来渲染,其中对象是场景图的节点。

[0060] 在510处,方法500包括向计算设备提供对象的渲染。提供对象的渲染可以包括提供至少包括颜色缓冲的视频数据,如512所指示。在更具体的示例中,视频数据可以包括颜色缓冲和深度表示(例如深度缓冲)。此外,在一些示例中,视频数据可以被压缩,如514所指示。

[0061] 在516处,方法500包括从计算设备接收关于对对象进行的操纵的已更新的对象信

息。接收已更新的对象信息可以包括接收指示计算设备的用户的姿势/注视的(原始的和/或经处理的)传感器数据,如518所指示。在一些示例中,指示用户的姿势/注视的信息可以包括由客户端计算设备确定的虚拟相机定位。接收已更新的对象信息可以附加地或备选地包括接收关于操纵对象的用户输入的信息,诸如手势输入、语音输入和/或按钮按压。在这样的实例中,用户输入可以以原始的或经处理的形式而被发送(例如,作为对基于所接收的用户输入而要对场景图做出的改变的规范)。在其他一些示例中,可以接收任何其他合适的信息,诸如关于与对象相关联的动画的信息。

[0062] 方法500还包括,在520处,基于所接收的已更新的对象信息来渲染对象。当已更新的对象信息是基于计算设备的用户的姿势/注视时,渲染对象可以包括基于姿势和/或注视来渲染,如522所指示。例如,远程计算系统可以基于所接收的已确定的虚拟相机定位来渲染对象。当已更新的对象信息是基于对计算设备的用户输入时,渲染对象可以包括基于用户输入进行渲染。在其他一些示例中,远程计算系统可以基于所接收的任何其他信息来渲染对象。在524处,方法500包括向计算设备提供对象的已更新渲染。

[0063] 在一些示例中,本文描述的方法和过程可以被绑定到一个或多个计算设备的计算系统。特别地,这样的方法和过程可以被实现为计算机应用程序或服务、应用编程接口(API)、库和/或其他计算机程序产品。

[0064] 图6示意性地示出了可以实施上述方法和过程中的一个或多个方法和过程的计算系统600的非限制性实施例。计算系统600以简化的形式被示出。计算系统600可以采取一个或多个个人计算机、服务器计算机、平板计算机、家庭娱乐计算机、网络计算设备、游戏设备、移动计算设备(例如头戴式显示设备)、移动通信设备(例如,智能手机)和/或其他计算设备的形式。

[0065] 计算系统600包括逻辑机器602和存储机器604。计算系统600可以可选地包括显示子系统606、输入子系统608、通信子系统610和/或图6中未示出的其他组件。

[0066] 逻辑机器602包括被配置为执行指令的一个或多个物理设备。例如,逻辑机器可以被配置为执行作为一个或多个应用、服务、程序、例程、库、对象、组件、数据结构或其他逻辑构造的一部分的指令。这样的指令可以被实现来执行任务、实现数据类型、变换一个或多个组件的状态、达成技术效果或者以其他方式达成期望结果。

[0067] 逻辑机器602可以包括被配置为执行软件指令的一个或多个处理器。附加地或备选地,逻辑机器可以包括被配置为执行硬件或固件指令的一个或多个硬件或固件逻辑机器。在各种示例中,这样的处理器可以被实现为一个或多个CPU和/或GPU。逻辑机器的处理器可以是单核或多核的,并且在其上所执行的指令可以被配置用于顺序、并行和/或分布式处理。逻辑机器的个体组件可选地可以分布在两个或更多个单独设备之中,这些设备可以位于远程和/或被配置用于协调的处理。逻辑机器的各方面可以由在云计算配置中所配置的远程可访问的联网计算设备来虚拟化和执行。

[0068] 存储机器604包括一个或多个物理设备,该一个或多个物理设备被配置为持有指令,该指令由逻辑机器可执行以实现本文描述的方法和过程。当这样的方法和过程被实现时,存储机器604的状态可以被变换—例如,以持有不同的数据。

[0069] 存储机器604可以包括可移除和/或内置设备。存储机器604可以包括光学存储器(例如,CD、DVD、HD-DVD、蓝光光盘等)、半导体存储器(例如,RAM、EPROM、EEPROM等)和/或磁

存储器(例如,硬盘驱动器、软盘驱动器、磁带驱动器、MRAM等)等等。存储机器604可以包括易失性、非易失性、动态、静态、读/写、只读、随机访问、顺序访问、位置可寻址、文件可寻址和/或内容可寻址的设备。

[0070] 应当理解,存储机器604包括一个或多个物理设备。然而,本文描述的指令的各方面备选地可以在有限持续时间内由不被物理设备持有的通信介质(例如,电磁信号、光信号等)传播。

[0071] 逻辑机器602和存储机器604的各方面可以被一起集成到一个或多个硬件逻辑组件中。这样的硬件逻辑组件可以包括现场可编程门阵列(FPGA)、程序和专用集成电路(PASIC/ASIC)、程序和专用标准产品(PSSP/ASSP)、片上系统级(SOC)和复杂可编程逻辑器件(CPLD)。

[0072] 术语“程序”和“引擎”可以被用来描述被实现为执行特定功能的计算系统600的方面。在一些情况下,程序或引擎可以经由执行由存储机器604持有的指令的逻辑机器602来实例化。应当理解,不同的程序和/或引擎可以从相同的应用、服务、代码块、对象、库、例程、API、函数等被实例化。同样,相同的程序和/或引擎可以由不同的应用、服务、代码块、对象、例程、API、函数等来实例化。术语“程序”和“引擎”可以包含可执行文件、数据文件、库、驱动程序、脚本、数据库记录等的个体或组。

[0073] 应当理解,如本文中所使用的,“服务”是跨多个用户会话可执行的应用程序。服务可以对一个或多个系统组件、程序和/或其他服务可用。在一些实现中,服务可以在一个或多个服务器计算设备上运行。

[0074] 当被包括时,显示子系统606可以被用来呈现由存储机器604持有的数据的视觉表示。该视觉表示可以采取图形用户接口(GUI)的形式。由于本文描述的方法和过程改变了存储机器所持有的数据,并且从而变换了存储机器的状态,因此显示子系统606的状态同样可以被转换以视觉地表示底层数据的变化。显示子系统606可以包括利用几乎任何类型的技术的一个或多个显示设备。这样的显示设备可以与逻辑机器602和/或存储机器604组合在共享外壳中,或者这样的显示设备可以是外围显示设备。

[0075] 当被包括时,输入子系统608可以包括一个或多个用户输入设备,诸如键盘、鼠标、触摸屏或游戏控制器,或者与该用户输入设备进行接口。在一些实施例中,输入子系统可以包括经选择的自然用户输入(NUI)组件或者与经选择的自然用户输入(NUI)组件进行接口。这样的组件可以是集成的或外围的,并且输入动作的转导和/或处理可以在板上或板外处理。示例NUI组件可以包括:用于语音和/或嗓音识别的麦克风;用于机器视觉和/或手势识别的红外、彩色、立体和/或深度相机;用于运动检测和/或意图识别的头部跟踪器、眼睛跟踪器、加速度计和/或陀螺仪;以及用于评估大脑活动的电场感测组件。

[0076] 当被包括时,通信子系统610可以被配置为将计算系统600与一个或多个其他计算设备通信耦合。通信子系统610可以包括与一种或多种不同通信协议兼容的有线和/或无线通信设备。作为非限制性示例,通信子系统可以被配置用于经由无线网络、或者有线或无线局域网或广域网进行通信。在一些实施例中,通信子系统可以允许计算系统600经由诸如互联网的网络向和/或从其他设备发送和/或接收消息。

[0077] 另一示例提供了一种计算设备,包括处理设备和存储设备,该存储设备存储指令,该指令由处理设备可执行以:在第一本地过程中执行输出用于渲染和显示的数字内容的应

用;在应用的执行期间,向第二本地或远程过程提供关于要由第二本地或远程过程渲染的对象的对象信息;从第二本地或远程过程接收对象的渲染;向显示器输出对象的渲染以显示对象;接收对对象进行的操纵;基于对对象进行的操纵,向第二本地或远程过程提供已更新的对象信息;从第二本地或远程过程接收对象的已更新渲染;并向显示器输出对象的已更新渲染以显示对象。在这样的示例中,指令可以附加地或可备选地可执行以通过提供包括对象的几何形状的三维模型来提供对象信息。在这样的示例中,指令可以附加地或备选地可执行以通过提供用于更新被数字渲染的内容的数据结构的请求来提供对象信息。在这样的示例中,对象的渲染可以附加地或备选地包括视频数据,该视频数据包括颜色缓冲。在这样的示例中,视频数据可以附加地或备选地包括深度的表示。在这样的示例中,对象可以附加地或备选地包括第一对象,并且指令可以附加地或备选地可执行以本地地渲染第二对象来生成第二对象的渲染,并且合成第一对象的渲染和第二对象的渲染以用于作为合成帧显示。在这样的示例中,从第二本地或远程过程所接收的对象的渲染可以附加地或备选地包括第一颜色缓冲,并且指令可以附加地或备选地可执行以在合成第一对象的渲染和第二对象的渲染之前,执行第一颜色缓冲的重投影。在这样的示例中,指令可以附加地或备选地可执行以在合成第一对象的渲染和第二对象的渲染之后,执行合成帧的颜色缓冲的重投影。在这样的示例中,从第二本地或远程过程所接收的第一对象的渲染可以附加地或备选地包括第一深度数据,并且第二对象的渲染可以附加地或备选地包括第二深度数据,并且指令可以附加地或备选地可执行以通过如下操作来合成第一对象的渲染和第二对象的渲染:对第一深度数据和第二深度数据执行z测试,并且基于z测试来合成第一对象和第二对象。在这样的示例中,指令可以附加地或备选地可执行以通过接收指示头部姿势和/或眼睛注视方向的变化传感器数据、和/或通过接收用于操纵对象的用户输入来接收对对象进行的操纵。

[0078] 另一示例提供了一种计算系统,包括处理设备和存储设备,该存储设备存储指令,该指令由处理设备可执行以:从计算设备接收关于要被渲染的对象的对象信息;基于所接收的对象信息,渲染对象;向计算设备提供对象的渲染;从计算设备接收关于对对象进行的操纵的已更新的对象信息;基于所接收的关于对对象进行的操纵的已更新的对象信息来渲染对象;并且向计算设备提供对象的已更新渲染。在这样的示例中,指令可以附加地或备选地可执行以通过提供包括颜色缓冲和深度缓冲的视频数据来向计算设备提供对象的渲染。在这样的示例中,指令可以附加地或备选地可执行以压缩视频数据并且将经压缩的视频数据提供给计算设备。在这样的示例中,对象信息可以附加地或备选地包括用于加载包括对象的数字模型的请求,并且指令可以附加地或备选地可执行以基于被存储在针对数字模型的场景图中的信息来渲染对象。在这样的示例中,指令可以附加地或备选地可执行以通过接收指示计算设备的用户的姿势的传感器数据来接收已更新的对象信息,并且指令可以附加地或备选地可执行以基于用户的姿势来渲染对象。

[0079] 另一示例提供了一种在计算设备上实施的方法,该方法包括:执行输出用于渲染和显示的数字内容的应用;在该应用的执行期间,标识要远程地被渲染的数字内容的第一对象,并且还标识要本地地被渲染的数字内容的第二对象;向远程计算系统提供关于要由远程计算系统渲染的第一对象的对象信息;从远程计算系统接收第一对象的渲染;本地地渲染第二对象以生成第二对象的渲染;并且合成第一对象的渲染和第二对象的渲染以用于

作为合成帧显示。在这样的示例中,第一对象的渲染可以附加地或备选地包括第一颜色缓冲和第一深度缓冲,该方法可以附加地或备选地包括,在合成第一对象的渲染和第二对象的渲染之前,执行至少第一颜色缓冲的重投影。在这样的示例中,该方法可以附加地或备选地包括,在合成第一对象的渲染和第二对象的渲染之后,执行重投影。在这样的示例中,执行重投影可以附加地或备选地包括基于用户的头部姿势来执行重投影。在这样的示例中,第一对象的渲染可以附加地或备选地包括第一深度数据,并且第二对象的渲染可以附加地或备选地包括第二深度数据,该方法可以附加地或备选地包括对第一深度数据和第二深度数据执行深度测试,并且基于深度测试来合成第一对象的渲染和第二对象的渲染。

[0080] 应当理解,本文描述的配置和/或方法本质上是示例性的,并且这些具体实施例或示例不应被认为是限制性的,因为许多变化是可能的。本文描述的特定例程或方法可以表示任何数目的处理策略中的一种或多种。因此,所图示和/或描述的各种动作可以按照所图示和/或描述的顺序来执行、以其他顺序来执行、并行执行、或者被省略。同样,上述过程的顺序可以被改变。

[0081] 本公开的主题包括本文公开的各种过程、系统和配置以及其他特征、功能、作用和/或属性的所有新颖和非显而易见的组合和子组合,以及任何及其所有等效体。

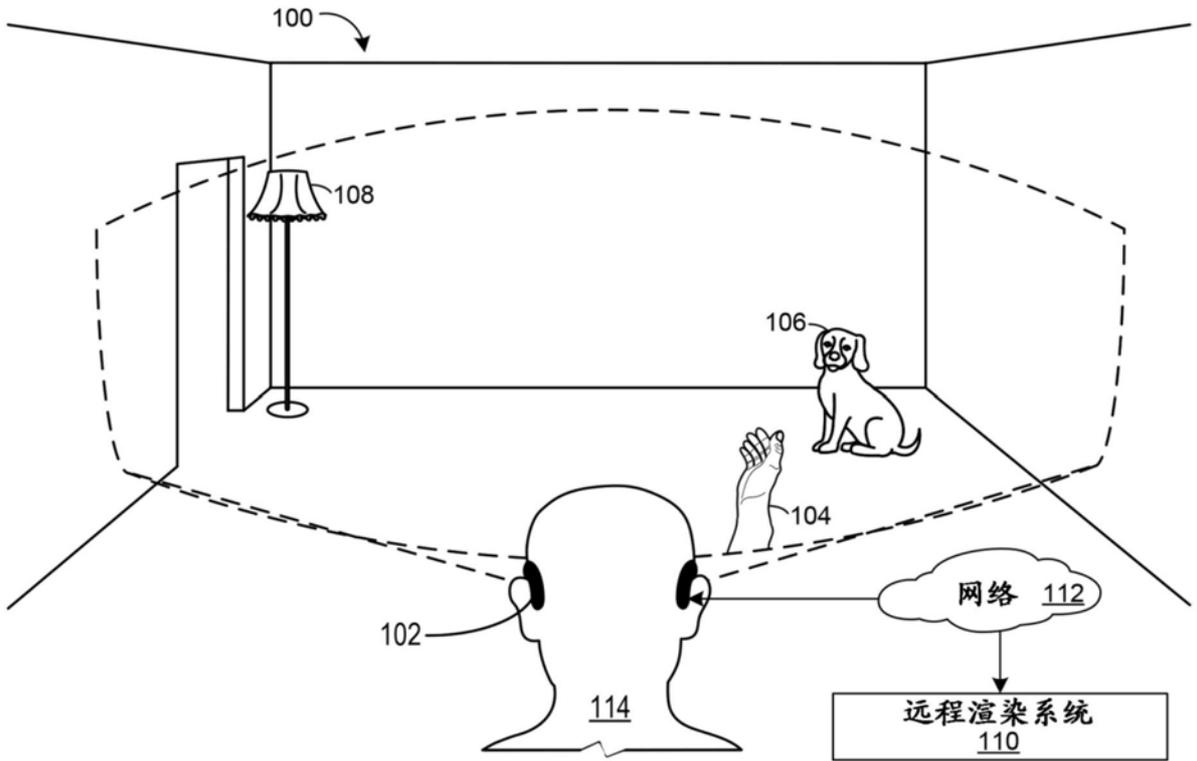


图1A

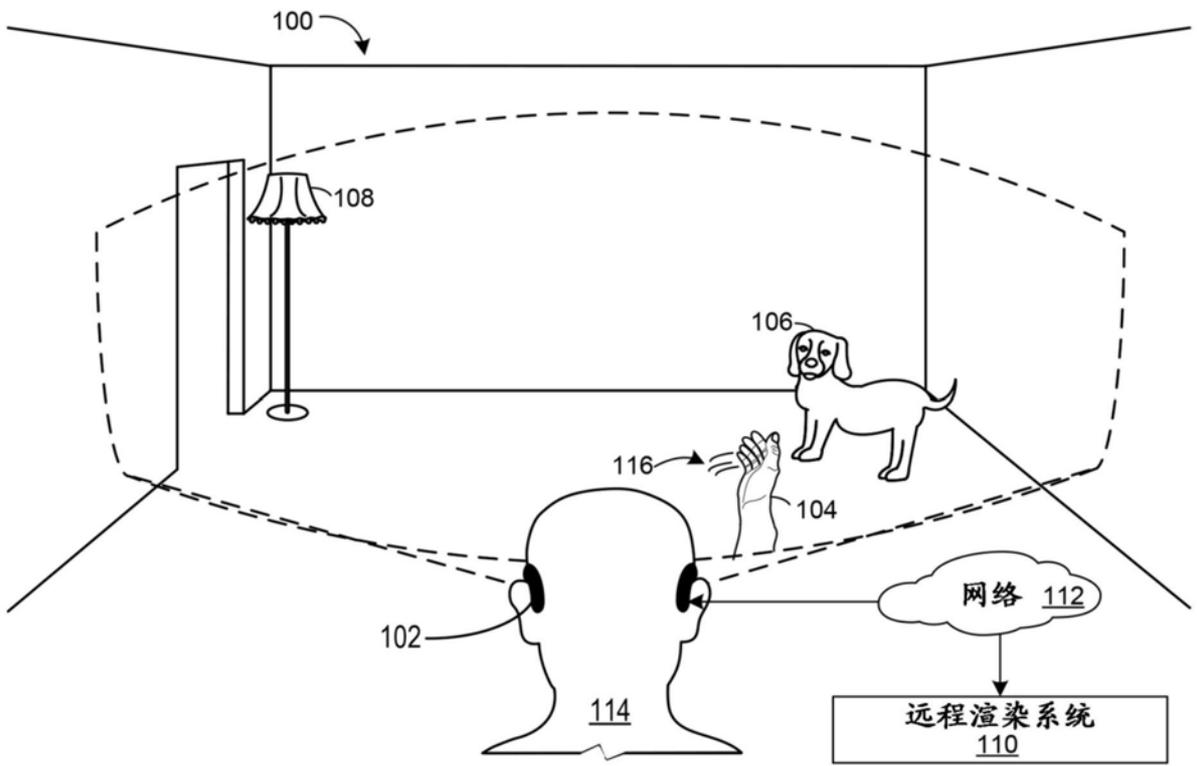


图1B

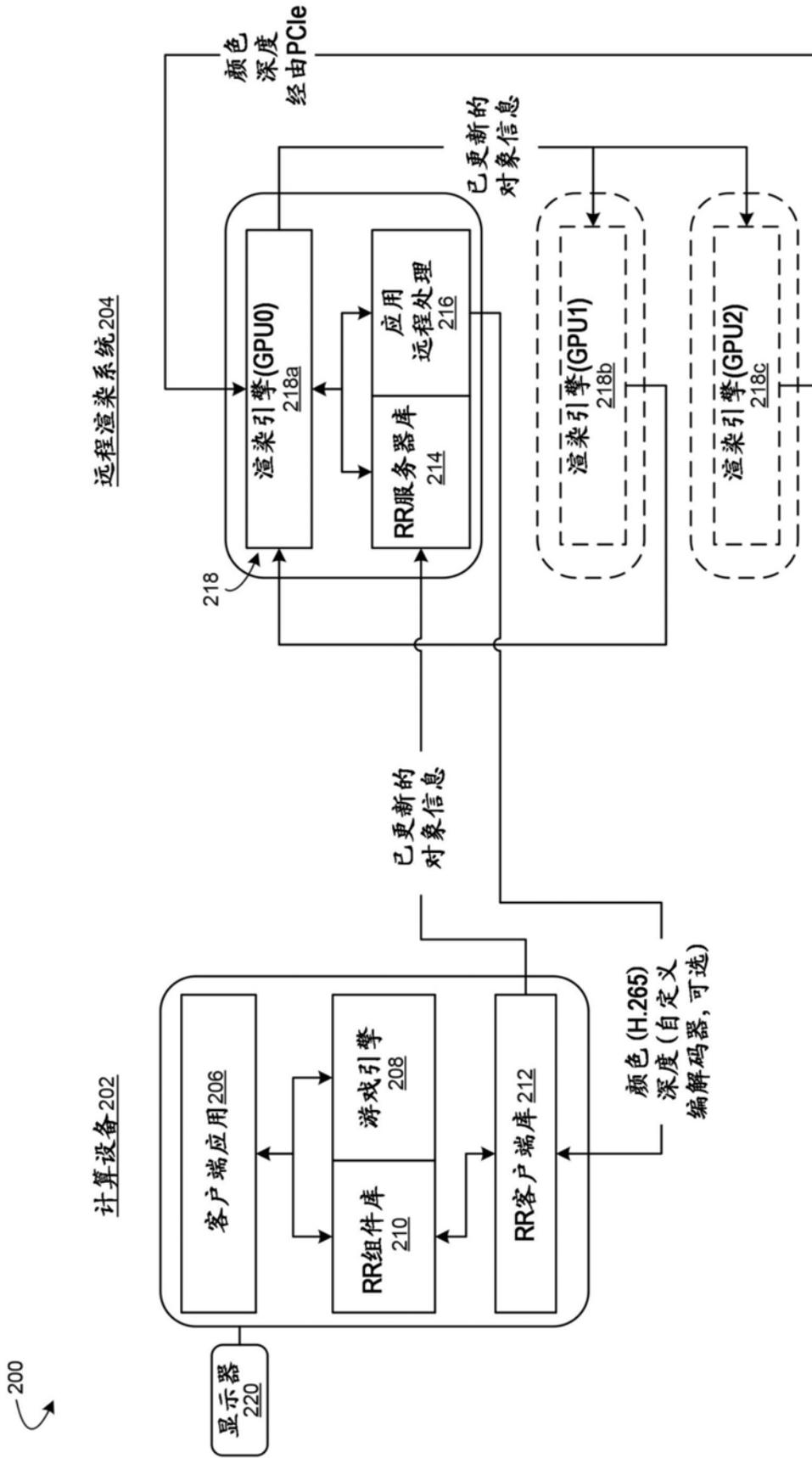


图2

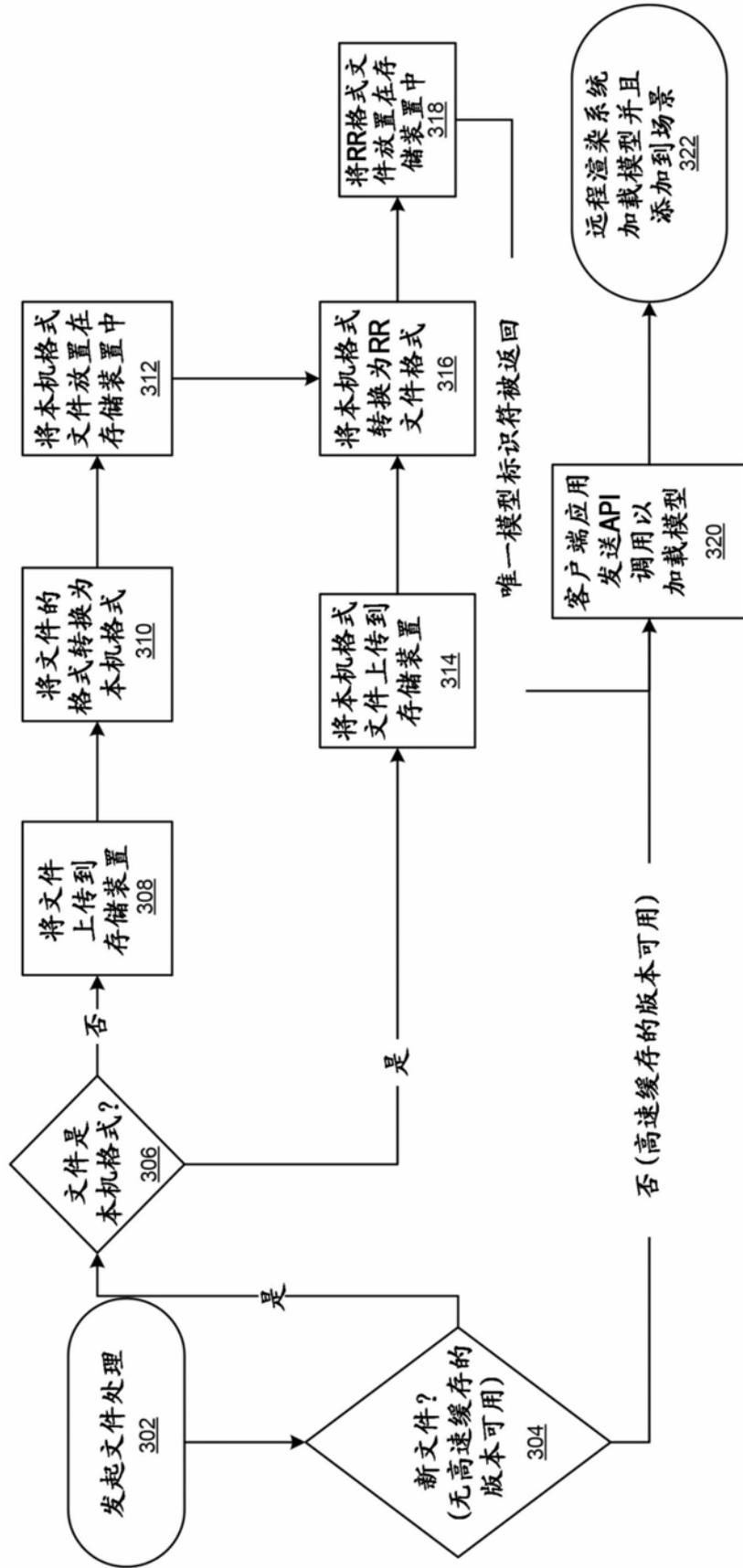


图3



图4

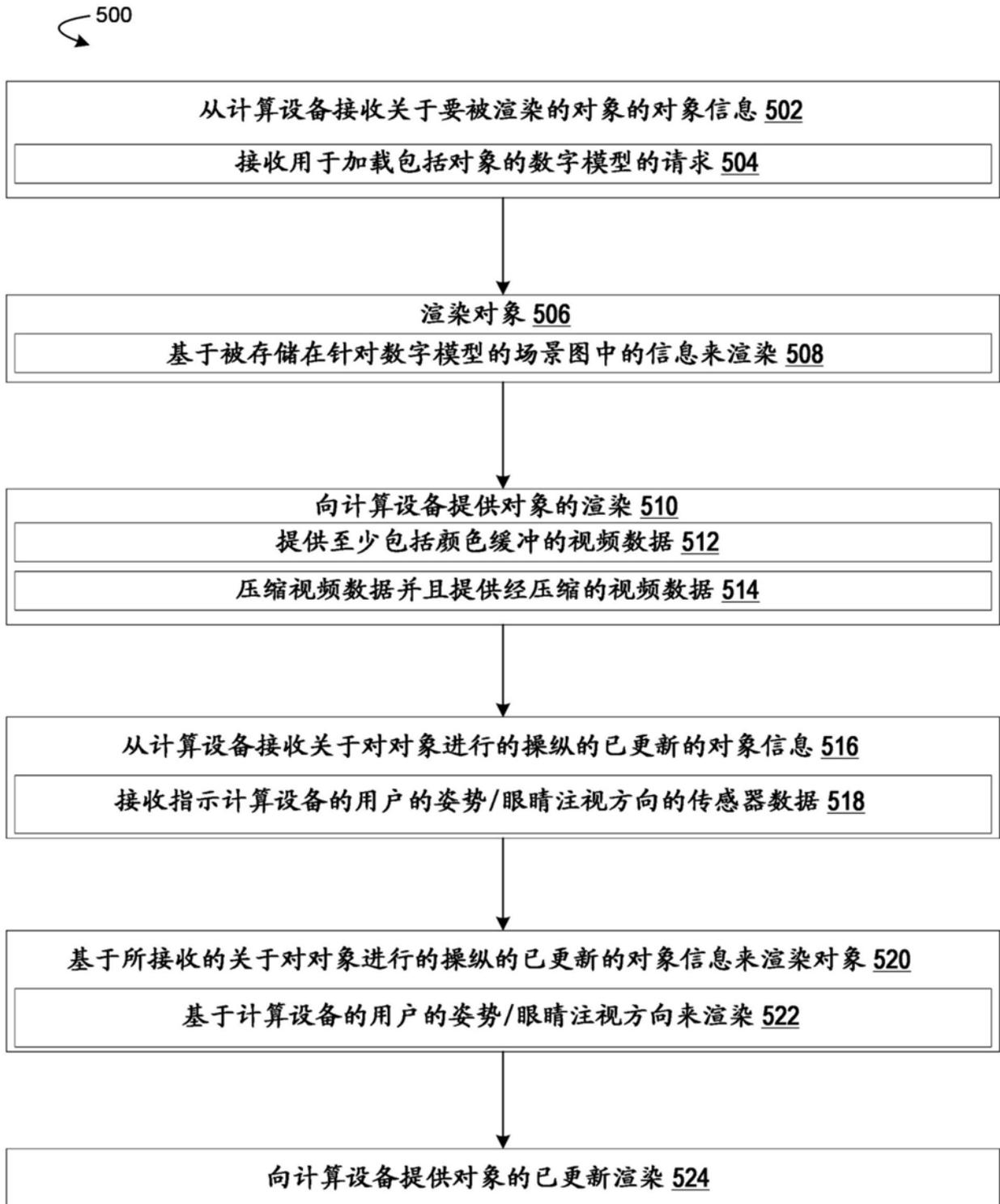


图5

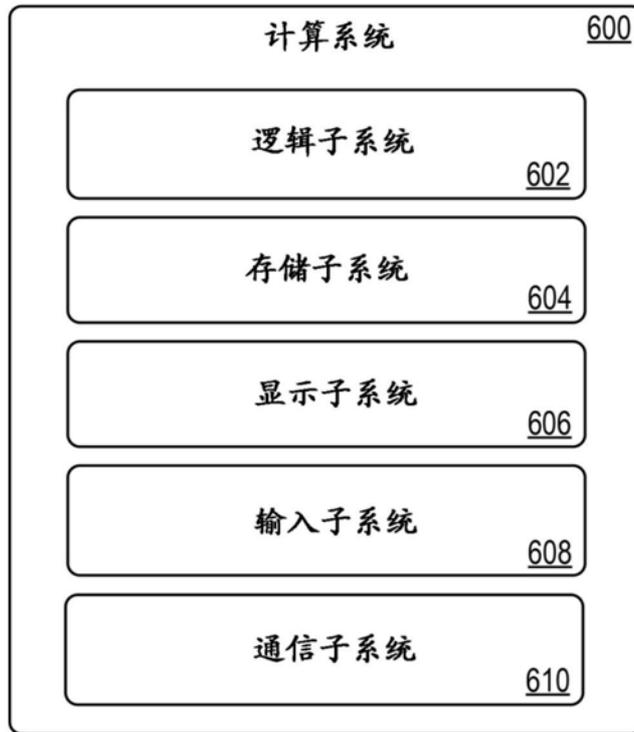


图6