

视频帧的渲染方法、装置、设备以及存储介质

本申请要求于 2021 年 08 月 31 日提交、申请号为 202111010058.0、发明名称为“视频帧的渲染方法、装置、设备以及存储介质”的中国专利申请的优先权，其全部内容通过引用结合在本申请中。

技术领域

本申请涉及计算机技术领域，特别涉及一种视频帧的渲染方法、装置、设备以及存储介质。

背景技术

随着云计算技术的成熟，用户可以通过云计算来实现终端难以完成的任务。例如，在游戏的领域，用户通过云计算技术可以玩终端本来无法流畅运行的游戏。与游戏相关的后台处理均由云游戏服务器来完成，终端只需向云游戏服务器发送控制信息，该控制信息用于控制游戏场景中的游戏对象。云游戏服务器能够基于该控制信息进行后台处理，得到视频帧，云游戏服务器将该视频帧发送给终端，终端显示该视频帧即可。

相关技术中，对于游戏场景中的某个游戏对象来说，该游戏对象的显示效果是技术人员在设计游戏时提前配置好的，比如，对于游戏场景中一辆汽车来说，该汽车的颜色和款式都是技术人员提前配置好的。

发明内容

本申请实施例提供了一种视频帧的渲染方法、装置、设备以及存储介质，在云应用中提供了一种二次渲染虚拟对象的功能。所述技术方案如下：

一方面，提供了一种视频帧的渲染方法，所述方法包括：

服务器获取第一终端对应的第一视频帧，所述第一视频帧是以目标虚拟场景中被控虚拟对象的视角，对所述目标虚拟场景进行渲染得到的视频帧，所述被控虚拟对象为所述第一终端控制的虚拟对象；

在所述第一视频帧显示有所述第一终端的目标虚拟对象的情况下，所述服务器基于所述第一终端的第一参数，对所述第一视频帧中的所述目标虚拟对象进行渲染，得到第二视频帧，所述目标虚拟对象为所述第一终端需要重新进行渲染的虚拟对象；

所述服务器将所述第二视频帧发送给所述第一终端，所述第一终端用于显示所述第二视频帧。

一方面，提供了一种视频帧的渲染系统，所述系统包括：

第一终端、第一服务器以及第二服务器，所述第一终端、所述第一服务器以及所述第二服务器之间通讯相连；

所述第一服务器用于以目标虚拟场景中被控虚拟对象的视角对所述目标虚拟场景进行渲染，得到所述第一终端对应的第一视频帧，将所述第一视频帧发送给所述第二服务器，所述被控虚拟对象为所述第一终端控制的虚拟对象；

所述第二服务器用于接收所述第一视频帧；

所述第二服务器还用于在所述第一视频帧显示有所述第一终端的目标虚拟对象的情况下，基于所述第一终端的第一参数，对所述第一视频帧中的所述目标虚拟对象进行渲染，得到第二视频帧；

所述第二服务器还用于向所述第一终端发送所述第二视频帧；

所述第一终端用于响应于接收到所述第二视频帧，显示所述第二视频帧。

一方面，提供了一种视频帧的渲染装置，所述装置包括：

第一视频帧获取模块，用于获取第一终端对应的第一视频帧，所述第一视频帧是以目标虚拟场景中被控虚拟对象的视角，对所述目标虚拟场景进行渲染得到的视频帧，所述被控虚拟对象为所述第一终端控制的虚拟对象；

渲染模块，用于在所述第一视频帧显示有所述第一终端的目标虚拟对象的情况下，所述服务器基于所述第一终端的第一参数，对所述第一视频帧中的所述目标虚拟对象进行渲染，得到第二视频帧，所述目标虚拟对象为所述第一终端需要重新进行渲染的虚拟对象；

发送模块，用于将所述第二视频帧发送给所述第一终端，所述第一终端用于显示所述第二视频帧。

一方面，提供了一种服务器，所述服务器包括一个或多个处理器和一个或多个存储器，所述一个或多个存储器中存储有至少一条计算机程序，所述计算机程序由所述一个或多个处理器加载并执行以实现所述视频帧的渲染方法。

一方面，提供了一种计算机可读存储介质，所述计算机可读存储介质中存储有至少一条计算机程序，所述计算机程序由处理器加载并执行以实现所述视频帧的渲染方法。

一方面，提供了一种计算机程序产品，该计算机程序产品包括程序代码，该程序代码被处理器执行时实现上述视频帧的渲染方法。

通过本申请实施例提供的技术方案，如果想要改变某个虚拟对象的显示效果，可以通过第一终端为该虚拟对象确定第一参数，基于该第一参数对视频帧进行二次渲染，得到第二视频帧，相较于第一视频帧，第二视频帧中显示的虚拟对象的显示效果也即是第一终端为该虚拟对象配置的显示效果。通过这样的技术方案，在云应用中提供了一种二次渲染虚拟对象的功能，用户能够快速、高效地调整虚拟场景中某个虚拟对象的显示效果，从而扩展了云应用的功能范围，提高了云应用的个性化，使得云应用传播更加广。

附图说明

为了更清楚地说明本申请实施例中的技术方案，下面将对实施例描述中所需要使用的附图作简单的介绍，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。

图1是本申请实施例提供的一种视频帧的渲染方法的实施环境的示意图；

图2是本申请实施例提供的一种视频帧的渲染方法流程图；

图3是本申请实施例提供的一种视频帧的渲染方法流程图；

图4是本申请实施例提供的一种界面示意图；

图5是本申请实施例提供的一种界面示意图；

图6是本申请实施例提供的一种视频帧的渲染方法流程图；

图7是本申请实施例提供的一种界面示意图；

图8是本申请实施例提供的一种视频帧的渲染装置的结构示意图；

图9是本申请实施例提供的一种服务器的结构示意图。

具体实施方式

为使本申请的目的、技术方案和优点更加清楚，下面将结合附图对本申请实施方式作进一步的详细描述。

本申请中术语“第一”“第二”等字样用于对作用和功能基本相同的相同项或相似项进行区分，应理解，“第一”、“第二”、“第n”之间不具有逻辑或时序上的依赖关系，也不对数量和执行顺序进行限定。

本申请中术语“至少一个”是指一个或多个，“多个”的含义是指两个或两个以上，例如，多个参照人脸图像是指两个或两个以上的参照人脸图像。

云应用：是将传统应用“本地安装、本地运算”的使用方式变为“即取即用”的服务，通过互联网或局域网连接并操控远程服务器集群，完成业务逻辑或运算任务的一种新型应用。

云应用运行在远程服务器集群中，但是云应用的界面显示在终端上，这样可以降低终端的运行成本，大大提高工作效率。

云游戏（Cloud Gaming）又可称为游戏点播（Gaming on Demand），是一种以云计算技术为基础的在线游戏技术。云游戏技术使图形处理与数据运算能力相对有限的轻端设备（Thin Client）能运行高品质游戏。在云游戏场景下，游戏并不在玩家游戏终端，而是在云服务器中运行，并由云游戏服务器将游戏场景渲染为视频音频流，通过网络传输给玩家游戏终端。玩家游戏终端无需拥有强大的图形运算与数据处理能力，仅需拥有基本的流媒体播放能力与获取玩家输入指令并发送给云游戏服务器的能力即可。

虚拟场景：是应用程序运行时显示（或提供）的虚拟场景。该虚拟场景可以是对真实世界的仿真环境，也可以是半仿真半虚构的虚拟环境，还可以是纯虚构的虚拟环境。虚拟场景可以是二维虚拟场景、2.5 维虚拟场景或者三维虚拟场景中的任意一种，本申请实施例对虚拟场景的维度不加以限定。例如，虚拟场景可以包括天空、陆地、海洋等，该陆地可以包括沙漠、城市等环境元素，用户可以控制虚拟对象在该虚拟场景中进行移动。

虚拟对象：是指在虚拟场景中的可活动对象。该可活动对象可以是虚拟人物、虚拟动物、动漫人物等，比如：在虚拟场景中显示的人物、动物、植物、油桶、墙壁、石块等。该虚拟对象可以是该虚拟场景中的一个虚拟的用于代表用户的虚拟形象。虚拟场景中可以包括多个虚拟对象，每个虚拟对象在虚拟场景中具有自身的形状和体积，占据虚拟场景中的一部分空间。

在一些实施例中，该虚拟对象是通过客户端上的操作进行控制的用户角色，或者是通过训练设置在虚拟场景对战中的人工智能（Artificial Intelligence, AI），或者是设置在虚拟场景中的非用户角色（Non-Player Character, NPC）。在一些实施例中，该虚拟对象是在虚拟场景中进行竞技的虚拟人物。在一些实施例中，该虚拟场景中参与互动的虚拟对象的数量是预先设置的，或者是根据加入互动的客户端的数量动态确定的。

以射击类游戏为例，用户能够控制虚拟对象在该虚拟场景的天空中自由下落、滑翔或者打开降落伞进行下落等，在陆地上中跑动、跳动、爬行、弯腰前行等，也可以控制虚拟对象在海洋中游泳、漂浮或者下潜等，当然，用户也可以控制虚拟对象乘坐虚拟载具在该虚拟场景中进行移动，例如，该虚拟载具可以是虚拟汽车、虚拟飞行器、虚拟游艇等，在此仅以上述场景进行举例说明，本申请实施例对此不作具体限定。用户也可以控制虚拟对象通过互动道具与其他虚拟对象进行战斗等方式的互动，例如，该互动道具可以是虚拟手雷、虚拟集束雷、虚拟粘性手雷（简称“虚拟粘雷”）等投掷类互动道具，也可以是虚拟机枪、虚拟手枪、虚拟步枪等射击类互动道具，本申请对互动道具的类型不作具体限定。

Android（安卓）容器：是将 Android 打包到容器镜像中，之后再通过标准容器镜像进行发布，载体可以是任意地支持 OCI（Open Container Initiative，开放容器标准）的容器外壳，所以可以很轻松地通过 k8s（kubernetes，一种容器编排引擎）来运维，借助于 k8s 强大的运维工具，在云上可以很轻易地部署到几千台服务器集群。

图 1 是本申请实施例提供的一种视频帧的渲染方法的实施环境示意图，参见图 1，该实施环境中可以包括第一终端 110、客户端服务器 120、第一服务器 130、游戏服务器 140 以及第二服务器 150。在一些实施例中，第一终端 110、客户端服务器 120、第一服务器 130、游戏服务器 140 以及第二服务器 150 为区块链系统中的节点，第一终端 110、客户端服务器 120、第一服务器 130、游戏服务器 140 以及第二服务器 150 之间相互传输的数据也存储在区块链上。

第一终端 110 通过无线网络或有线网络与客户端服务器 120 相连。可选地，第一终端 110 是智能手机、平板电脑、笔记本电脑、台式计算机、智能手表等，但并不局限于此。第一终端 110 安装和运行有支持虚拟场景显示的客户端。

客户端服务器 120 是为客户端提供服务的服务器，客户端服务器 120 是独立的物理服务

器, 或者是多个物理服务器构成的服务器集群或者分布式系统, 或者是提供云服务、云数据库、云计算、云函数、云存储、网络服务、云通信、中间件服务、域名服务、安全服务、分发网络(Content Delivery Network, CDN)、以及大数据和人工智能平台等基础云计算服务的服务器。第一终端 110 通过运行的客户端登录客户端服务器 120, 客户端服务器 120 提供与用户账号相关的服务, 比如提供用户账号验证的相关服务, 或者提供确定用户账号对应的云游戏时长的服务, 或者提供存储用户账号对应的个性化设置的服务, 本申请实施例对此不做限定。可以说, 客户端服务器 120 是连接着第一终端 110 上运行的客户端和第一服务器 130 的中间人。

第一服务器 130 是独立的物理服务器, 或者是多个物理服务器构成的服务器集群或者分布式系统, 或者是提供云服务、云数据库、云计算、云函数、云存储、网络服务、云通信、中间件服务、域名服务、安全服务、分发网络、以及大数据和人工智能平台等基础云计算服务的服务器。第一服务器 130 与显示虚拟场景相关的后台服务, 第一终端 110 与第一服务器 130 之间通过无线网络或有线网络相连, 第一终端 110 能够向第一服务器 130 发送控制信息, 该控制信息用于控制虚拟场景中的虚拟对象。第一服务器 130 基于该控制信息对虚拟场景进行渲染。第一服务器 130 与客户端服务器 120 之间通过无线网络或有线网络相连, 第一服务器 130 能够从客户端服务器 120 获取用户账号的相关信息, 基于该相关信息来进行游戏初始化等相关操作。可选地, 该第一服务器 130 为云服务器。

游戏服务器 140 是独立的物理服务器, 或者是多个物理服务器构成的服务器集群或者分布式系统, 或者是提供云服务、云数据库、云计算、云函数、云存储、网络服务、云通信、中间件服务、域名服务、安全服务、分发网络、以及大数据和人工智能平台等基础云计算服务的服务器。游戏服务器 140 与第一服务器 120 之间通过无线网络或有线网络相连, 游戏服务器 140 中存储有与游戏角色相关的信息, 比如游戏角色的好友、通讯录、游戏角色的等级、游戏角色的名称等相关信息。第一服务器 130 能够从游戏服务器 140 中获取与游戏角色相关的信息, 其中, 一个用户账号下可能对应于多个游戏角色, 客户端服务器 120 是用于存储用户账号相关的信息, 游戏服务器 140 是用于存储于游戏角色相关的信息。

第二服务器 150 是独立的物理服务器, 或者是多个物理服务器构成的服务器集群或者分布式系统, 或者是提供云服务、云数据库、云计算、云函数、云存储、网络服务、云通信、中间件服务、域名服务、安全服务、分发网络、以及大数据和人工智能平台等基础云计算服务的服务器。第二服务器 150 提供与虚拟场景二次渲染的相关服务, 第二服务器 150 与第一服务器 120 之间通过无线网络或有线网络相连, 第一服务器 120 能够将对虚拟场景进行渲染后的视频帧发送给第二服务器 150, 由第二服务器 150 对该视频帧进行二次渲染, 以实现用户的个性化处理。第二服务器 150 能够将二次渲染后的视频帧发送给第一终端 110, 由第一终端 110 对二次渲染后的视频帧进行显示。另外, 第二服务器 150 与客户端服务器 130 之间通过无线网络或有线网络相连, 第二服务器 150 能够从客户端服务器 130 上获取用户账号对应的个性化设置, 并基于该个性化设置对第一服务器 120 发送的视频帧进行二次渲染, 得到二次渲染后的视频帧。在一些实施例中, 第二服务器 150 集成在第一服务器 130 中, 第二服务器 150 的功能由第一服务器 130 上运行的系统来实现。可选地, 第二服务器 150 可以为场景管理服务器。

本领域技术人员可以知晓, 上述第一终端的数量可以更多或更少。比如上述第一终端仅为一个, 或者上述第一终端为几十个或几百个, 或者更多数量, 此时上述实施环境中还包括其他第一终端。本申请实施例对第一终端的数量和设备类型不加以限定。

在介绍完本申请实施例提供的视频帧的渲染方法的应用场景之后, 下面将结合上述实施环境, 对本申请实施例提供的视频帧的渲染方法的应用场景进行说明, 在下述说明过程中, 第一终端也即是上述实施环境中的第一终端 110, 客户端服务器也即是上述实施环境中的客户端服务器 120, 第一服务器也即是上述实施环境中的第一服务器 130, 游戏服务器也即是上

述实施环境中的游戏服务器 140，第二服务器也即是上述实施环境中的第二服务器 150。

本申请实施例提供的视频帧的渲染方法能够应用在各类云游戏的场景下，比如应用在第一人称射击类（First-person Shooting，FPS）游戏中，或者应用在第三人称射击游戏（Third-Personal Shooting，TPS）游戏中，或者应用在多人在线战术竞技游戏（Multiplayer Online Battle Arena，MOBA）中，或者应用在战棋游戏或者自走棋游戏中，本申请实施例对此不做限定。

以本申请实施例提供的视频帧的渲染方法应用在 FPS 游戏中为例，用户在第一终端上启动云游戏客户端，在云游戏客户端中登录用户账号，也即是，用户在云游戏客户端中输入用户账号和对应的密码，点击登录控件来进行登录。响应于检测到对登录控件的点击操作，第一终端向客户端服务器发送登录请求，登录请求中携带有用户账号和对应的密码。客户端服务器接收到登录请求之后，从登录请求中获取用户账号和对应的密码，对用户账号和对应的密码进行验证。客户端服务器对用户账号和对应的密码验证通过之后，向第一终端发送登录成功信息。第一终端接收到登录成功信息之后，向客户端服务器发送云游戏获取请求，云游戏获取请求中携带有用户账号。客户端服务器获取云游戏获取请求之后，基于云游戏获取请求中携带的用户账号进行查询，获取该用户账号对应的多个云游戏，将该多个云游戏的标识发送给第一终端，由第一终端将该多个云游戏的标识展现在云游戏客户端中。用户通过第一终端，在云游戏客户端中展示的多个云游戏的标识中选择想玩的 FPS 游戏的标识，也即是选择想玩的 FPS 游戏。用户在云游戏客户端中选择 FPS 游戏之后，第一终端向客户端服务器发送游戏启动指令，游戏启动指令中携带有用户账号、该 FPS 游戏的标识以及第一终端的硬件信息，其中，第一终端的硬件信息包括第一终端的屏幕分辨率、第一终端的型号等，本申请实施例对此不做限定。客户端服务器接收到该游戏启动指令之后，将游戏启动指令发送给第一服务器，第一服务器接收到该游戏启动指令之后，从该游戏启动指令中获取用户账号、该 FPS 游戏的标识以及第一终端的硬件信息。第一服务器基于第一终端的硬件信息对该 FPS 游戏进行初始化，以实现渲染得到的游戏画面与第一终端之间的匹配，向与该 FPS 游戏对应的游戏服务器发送该用户账号。游戏服务器接收到该用户账号之后，将与该游戏账号对应的信息发送给第一服务器，第一服务器基于与该游戏账号对应的信息启动该 FPS 游戏。在运行该 FPS 游戏的过程中，用户能够通过第一终端控制该 FPS 中的被控虚拟对象进行移动，也即是第一终端将对被控虚拟对象的控制信息发送给第一服务器，第一服务器基于该控制信息对该 FPS 的虚拟场景进行渲染，得到第一视频帧。第一服务器将该第一视频帧发送给第二服务器，若用户在游戏开始之前对该 FPS 游戏中的某个虚拟对象进行了个性化设置，比如将该 FPS 游戏中的车辆设置为红色，那么第二服务器能够对第一视频帧进行图像识别，确定该第一视频帧中是否包括车辆。若第二服务器确定该第一视频帧显示有车辆，那么从客户端服务器获取对该车辆的第一参数，该第一参数是客户端服务器基于用户对该车辆的设置参数确定的。第二服务器基于该第一参数对该第一视频帧中的该车辆进行渲染，得到第二视频帧，将第二视频帧发送给第一终端，用户通过第一终端就能够查看该第二视频帧。若在该 FPS 游戏中，技术人员将该车辆的颜色配置为蓝色，那么通过上述步骤，能够将该车辆的颜色调整为用户设置的红色，实现对该车辆的个性化配置。

需要说明的是，上述是以对第一视频帧进行处理为例进行说明的，在运行该 FPS 游戏的过程中，存在连续不断的多个视频帧，对于该多个视频帧，均采用上述方式进行处理，以供第一终端进行显示。

另外，对于 MOBA 游戏、TPS 游戏、战棋游戏以及自走棋游戏来说，均能够采用上述步骤来进行处理，比如在 MOBA 游戏中，用户能够通过第一终端将 MOBA 游戏中的树木设置为桃花，那么第二服务器能够从第一视频帧中识别出树木，将树木二次渲染为桃花，得到第二视频帧，将第二视频帧发送给第一终端进行显示。

还有，本申请实施例提供的视频帧的渲染方法除了能够应用在上述 FPS 游戏、MOBA 游戏、战棋游戏或者自走棋游戏之外，也能够应用在其他类型的云游戏中，本申请实施例对此

不做限定。

在介绍完本申请实施例提供的视频帧的渲染方法的实施环境和应用场景之后，下面对本申请实施例提供的视频帧的渲染方法进行说明。

图 2 是本申请实施例提供的一种视频帧的渲染方法的流程图，参见图 2，以方法由第二服务器执行为例，该方法包括：

201、第二服务器获取第一终端对应的第一视频帧，该第一视频帧是以目标虚拟场景中被控虚拟对象的视角，对该目标虚拟场景进行渲染得到的视频帧，该被控虚拟对象为该第一终端控制的虚拟对象。

其中，以目标虚拟场景中被控虚拟对象的视角，对该目标虚拟场景进行渲染是指：对目标虚拟场景中，被控虚拟对象观察到的画面进行渲染。由于被控虚拟对象是第一终端控制的虚拟对象，那么被控虚拟对象在目标虚拟场景中观察到的画面，能够模拟第一终端的用户在目标虚拟场景中看到的画面。

针对不同的第一终端来说，不同的第一终端具有不同的被控虚拟对象，在同一个目标虚拟场景中，不同被控虚拟对象观察到的画面应当是不同的，因此不同的第一终端显示出的画面也应当是不同的。为此，需要以第一终端的被控虚拟对象的视角，对该目标虚拟场景进行渲染，所得到的视频帧才是与第一终端对应的。

可选地，该第一视频帧由第一服务器渲染得到后，发送给第二服务器。

202、在该第一视频帧显示有该第一终端的目标虚拟对象的情况下，第二服务器基于该第一终端的第一参数，对该第一视频帧中的该目标虚拟对象进行渲染，得到第二视频帧。

其中，目标虚拟对象为第一终端设置的虚拟对象，是第一终端需要重新进行渲染的虚拟对象。例如目标虚拟对象为用户通过第一终端选择的虚拟对象，或者为第一终端根据被控虚拟对象的位置选择的虚拟对象，或者为采用其他方式设置的虚拟对象。第一参数为第一终端的第一参数，例如，第一参数是用户通过第一终端确定的第一参数，或者为第一终端默认设置的第一参数。该第一参数用于在第一视频帧中对目标虚拟对象进行重新渲染，得到第二视频帧。由于目标虚拟对象和第一参数均是通过第一终端配置的，不受其他的终端的影响，也就实现了对虚拟场景的个性化配置，当想要改变虚拟场景中某个虚拟对象的显示效果时，直接通过第一终端进行配置即可，无需联系技术人员改变底层代码和文件，效率较高。

203、第二服务器将该第二视频帧发送给该第一终端，该第一终端用于显示该第二视频帧。

可选地，本申请实施例提供的视频帧渲染方法应用在云游戏场景下，那么对虚拟场景的渲染以及对视频帧的渲染过程均在云端实现，这里的云端为第二服务器（如场景管理服务器）、第一服务器（如云服务器）以及其他相关服务器的统称。第二服务器对第一视频帧进行二次渲染之后，将得到的第二视频帧发送给第一终端，第一终端无需进行后台处理，直接对第二视频帧进行显示即可，效率较高。

需要说明的是，本申请实施例仅是以第二服务器执行该方法为例，在其他的实施例中，该方法还可以由其他的服务器执行。

通过本申请实施例提供的技术方案，如果想要改变某个虚拟对象的显示效果，可以通过第一终端为该虚拟对象确定第一参数，基于该第一参数对视频帧进行二次渲染，得到第二视频帧，相较于第一视频帧，第二视频帧中显示的虚拟对象的显示效果也即是第一终端为该虚拟对象配置的显示效果。通过这样的技术方案，在云应用中提供了一种二次渲染虚拟对象的功能，用户能够快捷、高效地调整虚拟场景中某个虚拟对象的显示效果，从而扩展了云应用的功能范围，提高了云应用的个性化，使得云应用传播更加广。

上述步骤 201-203 是对本申请实施例的简单介绍，下面将结合一些例子，对本申请实施例提供的技术方案进行更加清楚地描述，以方法由第二服务器执行为例，参见图 3，方法包括：

301、第二服务器获取第一终端对应的第一视频帧，该第一视频帧是以目标虚拟场景中被控虚拟对象的视角，对该目标虚拟场景进行渲染得到的视频帧，该被控虚拟对象为该第一终端控制的虚拟对象。

可选地，在游戏场景下，目标虚拟场景也即是云游戏的游戏场景，被控虚拟对象的视角也即是被控虚拟对象的虚拟摄像机的视角，在FPS游戏中，被控虚拟对象的虚拟摄像机位于被控虚拟对象的头部，用户通过第一终端控制被控虚拟对象在目标虚拟场景中进行移动时，该虚拟摄像机也会随着被控虚拟对象的移动而移动，该虚拟摄像机拍摄到的画面也即是被控虚拟对象在目标虚拟场景中观察到的画面。在TPS游戏中，被控虚拟对象的虚拟摄像机位于被控虚拟对象的上方，用户通过第一终端控制被控虚拟对象在目标虚拟场景中进行移动时，该虚拟摄像机也会随着被控虚拟对象的移动而移动，该虚拟摄像机拍摄到的画面也即是在被控虚拟对象上方观察到的画面。在云游戏场景下，该虚拟摄像机拍摄到的画面是由云游戏服务器渲染的，由于游戏过程中存在时序上连续的多个画面，该多个画面也被称为视频帧。同理地，在其他场景下，第一服务器也可以通过虚拟摄像机拍摄目标虚拟场景得到画面，所拍摄到的画面也被称为视频帧。可选地，第一服务器为云服务器。

在一种可能的实施方式中，第一终端向第一服务器发送对被控虚拟对象的控制信息，第一服务器接收到该控制信息之后，基于该控制信息确定目标虚拟场景中被控虚拟对象的视角。第一服务器基于该被控虚拟对象的视角，对该目标虚拟场景进行渲染，得到第一视频帧，该第一视频帧也即是第一终端对应的视频帧。第一服务器将该第一视频帧发送给第二服务器，第二服务器获取该第一视频帧。其中，对被控虚拟对象的控制信息用于改变被控虚拟对象在目标场景中的位置、朝向以及动作，比如，控制信息能够控制被控虚拟对象在目标虚拟场景中向前后左右进行移动，或者能够控制被控虚拟对象在目标虚拟场景中向左或者向右旋转，或者能够控制被控虚拟对象在目标虚拟场景中执行下蹲、匍匐以及使用虚拟道具等动作。当然，在第一服务器基于控制信息，控制被控虚拟对象在目标虚拟场景中运动或者执行动作时，与被控虚拟对象绑定的虚拟摄像机也会随着被控虚拟对象的运动而运动，被控虚拟对象进行运动或者执行动作会导致被控虚拟对象观察目标虚拟场景的视角发生变化，与被控虚拟对象绑定的虚拟摄像机能够记录这一变化。可选地，第一服务器为场景管理服务器。

在这种实施方式下，能够通过第一终端来向第一服务器发送对被控虚拟对象的控制信息，第一服务器能够基于该控制信息来确定被控虚拟对象在目标虚拟场景中的视角，基于被控虚拟对象在目标虚拟场景中的视角，对目标虚拟场景进行渲染，得到第一视频帧，无需第一终端进行渲染，提高了渲染效率，第二服务器直接对第一视频帧进行后续处理即可。

举例来说，第一终端显示目标虚拟场景，在目标虚拟场景中显示被控虚拟对象。第一终端与第一服务器之间建立有长连接，响应于对被控虚拟对象的操作，第一终端向第一服务器发送与该操作对应的控制信息，其中，长连接是指在一个连接上可以连续发送多个数据包，在连接保持期间，如果没有数据包发送，需要双方发链路检测包。在一些实施例中，对该被控虚拟对象的操作包括点击操作和拖动操作，其中，点击操作为对操作控件的点击操作，比如在FPS游戏中，对开火控件的点击操作。拖动操作为对操作控件或者对被控虚拟对象的拖动操作等，比如在MOBA游戏中，对技能控件的拖动操作等。第一服务器接收到该控制信息之后，基于该控制信息确定被控虚拟对象在目标虚拟场景中的位置、姿态以及动作。第一服务器基于被控虚拟对象在目标虚拟场景中的位置、姿态以及动作，确定被控虚拟对象的视角。第一服务器基于被控虚拟对象的视角，对该目标虚拟场景进行渲染，得到与第一终端对应的第一视频帧，该第一视频帧也即是基于第一终端发送的控制信息进行渲染得到的视频帧。比如，用户对该被控虚拟对象的操作为点击下蹲控件，那么第一终端向第一服务器发送的控制信息也就用于控制被控虚拟对象下蹲。第一服务器在目标虚拟场景中控制被控虚拟对象下蹲，确定下蹲后被控虚拟对象的视角。基于下蹲后被控虚拟对象的视角对目标虚拟场景进行渲染，得到第一视频帧。第一服务器将该第一视频帧发送给第二服务器，第二服务器获取该第一视频帧。

302、第二服务器对该第一视频帧进行图像识别，确定该第一视频帧的类型。

在一种可能的实施方式中，第二服务器采用目标虚拟对象的模板图像在该第一视频帧中进行检测。响应于检测到该第一视频帧中存在与该目标虚拟对象的模板图像匹配的区域，第二服务器将该第一视频帧确定为第一类型，该第一类型表示该第一视频帧显示有该目标虚拟对象。响应于检测到该第一视频帧中不存在与该目标虚拟对象的模板图像匹配的区域，第二服务器将该第一视频帧确定为第二类型，该第二类型表示该第一视频帧未显示有该目标虚拟对象。其中，目标虚拟对象的标识是客户端服务器在开始时发送给第二服务器的，第二服务器能够基于目标虚拟对象的标识确定需要进行识别的目标虚拟对象，例如第二服务器基于目标虚拟对象的标识查询与该标识对应存储的模板图像，即为该目标虚拟对象的模板图像，之后每次获取到第一终端对应的视频帧时，采用该模板图像进行检测。

其中，目标虚拟对象为第一终端设置的虚拟对象，目标虚拟对象为目标虚拟场景中的任意一虚拟对象。比如，目标虚拟对象为目标虚拟场景中的虚拟汽车、或者目标虚拟对象为目标虚拟场景中的虚拟树木，或者目标虚拟对象为虚拟场景中的虚拟房屋，或者目标虚拟对象为虚拟枪械等，本申请实施例对此不做限定。

在这种实施方式下，第二服务器能够通过模板匹配的方式来确定第一视频帧的类型，也即是确定第一视频帧中是否显示有目标虚拟对象，由于模板匹配的速度较快，那么采用模板匹配的方式来确定第一视频帧的类型时，效率也就较高。

下面通过两个例子对上述实施方式进行说明。

例 1、第二服务器采用该目标虚拟对象的模板图像，获取该目标虚拟对象的模板图像与第一视频帧中多个区域之间的相似度。响应于多个区域中存在与该目标虚拟对象的模板图像之间的相似度符合目标相似度条件的区域，第二服务器确定该区域与该目标虚拟对象的模板图像匹配，将该第一视频帧确定为第一类型。响应于多个区域与该目标虚拟对象的模板图像之间的相似度均不符合目标相似度条件，第二服务器确定该第一视频帧中不存在与该目标虚拟对象的模板图像匹配的区域，将该第一视频帧确定为第二类型。

比如，第二服务器采用该目标虚拟对象的模板图像在该第一视频帧上进行滑动，获取该目标虚拟对象的模板图像与第一视频帧上多个区域之间的相似度，该多个区域也即是目标虚拟对象的模板图像在第一视频帧上进行滑动时覆盖的区域。在一些实施例中，第二服务器确定该目标虚拟对象的模板图像与多个区域之间的相似度时，能够采用颜色值相似度或者灰度值相似度来实现，本申请实施例对此不做限定。响应于多个区域中存在与该目标虚拟对象的模板图像之间的相似度大于或等于相似度阈值的区域，第二服务器确定该区域与该目标虚拟对象的模板图像匹配，将该第一视频帧确定为第一类型，也即是该第一视频帧显示有该目标虚拟对象。响应于多个区域与该目标虚拟对象的模板图像之间的相似度小于相似度阈值，第二服务器确定该第一视频帧中不存在与该目标虚拟对象的模板图像匹配的区域，将该第一视频帧确定为第二类型，也即是该第一视频帧未显示有该目标虚拟对象。

例 2、第二服务器对目标虚拟对象的模板图像进行缩放处理，得到目标虚拟对象的多个尺寸的模板图像。第二服务器基于多个尺寸的模板图像，获取该目标虚拟对象的模板图像与第一视频帧上多个区域之间的相似度，多个区域包括尺寸不同的区域。响应于多个区域中存在与该目标虚拟对象的模板图像之间的相似度符合目标相似度条件的区域，第二服务器确定该区域与该目标虚拟对象的模板图像匹配，将该第一视频帧确定为第一类型。响应于多个区域与该目标虚拟对象的模板图像之间的相似度均不符合目标相似度条件，第二服务器确定该第一视频帧中不存在与该目标虚拟对象的模板图像匹配的区域，将该第一视频帧确定为第二类型。

由于被控虚拟对象在目标虚拟场景中会进行运动，被控虚拟对象与目标虚拟对象之间的距离在不同视频帧中可能是不同的，那么目标虚拟对象的尺寸在不同视频帧中也可能是不同的，采用将目标虚拟对象的模板图像缩放成多个尺寸的模板图像来进行模板匹配，能够提高模板匹配的准确性。

比如,第二服务器对该目标虚拟对象的模板图像进行缩放处理,得到目标虚拟对象的多尺寸尺寸的模板图像。第二服务器控制多个尺寸的模板图像在该第一视频帧上进行滑动,获取该多个尺寸的模板图像与第一视频帧上多个区域之间的相似度,该多个区域也即是目标虚拟对象的模板图像在第一视频帧上进行滑动时覆盖的区域。在一些实施例中,第二服务器确定多个尺寸的模板图像与多个区域之间的相似度时,能够采用颜色值相似度或者灰度值相似度来实现,本申请实施例对此不做限定。响应于多个区域中存在与该多个尺寸的模板图像中任一模板图像之间的相似度大于或等于相似度阈值的区域,第二服务器确定该区域与该目标虚拟对象的模板图像匹配,将该第一视频帧确定为第一类型,也即是该第一视频帧显示有该目标虚拟对象。响应于多个区域与该目标虚拟对象的模板图像之间的相似度小于相似度阈值,第二服务器确定该第一视频帧中不存在与该目标虚拟对象的模板图像匹配的区域,将该第一视频帧确定为第二类型,也即是该第一视频帧未显示有该目标虚拟对象。

在一种可能的实施方式中,第二服务器将该第一视频帧输入图像识别模型。第二服务器通过该图像识别模型,对该第一视频帧进行特征提取和分类,输出该第一视频帧的类型。

其中,该图像识别模型是基于样本视频帧集训练得到的,样本视频帧集包括正样本视频帧和负样本视频帧,正样本视频帧为显示有目标虚拟对象的视频帧,负样本视频帧为未显示有目标虚拟对象的视频帧。采用样本视频帧集训练得到的图像识别模型,具有判断视频帧中是否显示有目标虚拟对象的能力。在一些实施例中,图像识别模型是第二服务器提前训练好的,在游戏过程中能够直接使用该图像识别模型来确定视频帧的类型。

下面通过两个例子对上述实施方式进行说明。

例 1、第二服务器将该第一视频帧输入该图像识别模型。第二服务器通过该图像识别模型,对该第一视频帧进行特征提取和全连接处理,得到该第一视频帧对应的概率,该概率为该第一视频帧显示有该目标虚拟对象的概率。响应于该概率大于或等于概率阈值,第二服务器将该第一视频帧确定为第一类型,该第一类型表示该第一视频帧显示有该目标虚拟对象。响应于该概率小于该概率阈值,第二服务器将该第一视频帧确定为第二类型,该第二类型表示该第一视频帧未显示有该目标虚拟对象。

比如,第二服务器将该第一视频帧输入该图像识别模型,通过该图像识别模型,对该第一视频帧进行卷积处理,得到第一视频帧的特征图。第二服务器通过该图像识别模型,对第一视频帧的特征图进行全连接处理和归一化处理,得到该第一视频帧对应的概率分布列,其中,归一化处理可以采用 S 型生长曲线(Sigmoid)或者软最大化(Softmax)函数来进行,本申请实施例对此不做限定。响应于概率分布列中对应于第一类型的概率大于或等于概率阈值,第二服务器将该第一视频帧确定为第一类型,也即是第一视频帧显示有目标虚拟对象。响应于概率分布列中对应于第二类型的概率大于或等于概率阈值,第二服务器将该第一视频帧确定为第二类型,也即是第一视频帧未显示有目标虚拟对象。

例 2、第二服务器将该第一视频帧划分为多个图像块,将该多个图像块输入该图像识别模型。第二服务器通过该图像识别模型,对该多个图像块进行特征提取和全连接处理,得到该多个图像块分别对应的多个概率,该概率为对应的图像块包括该目标虚拟对象的概率。响应于该多个概率中的任一概率大于或等于概率阈值,第二服务器将该第一视频帧确定为第一类型,该第一类型表示该第一视频帧显示有该目标虚拟对象。响应于该多个概率均小于该概率阈值,第二服务器将该第一视频帧确定为第二类型,该第二类型表示该第一视频帧未显示有该目标虚拟对象。

比如,第二服务器将该第一视频帧划分为尺寸相同的多个图像块,将该多个图像块输入该图像识别模型,通过该图像识别模型,对该多个图像块进行卷积处理,得到多个图像块的特征图。第二服务器通过该图像识别模型,对多个图像块的特征图进行全连接处理和归一化处理,得到该多个图像块对应的多个概率,该概率为对应的图像块显示有目标虚拟对象的概率,其中,归一化处理可以采用 S 型生长曲线(Sigmoid)或者软最大化(Softmax)函数来进行,本申请实施例对此不做限定。响应于任一概率大于或等于概率阈值,第二服务器将第

一视频帧确定为第一类型，也即是第一视频帧显示有目标虚拟对象。响应于多个概率均小于概率阈值，第二服务器将该第一视频帧确定为第二类型，也即是第一视频帧未显示有目标虚拟对象。

需要说明的是，在步骤 302 之后，在该类型表示第一视频帧显示有目标虚拟对象的情况下，那么第二服务器执行下述步骤 303；在该类型表示第一视频帧未显示有目标虚拟对象的情况下，那么第二服务器执行下述步骤 304。

303、在该类型表示第一视频帧显示有该第一终端的目标虚拟对象的情况下，第二服务器基于该第一终端的第一参数，对该第一视频帧中的该目标虚拟对象进行渲染，得到第二视频帧。

在一种可能的实施方式中，在该第一视频帧显示有该第一终端的目标虚拟对象的情况下，第二服务器获取第一渲染参数，该第一渲染参数为该第一参数中与第一角度和第一距离对应的渲染参数，该第一角度为该被控虚拟对象与该目标虚拟对象之间的角度，该第一距离为该被控虚拟对象与该目标虚拟对象之间的距离。第二服务器基于该第一渲染参数对该第一视频帧中的该目标虚拟对象进行渲染，得到该第二视频帧。

在这种实施方式下，在第一视频帧显示有目标虚拟对象的情况下，第二服务器能够基于第一渲染参数对目标虚拟对象进行二次渲染，以改变目标虚拟对象的显示效果，在游戏过程中，被控虚拟对象与目标虚拟对象之间的距离和角度可能随时发生变化，不同的距离和角度对应于不同的渲染参数，第二服务器能够基于第一角度和第一距离直接从第一参数中获取第一渲染参数，无需基于第一角度和第一距离进行二次运算来获取第一渲染参数，效率较高。

举例来说，在该第一视频帧显示有该第一终端的目标虚拟对象的情况下，第二服务器从第一服务器获取被控虚拟对象与目标虚拟对象之间的第一角度和第一距离。第二服务器将第一角度、第一距离以及目标虚拟对象的标识发送给客户端服务器。客户端服务器接收到第一角度、第一距离以及目标虚拟对象的标识之后，基于目标虚拟对象的标识确定第一参数。客户端服务器基于第一角度以及第一距离在第一参数中进行查询，得到第一渲染参数，将第一渲染参数发送给第二服务器。第二服务器基于该第一渲染参数，确定该第一视频帧中该目标虚拟对象的多个目标像素点的目标像素值。第二服务器采用该目标像素值，更新该第一视频帧中该多个目标像素点的像素值，得到该第二视频帧。

其中，该目标像素点是指该第一视频帧中该目标虚拟对象的像素点，在该第一视频帧中该目标像素点具有像素值，而该目标像素值是指该目标像素点需要更新得到的像素值，因此会采用该目标像素值更新该目标像素值的原像素值。

其中，被控虚拟对象和目标虚拟对象之间的角度，可以为被控虚拟对象的朝向与目标虚拟对象的朝向之间的角度。比如，被控虚拟对象为游戏场景中持有虚拟枪械的游戏人物，那么被控虚拟对象的朝向也即是虚拟枪械的瞄准方向。目标虚拟对象为虚拟车辆，那么目标虚拟对象的朝向也即是虚拟车辆的正前方。被控虚拟对象与目标虚拟对象之间的距离，为被控虚拟对象的中心点与目标虚拟对象的中心点之间的距离。

在一种可能的实施方式中，在该第一视频帧显示有该第一终端的目标虚拟对象的情况下，第二服务器从该第一参数中获取第一渲染参数和第二渲染参数。第二服务器在该第一视频帧中，采用该第一渲染参数和该第二渲染参数，分别对该目标虚拟对象和该被控虚拟对象进行渲染，得到该第二视频帧。

在这种实施方式下，由于第一参数是第一终端的参数，该第一参数中既携带了用于对目标虚拟对象进行渲染的第一渲染参数，也携带了对被控虚拟对象进行渲染的第二渲染参数，也就表示用户不仅能够对虚拟场景中的目标虚拟对象进行二次渲染，也能够对第一终端控制的被控虚拟对象进行二次渲染，提高了个性化和可玩性。

举例来说，在该第一视频帧显示有该第一终端的目标虚拟对象的情况下，第二服务器从第一服务器获取被控虚拟对象与目标虚拟对象之间的第一角度和第一距离。第二服务器将第一角度、第一距离以及目标虚拟对象的标识发送给客户端服务器。客户端服务器接收到第一

角度、第一距离以及目标虚拟对象的标识之后，基于目标虚拟对象的标识确定第一参数。客户端服务器基于第一角度以及第一距离在第一参数中进行查询，得到第一渲染参数和第二渲染参数，将第一渲染参数和第二渲染参数发送给第二服务器。第二服务器接收到第一渲染参数和第二渲染参数之后，在第一视频帧中，采用第一渲染参数对目标虚拟对象进行渲染，采用第二渲染参数对被控虚拟对象进行渲染，得到第二视频帧。

比如，用户既想要对目标虚拟对象的显示效果进行个性化设置，也想要在显示目标虚拟对象时对自己控制的被控虚拟对象的显示效果也进行个性化设置，那么能够通过上述实施方式来实现。以目标虚拟对象为虚拟车辆为例，用户想要将虚拟车辆的颜色设置为红色，同时还想要在目标虚拟场景中出现该虚拟车辆时，能够将第一终端控制的游戏人物（虚拟对象）的衣服显示为蓝色，那么也就能够通过上述实施方式，在第一参数中配置第一渲染参数和第二渲染参数。当第一视频帧显示有目标虚拟对象时，第二服务器也就能够在采用第一渲染参数对目标虚拟对象进行二次渲染的同时，采用第一渲染参数对被控虚拟对象进行二次渲染，提高了个性化。

在一种可能的实施方式中，在该第一视频帧显示有该第一终端的目标虚拟对象的情况下，第二服务器确定该被控虚拟对象在该目标虚拟场景中的位置。在该被控虚拟对象处于该目标虚拟场景的目标子场景的情况下，第二服务器基于该第一参数对该第一视频帧中的该目标虚拟对象进行渲染，得到该第二视频帧。在该被控虚拟对象不处于该目标虚拟场景的目标子场景的情况下，第二服务器将该第一视频帧确定为该第二视频帧。

其中，目标虚拟场景包括多个子场景，多个子场景也就构成了目标虚拟场景，比如，多个子场景分为虚拟社交场景和虚拟战斗场景，其中，虚拟社交场景包括虚拟钓鱼场景、虚拟聊天室、虚拟舞蹈室或者虚拟棋牌室等，在虚拟社交场景中，用户通过控制虚拟对象与其他用户进行交流。虚拟战斗场景也即是用户控制虚拟对象进行战斗的场景。目标子场景由第一终端确定，既可以是虚拟社交场景，也可以是虚拟战斗场景，本申请实施例对此不做限定。

在上述实施方式下，用户既能够通过第一终端确定第一参数，也能够确定对目标虚拟对象进行渲染的目标子场景，大大提升了个性化程度，提高了用户黏性。

举例来说，在该第一视频帧显示有该第一终端的目标虚拟对象的情况下，第二服务器从第一服务器获取被控虚拟对象在目标虚拟场景中的位置。也即是第二服务器向第一服务器发送位置获取请求，位置获取请求中携带有第一视频帧的时间戳和被控虚拟对象的标识。第一服务器获取位置获取请求之后，从位置获取请求中获取第一视频帧的时间戳和被控虚拟对象的标识，基于第一视频帧的时间戳和被控虚拟对象的标识进行查询，得到在第一视频帧的时间戳指示的时间点被控虚拟对象在目标虚拟场景中的位置。第一服务器向第二服务器发送被控虚拟对象在目标虚拟场景中的位置，该位置用于指示被控虚拟对象在该目标虚拟场景中所处的子场景。在第二服务器向第一服务器发送位置获取请求的同时，第二服务器向客户端服务器发送参数获取请求，参数获取请求中携带有目标虚拟对象的标识。客户端服务器接收到参数获取请求之后，从参数获取请求中获取目标虚拟对象的标识。客户端服务器基于目标虚拟对象的标识进行查询，得到与目标虚拟对象对应的第一参数和目标子场景的标识，将第一参数和目标子场景的标识发送给第二服务器。第二服务器获取第一服务器发送的位置、客户端服务器发送的第一参数以及目标子场景的标识。在该位置指示的子场景为目标子场景的情况下，第二服务器采用第一参数，对该第一视频帧中的该目标虚拟对象进行渲染，得到该第二视频帧。在该位置指示的子场景不为目标子场景的情况下，第二服务器不对第一视频帧进行渲染，直接将第一视频帧确定为该第二视频帧，也即是第二服务器能够直接将第一视频帧发送给第一终端，第一终端显示该第一视频帧。

比如，目标子场景为虚拟钓鱼场景，目标虚拟对象为虚拟车辆，当第二服务器确定第一视频帧显示有该虚拟车辆，同时被控虚拟对象位于该虚拟钓鱼场景时，能够采用第一参数对该虚拟车辆进行二次渲染，以改变该虚拟车辆的显示效果。当第二服务器确定第一视频帧显示有该虚拟车辆，但是被控虚拟对象不位于该虚拟钓鱼场景时，不采用第一渲染参数对第一

视频帧进行二次渲染, 直接将该第一视频帧确定为第二视频帧。

需要说明的是, 在上述步骤 302 中, 第二服务器除了确定出第一视频帧的类型之外, 在第一视频帧为第一类型的情况下, 还能够确定出第一视频帧中目标虚拟对象的位置。若第二服务器采用模板匹配的方法来确定第一视频帧的类型, 那么场景服务器能够将第一视频帧中, 与目标虚拟对象的模板图像匹配的区域确定为目标虚拟对象在第一视频帧中的位置。若第二服务器采用图像识别模型来确定第一视频帧的类型, 那么图像识别模型能够在第一视频帧中输出一个检测框, 该检测框也即是第一视频帧中目标虚拟对象所在的位置。在第二服务器确定目标虚拟对象在第一视频帧中的位置后, 就能够基于第一渲染参数, 对第一视频帧中目标虚拟对象所在的位置进行二次渲染, 得到第二视频帧。

比如, 在该第一视频帧显示有该第一终端的目标虚拟对象的情况下, 第二服务器向第一服务器发送相对位置获取请求, 该相对位置获取请求用于获取被控虚拟对象和目标虚拟对象之间的距离, 该相对位置获取请求中携带有被控虚拟对象的标识和目标虚拟对象的标识。第一服务器接收到相对位置获取请求之后, 从该相对位置获取请求中获取被控虚拟对象的标识和目标虚拟对象的标识, 基于被控虚拟对象的标识和目标虚拟对象的标识确定被控虚拟对象和目标虚拟对象之间的第一角度和第一距离。第一服务器向第二服务器发送被控虚拟对象和目标虚拟对象之间的第一角度和第一距离。第二服务器接收到被控虚拟对象和目标虚拟对象之间的第一角度和第一距离之后, 向客户端服务器发送渲染参数获取请求, 渲染参数获取请求中携带有该第一角度、第一距离以及目标虚拟对象的标识发送给客户端服务器。客户端服务器接收到渲染参数获取请求之后, 从渲染参数获取请求中获取第一角度、第一距离以及目标虚拟对象的标识, 基于目标虚拟对象的标识确定第一参数。客户端服务器基于第一角度以及第一距离在第一参数中进行查询, 得到第一渲染参数, 将第一渲染参数发送给第二服务器。第二服务器基于该第一渲染参数, 确定该第一视频帧中该目标虚拟对象的多个目标像素点的目标像素值。第二服务器采用该目标像素值, 更新该第一视频帧中该多个目标像素点的像素值, 得到该第二视频帧, 也即是采用该目标像素值, 替换多个目标像素点的像素值。在一些实施例中, 目标像素值也被称为纹理数据。

参见图 4, 第一服务器以目标虚拟场景中被控虚拟对象的视角, 对目标虚拟场景进行渲染得到第一视频帧 401, 第一视频帧 401 包括虚拟车辆 402, 该虚拟车辆 402 也即是目标虚拟对象。第二服务器采用第一渲染参数, 对虚拟车辆 402 进行二次渲染, 得到第二视频帧 403。可以看出, 第二视频帧 403 中的虚拟车辆 404 与第一视频帧 401 具有不同的显示效果。

在一些实施例中, 第一服务器内部有安卓容器, 安卓容器的渲染技术都采用安卓容器内渲染的技术, 其原理是因为在安卓容器内可以直接访问 GPU (Graphics Processing Unit, 图形处理器), GPU 具有高迸发的计算能力, 可以直接在安卓容器内完成渲染, 并且可以提升渲染性能, 大大降低了渲染多带来的延迟。一个安卓容器就是一个完整的 Android (安卓) 系统, 兼容性也完美, 可以实现游戏的渲染。游戏开始时, 第一服务器能够向客户端服务器发送渲染完毕的指令, 客户端服务器能够实时地记录当前用户账号的游戏时长信息, 如果时长不足, 客户端服务器可以通过信令告知客户端做处理 (比如, 时长不足时弹框提示时长不用。云游戏异常时, 让客户端弹框提示用户异常) 等等。

下面对客户端服务器生成目标虚拟对象的第一参数的方法进行说明。

在一种可能的实施方式中, 第一终端显示目标虚拟对象的配置界面, 该配置界面用于获取目标虚拟对象的配置信息。响应于在该配置界面上的操作, 将目标虚拟对象的配置信息发送给客户端服务器。客户端服务器接收目标虚拟对象的配置信息, 基于目标虚拟对象的配置信息, 创建目标虚拟对象的三维模型, 获取该三维模型对应的第一参数。在一些实施例中, 客户端服务器能将第一终端登录的用户账号 (User ID)、目标虚拟场景的标识 (Game ID)、目标虚拟对象的标识 (Object ID) 以及第一参数进行绑定存储, 这样在后续调用过程中, 就能够基于表示来确定第一参数, 效率较高。

举例来说, 第一终端显示目标云游戏的配置界面, 该配置界面中显示有该目标云游戏中

的多个虚拟对象，该目标云游戏是用户选中的云游戏，该目标云游戏的游戏场景也即是目标虚拟场景。响应于该多个虚拟对象中的目标虚拟对象被选中，第一终端显示目标虚拟对象的配置界面，该配置界面中显示有多个配置选项。第一终端基于多个配置选项中被选中的选项生成该目标虚拟对象的配置信息，将该目标虚拟对象的配置信息发送给客户端服务器。客户端服务器接收到该目标虚拟对象的配置信息之后，基于该目标虚拟对象的配置信息对目标虚拟对象的初始三维模型进行渲染，得到该目标虚拟对象的三维模型。客户端服务器获取虚拟摄像机与该目标虚拟对象的三维模型在不同角度和不同距离的渲染参数，该不同角度和不同距离的渲染参数构成该三维模型对应的第一参数。

比如，第一终端启动云游戏客户端，显示云游戏客户端的运行界面，运行界面上显示有待选的多个云游戏。响应于多个云游戏中的目标云游戏被选中，第一终端显示目标云游戏的配置界面，该配置界面中显示有该目标云游戏中可供进行个性化处理的多个虚拟对象，比如，该多个虚拟对象包括虚拟车辆、虚拟树木、虚拟房屋等。响应于该多个虚拟对象中的目标虚拟对象被选中，以目标虚拟对象为虚拟车辆为例，参见图 5，第一终端显示虚拟车辆的配置界面 501，配置界面 501 中显示有多个一级配置选项 502，一级配置选项用于选择虚拟车辆的类型，在一些实施例中，第一终端采用车辆的品牌来表示虚拟车辆的类型。响应于多个一级配置选项中的任一一级配置选项配选中，第一终端显示该一级配置选项对应的多个二级配置选项 503，二级配置选项用于选择虚拟车辆的款式，在一些实施例中，第一终端采用车辆的型号来表示虚拟车辆的款式。响应于多个二级配置选项中的任一二级配置选项配选中，第一终端显示该二级配置选项对应的多个三级配置选项 504，三级配置选项用于选择虚拟车辆的颜色，比如为红色、黑色、蓝色等。在一些实施例中，响应于多个三级配置选项中的任一三级配置选项配选中，第一终端显示多个四级配置选项 505，四级配置选项用于选择对目标虚拟对象的配置信息的应用范围，应用范围包括“仅自己可见”、“所有人可见”、“永久使用”以及“单次使用”等，其中，“仅自己可见”用于指示第二服务器在游戏过程中，仅对第一终端对应的视频帧进行二次渲染，不对其他终端对应的视频帧进行二次渲染。“所有人可见”用于指示第二服务器在游戏过程中，对所有终端对应的视频帧均进行二次渲染。比如，目标虚拟场景中的虚拟车辆默认显示为类型 A-款式 B-红色，第一终端的用户通过虚拟车辆的配置界面，将该虚拟车辆设置为类型 C-款式 D-白色，若第一终端的用户选择了“仅自己可见”，那么在游戏过程中，在第一终端显示的目标虚拟场景中，虚拟车辆显示为类型 C-款式 D-白色，而对于显示该目标虚拟场景的其他终端来说，该虚拟车辆还是显示为类型 A-款式 B-红色。当然，若第一终端的用户选择了“所有人可见”，那么在游戏过程中，显示目标虚拟场景的所有终端均会将该虚拟车辆显示为类型 C-款式 D-白色。对于选项“永久使用”来说，第二服务器每次均会按照该次确定的配置信息来确定渲染参数。对于选项“单次使用”来说，第二服务器只会在此次游戏过程中按照该次确定的配置信息来确定渲染参数。第一终端基于上述一级配置选项、二级配置选项、三级配置选项以及四级配置选项，确定该目标虚拟对象的设置参数，将该设置参数发送给客户端服务器，由客户端服务器基于该目标虚拟对象的设置参数，借助开放图形库（Open Graphics Library, OpenGL）来创建目标虚拟对象的三维模型。客户端服务器获取虚拟摄像机与该目标虚拟对象的三维模型在不同角度和不同距离的渲染参数，该不同角度和不同距离的渲染参数构成该三维模型对应的第一参数。

在一些实施例中，以目标虚拟场景包括虚拟车辆为例，技术人员能够通过客户端服务器，基于车辆基本信息（品牌 CarType、款式 CarStyle、颜色 CarColor 等）提前创建大量的虚拟车辆的三维模型，这样能够在游戏过程中直接调用对应的三维模型，效率较高。

可选地，第二服务器获取第二视频帧之后，还能够执行下述步骤：

在一种可能的实施方式中，第二服务器获取该目标虚拟对象对应的第二动画和第二音频。第二服务器在该第二视频帧中添加与该目标虚拟对象对应的第二动画，得到第七视频帧。第二服务器将该第七视频帧和该第二音频发送给该第一终端，该第一终端在显示该第七视频帧的同时播放该第二音频。其中，在第二视频帧中添加第二动画的位置由用户或者技术人员确

定，比如将第二动画添加在与目标虚拟对象相邻的位置上，比如为目标虚拟对象的上方，或者旁边等，本申请实施例对此不做限定。

在这种实施方式下，第二服务器基于第一参数对第一视频帧进行二次渲染，得到第二视频帧之后，还能够在第二视频帧的基础上进行进一步的渲染，也即是在第二视频帧上添加与目标虚拟对象对应的第二动画，得到第七视频帧。除此之外，还获取了目标虚拟对象对应的第二音频，将第七视频帧和第二音频同时发送给第一终端之后，第一终端就能够在显示第七视频帧的同时播放第二音频，由于第一参数、第二动画以及第二音频均是用户通过第一终端确定的，那么对于同一个第一视频帧来说，由于不同的设置，可能被第二服务器渲染为不同的第七视频帧，以及配有不同的第二音频，从而提高了个性化。

举例来说，第二服务器从客户端服务器获取目标虚拟对象对应的第二动画和第二音频，第二动画和第二音频均是由第一终端确定的。也即是，第二服务器向客户端服务器发送第一动画获取请求，第一动画获取请求中携带有目标虚拟对象的标识。客户端服务器接收到第一动画获取请求之后，从第一动画获取请求中获取目标虚拟对象的标识，基于目标虚拟对象的标识进行查询，获取目标虚拟对象对应的第二动画和第二音频，将第二动画和第二音频发送给第二服务器。第二服务器接收第二动画和第二音频之后，在第二视频帧中添加第二动画，比如在第二视频帧中添加第二动画的第一帧，得到第七视频帧。第二服务器将第七视频帧和第二音频发送给第一终端，第一终端接收第七视频帧和第二音频，在显示第七视频帧的同时，播放第二音频。

从用户的角度来说，若目标虚拟对象为虚拟车辆，第二动画为一个逐渐出现的大拇指，第一参数为将该虚拟车辆调整为红色。那么在第一终端显示的第七视频帧中，该虚拟车辆的颜色为红色，在该虚拟车辆的旁边，显示该大拇指，同时，播放第二音频。

304、在该类型表示第一视频帧未显示有该第一终端的目标虚拟对象的情况下，第二服务器将该第一视频帧发送给该第一终端，该第一终端用于显示该第一视频帧。

通过步骤 304，当第一视频帧未显示有目标虚拟对象时，第二服务器也就无需对第一视频帧进行二次渲染，直接将第一视频帧转发给第一终端，由第一终端显示第一视频帧即可。

305、第二服务器将该第二视频帧发送给该第一终端，该第一终端用于显示该第二视频帧。

通过步骤 305，在第一视频帧显示有目标虚拟对象的情况下，第二服务器基于第一渲染参数对第一视频帧进行二次渲染，得到第二视频帧，将第二视频帧发送给第一终端，第一终端上显示的第二视频帧中的目标虚拟对象以通过第一终端配置的显示效果进行显示，从而实现了目标虚拟对象显示效果的快捷调整。

可选地，第二服务器将第二视频帧发送给第一终端之后，还能够执行下述任一个步骤：

在一种可能的实施方式中，第二服务器将多个该第二视频帧聚合为第一视频帧集合，将该第一视频帧集合发送给该第一终端，该第一终端用于向其他终端分享该第一视频帧集合。

在这种实施方式下，由于第二视频帧是第二服务器基于用户设置进行二次渲染的视频帧，那么该二次视频帧的显示效果也就是个性化的，第二服务器能够将该第二视频帧聚合为一个第一视频帧集合，将该第一视频帧集合发送给第一终端，第一终端就能够向其他终端分享该第一视频帧集合，也即是用户与其他用户分享自己对云应用的个性化设置，激发其他用户玩该云应用，使得该云应用的传播更广。

在一些实施例，第一终端获取该第一视频帧集合之后，能够通过社交网络分享该第一视频帧集合，以使得该第一视频帧集合进行更大范围的传播。

在一种可能的实施方式中，第二服务器将该第二视频帧和该第一视频帧进行拼接，得到拼接视频帧。第二服务器将多个该拼接视频帧聚合为第二视频帧集合，将该第二视频帧集合发送给该第一终端，该第一终端用于向其他终端分享该第二视频帧集合。

在这种实施方式下，第二视频帧是第二服务器基于用户设置进行二次渲染的视频帧，第一视频帧是第一服务器默认渲染的视频帧，将第二视频帧和第一视频帧拼接为拼接视频帧之后，用户就能够通过查看拼接视频帧来更清楚的看到用户的个性化设置。第二服务器能够将

该拼接视频帧聚合为一个第二视频帧集合，将该第二视频帧集合发送给第一终端，第一终端就能够向其他终端分享该第二视频帧集合，也即是用户与其他用户分享自己对云应用的个性化设置，激发其他用户玩该云应用，使得该云应用的传播更广。

需要说明的是，在上述步骤 301-305 中，是以第二服务器对第一视频帧进行处理，得到第二视频帧为例进行说明的，在游戏过程中，第一服务器对目标场景进行实时渲染，得到一系列视频帧，这一系列的视频帧均是与第一终端对应的视频帧，第一视频帧为这一系列视频帧中的一个视频帧。对于这一系列视频帧来说，第二服务器均能够执行上述步骤 301-305。

可选地，若第一终端选择了全局可见，那么第二服务器还能够执行下述步骤 306-308，以使第二服务器对其他终端对应的视频帧进行二次渲染，以改变虚拟场景中目标虚拟对象的显示效果。

306、第二服务器获取该第二终端对应的第五视频帧，该第五视频帧是以该第二虚拟对象的视角，对该目标虚拟场景进行渲染得到的视频帧。

其中，第二服务器获取第二终端对应的第五视频帧的方法，与第二服务器获取第一终端对应的第一视频帧的方法属于同一发明构思，实现过程参见上述步骤 301 的相关描述，在此不再赘述。

307、在该第五视频帧显示有该目标虚拟对象的情况下，第二服务器该第五视频帧进行目标处理，得到第六视频帧。

其中，第二服务器确定第五视频帧是否显示有目标虚拟对象的方法与第二服务器确定第一视频帧是否显示有目标虚拟对象的方法属于同一发明构思，实现过程参见上述步骤 302 的相关描述，在此不再赘述。

在一种可能的实施方式中，在该第五视频帧显示有该目标虚拟对象的情况下，第二服务器基于该第二终端的第二参数对该第五视频帧中的该目标虚拟对象进行渲染，得到该第六视频帧。可选地，该第二参数由第二终端确定。

在这种情况下，若第二终端针对目标虚拟对象设置了第二参数，那么第二服务器能够采用第二参数对第五视频帧中的该目标虚拟对象进行渲染，得到第六视频帧，后续能够将该第六视频帧发送给第二终端，由第二终端进行显示。也即是，第二服务器能够采用第一参数对包括该目标虚拟对象的视频帧进行二次渲染，将二次渲染后的视频帧发送给第一终端。第二服务器采用第二参数对包括该目标虚拟对象的视频帧进行二次渲染，将二次渲染后的视频帧发送给第二终端。在第一终端和第二终端上显示的视频帧中，该目标虚拟对象可能具有不同的显示效果。比如，以该目标虚拟对象为虚拟车辆为例，第一参数指示将该虚拟车辆渲染为红色，第二参数指示将该虚拟车辆渲染为蓝色，那么在第一终端显示的视频帧中，该虚拟车辆为红色，在第二终端显示的视频帧中，该虚拟车辆为蓝色。

举例来说，若第二终端确定了对目标虚拟对象的第二参数，在该第五视频帧显示有该目标虚拟对象的情况下，第二服务器从第一服务器获取第二虚拟对象与目标虚拟对象之间的第二角度和第二距离。第二服务器将第二角度、第二距离以及目标虚拟对象的标识发送给客户端服务器。客户端服务器接收到第二角度、第二距离以及目标虚拟对象的标识之后，基于目标虚拟对象的标识确定第二参数。客户端服务器基于第二角度以及第二距离在第二参数中进行查询，得到第三渲染参数，将第三渲染参数发送给第二服务器。第二服务器接收到第三渲染参数之后，在第五视频帧中，采用第三渲染参数对目标虚拟对象进行渲染，得到第六视频帧。

在一种可能的实施方式中，在该第五视频帧显示有该目标虚拟对象的情况下，第二服务器基于该第一参数对该第五视频帧中的该目标虚拟对象进行渲染，得到该第六视频帧。

在这种情况下，若第二终端未针对目标虚拟对象设置第二参数，那么第二服务器能够采用第一参数对第五视频帧中的该目标虚拟对象进行渲染，得到第六视频帧，后续能够将该第六视频帧发送给第二终端，由第二终端进行显示。也即是，在第一终端和第二终端上显示的视频帧中，该目标虚拟对象具有相同的显示效果。

举例来说,若第二终端未确定对目标虚拟对象的第二参数,在该第五视频帧显示有该目标虚拟对象的情况下,服务器从第一服务器获取第二虚拟对象与目标虚拟对象之间的第二角度和第二距离。第二服务器将第二角度、第二距离以及目标虚拟对象的标识发送给客户端服务器。客户端服务器接收到第二角度、第二距离以及目标虚拟对象的标识之后,基于目标虚拟对象的标识确定第一参数。客户端服务器基于第二角度以及第二距离在第一参数中进行查询,得到第三渲染参数,将第三渲染参数发送给第二服务器。第二服务器接收到第三渲染参数之后,在第五视频帧中,采用第三渲染参数对目标虚拟对象进行渲染,得到第六视频帧。

在一种可能的实施方式中,在该第五视频帧显示有该目标虚拟对象的情况下,第二服务器在该第五视频帧中添加与该目标虚拟对象对应的第二动画,得到该第六视频帧。

在这种实施方式下,第二服务器能够在第五视频帧中添加目标虚拟对象对应的第二动画,从而提高了个性化。

在一种可能的实施方式中,该被控虚拟对象和该第二虚拟对象同时处于该目标虚拟场景的目标子场景,在该第五视频帧显示有该目标虚拟对象的情况下,第二服务器比较该被控虚拟对象和该第二虚拟对象的虚拟等级。在该被控虚拟对象的虚拟等级高于该第二虚拟对象的虚拟等级的情况下,第二服务器基于该第一参数对该第五视频帧中的该目标虚拟对象进行渲染,得到该第六视频帧。在该被控虚拟对象的虚拟等级低于该第二虚拟对象的虚拟等级的情况下,第二服务器基于该第二终端的第二参数对该第五视频帧中的该目标虚拟对象进行渲染,得到该第六视频帧。在该被控虚拟对象和该第二虚拟对象的虚拟等级相同的情况下,第二服务器基于第三参数对该第五视频帧中的该目标虚拟对象进行渲染,得到该第六视频帧。可选地,该第三参数由第一服务器设置。

其中,目标虚拟场景包括多个子场景,多个子场景也就构成了目标虚拟场景,比如,多个子场景分为虚拟社交场景和虚拟战斗场景,其中,虚拟社交场景包括虚拟钓鱼场景、虚拟聊天室、虚拟舞蹈室或者虚拟棋牌室等,在虚拟社交场景中,用户通过控制虚拟对象与其他用户进行交流。虚拟战斗场景也即是用户控制虚拟对象进行战斗的场景。目标子场景由第一终端确定,既可以是虚拟社交场景,也可以是虚拟战斗场景,本申请实施例对此不做限定。虚拟等级为虚拟对象在目标虚拟场景中的等级,等级越高,那么虚拟对象的战斗能力也就越强。或者虚拟等级为虚拟对象对应游戏账号的会员等级,会员等级越高,也就能够在云应用中享受更多的服务。

在这种实施方式下,当该被控虚拟对象和第二虚拟对象同时处于该目标子场景时,第二服务器能够基于被控虚拟对象和第二虚拟对象的虚拟等级来确定采用第一终端设置的第一参数还是第二终端的第二参数来对目标虚拟对象进行二次渲染,从而鼓励用户提升虚拟对象的虚拟等级,提高用户的游戏积极性。

举例来说,在该被控虚拟对象和该第二虚拟对象同时处于该目标虚拟场景的目标子场景的情况下,第二服务器向客户端服务器发送虚拟等级获取请求,该虚拟等级获取请求中携带有被控虚拟对象的标识以及第二虚拟对象的标识。客户端服务器获取该虚拟等级获取请求之后,从该虚拟等级获取请求中获取被控虚拟对象的标识以及第二虚拟对象的标识,基于被控虚拟对象的标识以及第二虚拟对象的标识进行查询,得到被控虚拟对象的第一虚拟等级以及第二虚拟对象的第二虚拟等级。客户端服务器将被控虚拟对象的第一虚拟等级以及第二虚拟对象的第二虚拟等级发送给第二服务器。第二服务器接收到被控虚拟对象的第一虚拟等级以及第二虚拟对象的第二虚拟等级之后,在第一虚拟等级大于第二虚拟等级的情况下,第二服务器从客户端服务器获取第一参数,采用第一参数对该第五视频帧中的该目标虚拟对象进行渲染,得到该第六视频帧。在第一虚拟等级小于第二虚拟等级的情况下,第二服务器从客户端服务器获取第二参数,采用第二参数对该第五视频帧中的该目标虚拟对象进行渲染,得到该第六视频帧。在第一虚拟等级等于第二虚拟等级的情况下,第二服务器从客户端服务器获取第三参数,采用第三参数对该第五视频帧中的该目标虚拟对象进行渲染,得到该第六视频帧,在一些实施例中,该第三参数基于第一虚拟等级确定。

308、第二服务器将该第六视频帧发送给该第二终端,该第二终端用于显示该第六视频帧。

上述所有可选技术方案,可以采用任意结合形成本申请的可选实施例,在此不再一一赘述。

需要说明的是,本申请实施例仅是以第二服务器执行该方法为例,在其他的实施例中,该方法还可以由其他的服务器执行。

通过本申请实施例提供的技术方案,如果想要改变某个虚拟对象的显示效果,可以通过第一终端为该虚拟对象确定第一参数,由服务器基于该第一参数对视频帧进行二次渲染,得到第二视频帧,相较于第一视频帧,第二视频帧中显示的虚拟对象的显示效果也即是第一终端为该虚拟对象配置的显示效果。通过这样的技术方案,在云应用中提供了一种二次渲染虚拟对象的功能,用户能够快捷、高效地调整虚拟场景中某个虚拟对象的显示效果,从而扩展了云应用的功能范围,提高了云应用的个性化,使得云应用传播更加广。

除了上述步骤 301-308 之外,本申请实施例还提供了另一种视频帧的渲染方法,这种视频帧的渲染方法适用于目标虚拟场景发生目标事件的情况,参见图 6,方法包括:

601、第二服务器获取该第一终端对应的第三视频帧,该第三视频帧是在该目标虚拟场景中发生目标事件后,以该被控虚拟对象的视角对该目标虚拟场景进行渲染得到的视频帧,该目标事件为与该被控虚拟对象相关联的事件。

在一种可能的实施方式中,响应于目标虚拟场景中发生目标事件,第一服务器向第二服务器发送第一终端对应的第三视频帧,第一服务器向第二服务器发送第三视频帧的同时,向第二服务器发送与第三视频帧绑定的提示信息,该提示信息中携带有目标事件的标识。第二服务器获取到第三视频帧的同时,能够基于该提示信息确定目标虚拟场景中发生目标事件,以触发后续的对第三视频帧的渲染。

举例来说,目标事件为被控虚拟对象在目标虚拟场景中击败了第一虚拟对象,这里的击败是指,被控虚拟对象在目标虚拟场景中的行为使得第一虚拟对象的生命值降低为 0,比如,被控虚拟对象使用虚拟枪械/虚拟匕首/虚拟手雷等攻击第一虚拟对象,使得第一虚拟对象的生命值降低为 0,第一虚拟对象为与被控虚拟对象处于不同队伍的对象,或者说第一虚拟对象是与被控虚拟对象敌对的虚拟对象。响应于被控虚拟对象在目标虚拟场景中击败了第一虚拟对象,第一服务器向第二服务器实时发送第三视频帧,该第三视频帧也即是第一虚拟对象被击败后的第一个视频帧。第一服务器向第二服务器发送第三视频帧的同时,发送与第三视频帧绑定的提示信息,该提示信息除了携带有目标事件的标识之外,还携带有第一虚拟对象被击败的位置,后续第二服务器除了能够基于该提示信息确定被控虚拟对象在目标虚拟场景中击败了第一虚拟对象,还能够在第三视频帧中确定第一虚拟对象被击败的区域。在一些实施例中,响应于目标虚拟场景中发生目标事件,第一服务器能够还用钩子(Hook)函数来 Hook 住目标事件,将目标事件告知第二服务器,也即是通过提示信息的方式来告知第二服务器目标虚拟场景中发生了目标事件。

需要说明的是,上述举例是以目标事件为被控虚拟对象在目标虚拟场景中击败了第一虚拟对象为例进行说明的,在其他可能的实施方式中,目标事件还可以为被控虚拟对象在目标虚拟场景中拾取了目标虚拟道具,或者为被控虚拟对象在目标虚拟场景中启动了目标虚拟载具、或者为被控虚拟对象在目标虚拟场景中连续击败了多个虚拟对象等,目标事件由技术人员根据实际情况进行设置,本申请实施例对此不做限定。

602、第二服务器在该第三视频帧中添加与该目标事件对应的第一动画,得到第四视频帧。

在一种可能的实施方式中,目标事件为被控虚拟对象在目标虚拟场景中击败了第一虚拟对象,第二服务器在该第三视频帧中确定该第一虚拟对象被击败的区域。第二服务器在该区域中添加与该目标事件对应的第一动画,得到该第四视频帧。其中,目标事件与动画之间的对应关系通过第一终端进行设置,例如,通过第一终端选择目标事件,从客户端服务器提供的多个动画中选择与目标事件绑定的动画,或者通过第一终端向客户端服务器上传与目标事

件对应的第一动画,客户端服务器接收到第一终端上传的动画后,将该动画与目标事件绑定。

举例来说,第一服务器向第二服务器发送第三视频帧的同时,还发送了与第三视频帧绑定的提示信息。第二服务器从该提示信息中获取目标事件的标识以及第一虚拟对象被击败的位置。第二服务器基于目标事件的标识,确定目标事件对应的第一动画,基于第一虚拟对象被击败的位置,确定第一虚拟对象被击败的区域。第二服务器在该区域中添加与该目标事件对应的第一动画,得到该第四视频帧。

比如,第一服务器向第二服务器发送第三视频帧的同时,还发送了与第三视频帧绑定的提示信息。第二服务器从该提示信息中获取目标事件的标识以及第一虚拟对象被击败的位置。第二服务器向客户端服务器发送第二动画获取请求,第二动画获取请求中携带有目标事件的标识。客户端服务器接收到第二动画获取请求之后,从第二动画获取请求中获取目标事件的标识,基于目标事件的标识进行查询,得到目标事件对应的第一动画,将目标事件对应的第一动画发送给第二服务器。第二服务器接收到目标事件对应的第一动画之后,将该动画添加到第一虚拟对象被击败的区域,得到第四视频帧。在一些实施例中,第二服务器能够将该动画的第一帧添加到第三视频帧中,得到第四视频帧。

603、第二服务器将该第四视频帧发送给该第一终端,该第一终端用于显示该第四视频帧。

可选地,在步骤 603 之前,第二服务器还能够获取与该目标事件对应的第一音频,该第一音频为该第一终端的音频。第二服务器将该第一音频发送给该第一终端,该第一终端用于在显示该第四视频帧的同时播放该第一音频。其中,目标事件与第一音频之间的对应关系通过第一终端进行设置,例如,通过第一终端选择目标事件,从客户端服务器提供的多个第一音频中进行选择与目标事件绑定的第一音频,或者通过第一终端向客户端服务器上传与目标事件对应的第一音频,客户端服务器接收到第一终端上传的第一音频后,将该第一音频与目标事件绑定。

举例来说,第一服务器向第二服务器发送第三视频帧的同时,还发送了与第三视频帧绑定的提示信息。第二服务器从该提示信息中获取目标事件的标识。第二服务器向客户端服务器发送动画和音频获取请求,动画和音频获取请求中携带有目标事件的标识。客户端服务器接收到第二动画获取请求之后,从第二动画获取请求中获取目标事件的标识,基于目标事件的标识进行查询,得到目标事件对应的第一动画和第一音频,将目标事件对应的第一动画和第一音频发送给第二服务器。第二服务器接收到目标事件对应的第一动画和第一音频之后,将该动画添加第三视频帧中,得到第四视频帧。第二服务器向第一终端发送第四视频帧的同时,向第一终端发送第一音频。第一终端接收到第四视频帧和第一音频之后,在显示第四视频帧的同时播放该第一音频。

还是以目标事件为被控虚拟对象在虚拟场景中击败了第一虚拟对象为例,响应于被控虚拟对象在虚拟场景中击败了第一虚拟对象,第一服务器向第二服务器发送第三视频帧以及与第三视频帧绑定的提示信息,该提示信息中携带有目标事件的标识以及第一虚拟对象被击败的位置。第二服务器从该提示信息中获取目标事件的标识以及第一虚拟对象被击败的位置。第二服务器向客户端服务器发送动画和音频获取请求,动画和音频获取请求中携带有目标事件的标识。客户端服务器接收到动画和音频获取请求之后,从动画和音频获取请求中获取目标事件的标识,基于目标事件的标识进行查询,得到目标事件对应的第一动画和第一音频,将目标事件对应的第一动画和第一音频发送给第二服务器。第二服务器接收到目标事件对应的第一动画和第一音频之后,将该动画添加第三视频帧中第一虚拟对象被击败的位置,得到第四视频帧。第二服务器向第一终端发送第四视频帧的同时,向第一终端发送第一音频。第一终端接收到第四视频帧和第一音频之后,在显示第四视频帧的同时播放该第一音频。若该动画为一个小人跳舞的动画,那么在第四视频帧中,第一虚拟对象被击败的位置上播放该小人跳舞的动画的同时播放第一音频,用户通过动画和第一音频来激励自己在游戏中取得更好的表现。参见图 7,第二服务器在第四视频帧 701 中第一虚拟对象被击败的位置 702 添加动画 703。

通过本申请实施例提供的技术方案,如果想要在目标虚拟场景中发送目标事件时播放指定动画,可以通过第一终端为目标事件选定一个目标动画。通过第二服务器对视频帧进行二次渲染,得到第四视频帧,相较于第三视频帧,第四视频帧中能够显示通过第一终端设置的动画。通过这样的技术方案,在云应用中提供了一种二次渲染动画的功能,用户能够快速、高效地调整虚拟场景的显示效果,从而扩展了云应用的功能范围,提高了云应用的个性化,使得云应用的传播更加广。

在上述说明过程中,第一终端、客户端服务器、第一服务器、第二服务器以及游戏服务器构成了一个视频帧的渲染系统,在图1中对视频帧的渲染方法的实施环境进行了介绍,里面简要介绍了各个组成部分的功能,下面将结合上述方法实施例,对本申请实施例提供的视频帧的渲染系统进行介绍,参见图1,系统包括:第一终端、第一服务器以及第二服务器,该第一终端、该第一服务器以及该第二服务器之间通讯相连。

可选地,第一服务器为云服务器。

可选地,第二服务器为场景管理服务器。

该第一服务器用于以目标虚拟场景中被控虚拟对象的视角对该目标虚拟场景进行渲染,得到该第一终端对应的第一视频帧,将该第一视频帧发送给该第二服务器,该被控虚拟对象为该第一终端控制的虚拟对象。

该第二服务器用于接收该第一视频帧。

该第二服务器还用于在该第一视频帧显示有该第一终端的目标虚拟对象的情况下,基于该第一终端的第一参数对该第一视频帧中的该目标虚拟对象进行渲染,得到第二视频帧。

该第二服务器还用于向该第一终端发送该第二视频帧。

该第一终端用于响应于接收到该第二视频帧,显示该第二视频帧。

在一种可能的实施方式中,该第一终端还用于向该第一服务器发送控制信息,该控制信息用于控制该被控虚拟对象在该目标虚拟场景中的动作。

该第一服务器还用于基于该控制信息,确定该目标虚拟场景中该被控虚拟对象的视角,基于该目标虚拟场景中该被控虚拟对象的视角,对该目标虚拟场景进行渲染,得到该第一视频帧。

在一种可能的实施方式中,该系统还包括客户端服务器,该客户端服务器与该第一终端、该第一服务器以及该第二服务器分别通讯相连。

该第一终端还用于显示该目标虚拟对象的配置界面,该配置界面用于获取该目标虚拟对象的配置信息。

该第一终端还用于响应于在该配置界面上的操作,将该目标虚拟对象的配置信息发送给该客户端服务器。

该客户端服务器用于接收该目标虚拟对象的配置信息,基于该目标虚拟对象的配置信息创建该目标虚拟对象的三维模型,获取该三维模型对应的该第一参数。

在一种可能的实施方式中,该第二服务器还用于在该第一视频帧显示有该第一终端的目标虚拟对象的情况下,获取第一渲染参数,该第一渲染参数为该第一参数中与第一角度和第一距离对应的渲染参数,该第一角度为该被控虚拟对象与该目标虚拟对象之间的角度,该第一距离为该被控虚拟对象与该目标虚拟对象之间的距离。基于该第一渲染参数对该第一视频帧中的该目标虚拟对象进行渲染,得到该第二视频帧。

在一种可能的实施方式中,该第二服务器还用于基于该第一渲染参数,确定该第一视频帧中该目标虚拟对象的多个目标像素点的目标像素值。采用该目标像素值,更新该第一视频帧中该多个目标像素点的像素值,得到该第二视频帧。

在一种可能的实施方式中,该第二服务器还用于在该第一视频帧显示有该第一终端的目标虚拟对象的情况下,从该第一参数中获取第一渲染参数和第二渲染参数。在该第一视频帧中,采用该第一渲染参数和该第二渲染参数,分别对该目标虚拟对象和该被控虚拟对象进行

渲染, 得到该第二视频帧。

在一种可能的实施方式中, 该第二服务器还用于在该第一视频帧显示有该第一终端的目标虚拟对象的情况下, 确定该被控虚拟对象在该目标虚拟场景中的位置。在该被控虚拟对象处于该目标虚拟场景的目标子场景的情况下, 基于该第一参数对该第一视频帧中的该目标虚拟对象进行渲染, 得到该第二视频帧。在该被控虚拟对象不处于该目标虚拟场景的目标子场景的情况下, 将该第一视频帧确定为该第二视频帧。

在一种可能的实施方式中, 该第二服务器还用于对该第一视频帧进行图像识别, 确定该第一视频帧的类型。在该类型表示该第一视频帧显示有该目标虚拟对象的情况下, 基于该第一参数对该目标虚拟对象进行渲染, 得到该第二视频帧。

在一种可能的实施方式中, 该第二服务器还用于采用该目标虚拟对象的模板图像在该第一视频帧中进行检测。响应于检测到该第一视频帧中存在与该目标虚拟对象的模板图像匹配的区域, 将该第一视频帧确定为第一类型, 该第一类型表示该第一视频帧显示有该目标虚拟对象。响应于检测到该第一视频帧中不存在与该目标虚拟对象的模板图像匹配的区域, 将该第一视频帧确定为第二类型, 该第二类型表示该第一视频帧未显示有该目标虚拟对象。

在一种可能的实施方式中, 该第二服务器还用于将该第一视频帧输入图像识别模型。通过该图像识别模型, 对该第一视频帧进行特征提取和分类, 输出该第一视频帧的类型。

在一种可能的实施方式中, 该第二服务器还用于将该第一视频帧划分为多个图像块, 将该多个图像块输入该图像识别模型。该通过该图像识别模型, 对该第一视频帧进行特征提取和分类, 输出该第一视频帧的类型包括: 通过该图像识别模型, 对该多个图像块进行特征提取和全连接处理, 得到该多个图像块分别对应的多个概率, 该概率为对应图像块包括该目标虚拟对象的概率。响应于该多个概率中的任一概率大于或等于概率阈值, 将该第一视频帧确定为第一类型, 该第一类型表示该第一视频帧显示有该目标虚拟对象。响应于该多个概率均小于该概率阈值, 将该第一视频帧确定为第二类型, 该第二类型表示该第一视频帧未显示有该目标虚拟对象。

在一种可能的实施方式中, 该第二服务器还用于从客户端服务器获取该第一参数, 该客户端服务器用于基于该第一终端上传的针对该目标虚拟对象的设置参数, 创建该目标虚拟对象的三维模型, 该第一参数包括该三维模型的多个渲染参数。

在一种可能的实施方式中, 该第一服务器还用于响应于该目标虚拟场景中发生目标事件, 以该被控虚拟对象的视角对该目标虚拟场景进行渲染, 得到该第一终端对应的第三视频帧, 将该第三视频帧发送给该第二服务器, 该目标事件为该被控虚拟对象在该目标虚拟场景中击败了第一虚拟对象。

该第二服务器还用于获取与该目标事件对应的第一动画以及与该目标事件对应的第一音频, 该第一音频为该第一终端的音频。

该第二服务器还用于在该第三视频帧中确定该第一虚拟对象被击败的区域。在该区域中添加与该目标事件对应的第一动画, 得到该第四视频帧。将该第四视频帧和该第一音频发送给该第一终端。

该第一终端还用于响应于接收到该第四视频帧和该第一音频, 在显示该第四视频帧的同时播放该第一音频。

在一种可能的实施方式中, 该系统还包括第二终端, 该第二终端、该第一服务器以及该第二服务器之间通讯相连, 该第二终端与该第一终端为不同终端。

该第一服务器还用于以该目标虚拟场景中第二虚拟对象的视角对该目标虚拟场景进行渲染, 得到该第二终端对应的第五视频帧, 将该第五视频帧发送给该第二服务器, 该第二虚拟对象为该第二终端控制的虚拟对象。

该第二服务器还用于在该第五视频帧显示有该目标虚拟对象的情况下, 对该第五视频帧进行目标处理, 得到第六视频帧。向该第二终端发送该第六视频帧。

该第二终端用于响应于接收到该第六视频帧, 显示该第六视频帧。

在一种可能的实施方式中，该第二服务器还用于执行下述任一项：

在该第五视频帧显示有该目标虚拟对象的情况下，基于该第二终端的第二参数对该第五视频帧中的该目标虚拟对象进行渲染，得到该第六视频帧。

在该第五视频帧显示有该目标虚拟对象的情况下，基于该第一参数对该第五视频帧中的该目标虚拟对象进行渲染，得到该第六视频帧。

在该第五视频帧显示有该目标虚拟对象的情况下，在该第五视频帧中添加与该目标虚拟对象对应的第二动画，得到该第六视频帧。

在一种可能的实施方式中，该第二服务器还用于在该第五视频帧显示有该目标虚拟对象的情况下，比较该被控虚拟对象和该第二虚拟对象的虚拟等级。在该被控虚拟对象的虚拟等级高于该第二虚拟对象的虚拟等级的情况下，基于该第一参数对该第五视频帧中的该目标虚拟对象进行渲染，得到该第六视频帧。在该被控虚拟对象的虚拟等级低于该第二虚拟对象的虚拟等级的情况下，基于该第二终端的第二参数对该第五视频帧中的该目标虚拟对象进行渲染，得到该第六视频帧。在该被控虚拟对象和该第二虚拟对象的虚拟等级相同的情况下，基于第三参数对该第五视频帧中的该目标虚拟对象进行渲染，得到该第六视频帧。

在一种可能的实施方式中，该第二服务器还用于获取该目标虚拟对象对应的第二动画和第二音频。在该第二视频帧中添加与该目标虚拟对象对应的第二动画，得到第七视频帧。将该第七视频帧和该第二音频发送给该第一终端，该第一终端用于在显示该第七视频帧的同时播放该第二音频。

在一种可能的实施方式中，该第二服务器还用于执行下述任一项：

将多个该第二视频帧聚合为第一视频帧集合，将该第一视频帧集合发送给该第一终端，该第一终端用于向其他终端分享该第一视频帧集合。

将该第二视频帧和该第一视频帧进行拼接，得到拼接视频帧。将多个该拼接视频帧聚合为第二视频帧集合，将该第二视频帧集合发送给该第一终端，该第一终端用于向其他终端分享该第二视频帧集合。

在一种可能的实施方式中，该第二服务器还用于从客户端服务器获取该第一参数，该客户端服务器用于基于该第一终端上传的针对该目标虚拟对象的设置参数，创建该目标虚拟对象的三维模型，该第一参数包括该三维模型的多个渲染参数。

下面将结合图1以及上述步骤301-308以及601-603，对本申请实施例提供的视频帧的渲染系统的进行说明。

第一终端显示目标虚拟对象的配置界面，该配置界面用于获取目标虚拟对象的配置信息。响应于在该配置界面上的操作，将目标虚拟对象的配置信息发送给客户端服务器。客户端服务器接收目标虚拟对象的配置信息，基于目标虚拟对象的配置信息，创建目标虚拟对象的三维模型，获取该三维模型对应的第一参数。以游戏场景为例，第一终端启动目标云游戏，保持与客户端服务器的长连接，向客户端服务器发送游戏启动指令，游戏启动指令中携带有用户账号、该目标云游戏的标识以及第一终端的硬件信息。客户端服务器接收到该游戏启动指令之后，将游戏启动指令发送给第一服务器，第一服务器接收到该游戏启动指令之后，从该游戏启动指令中获取用户账号、该目标云游戏的标识以及第一终端的硬件信息。第一服务器基于第一终端的硬件信息对该目标云游戏进行初始化，以实现渲染得到的游戏画面与第一终端之间的匹配，向与该目标云游戏对应的游戏服务器发送该用户账号。游戏服务器接收到该用户账号之后，将与该游戏账号对应的信息发送给第一服务器，第一服务器基于与该游戏账号对应的信息启动该目标云游戏。在运行该目标云游戏的过程中，显示目标云游戏的目标虚拟场景，目标虚拟场景中显示有被控虚拟对象。第一终端向第一服务器发送对被控虚拟对象的控制信息，第一服务器接收到该控制信息之后，基于该控制信息确定目标虚拟场景中被控虚拟对象的视角。第一服务器基于该被控虚拟对象的视角，对该目标虚拟场景进行渲染，得到第一视频帧，该第一视频帧也即是第一终端对应的视频帧。第一服务器将该第一视频帧发送给第二服务器，若用户在游戏开始之前对该目标云游戏中的某个虚拟对象进行了个性化设

置,比如将该目标云游戏中的车辆设置为红色,那么第二服务器能够对第一视频帧进行图像识别,确定该第一视频帧中是否包括车辆。若第二服务器确定该第一视频帧显示有车辆,那么从客户端服务器获取对该车辆的第一参数,该第一参数是客户端服务器基于用户对该车辆的设置参数确定的。第二服务器基于该第一参数对该第一视频帧中的该车辆进行渲染,得到第二视频帧,将第二视频帧发送给第一终端,用户通过第一终端就能够查看该第二视频帧。若在该目标云游戏中,技术人员将该车辆的颜色配置为蓝色,那么通过上述步骤,能够将该车辆的颜色调整为用户设置的红色,实现对该车辆的个性化配置。

通过本申请实施例提供的技术方案,如果想要改变某个虚拟对象的显示效果,可以通过第一终端为该虚拟对象确定第一参数,由服务器基于该第一参数对视频帧进行二次渲染,得到第二视频帧,相较于第一视频帧,第二视频帧中显示的虚拟对象的显示效果也即是第一终端为该虚拟对象配置的显示效果。通过这样的技术方案,在云应用中提供了一种二次渲染虚拟对象的功能,用户能够快捷、高效地调整虚拟场景中某个虚拟对象的显示效果,从而扩展了云应用的功能范围,提高了云应用的个性化,使得云应用传播更加广。

图8是本申请实施例提供的一种视频帧的渲染装置的结构示意图,参见图8,装置包括:第一视频帧获取模块801、渲染模块802以及发送模块803。

第一视频帧获取模块801,用于获取第一终端对应的第一视频帧,该第一视频帧是以目标虚拟场景中被控虚拟对象的视角,对该目标虚拟场景进行渲染得到的视频帧,该被控虚拟对象为该第一终端控制的虚拟对象。

渲染模块802,用于在该第一视频帧显示有该第一终端的目标虚拟对象的情况下,基于该第一终端的第一参数,对该第一视频帧中的该目标虚拟对象进行渲染,得到第二视频帧。

发送模块803,用于将该第二视频帧发送给该第一终端,该第一终端用于显示该第二视频帧。

在一种可能的实施方式中,该渲染模块802,用于在该第一视频帧显示有该第一终端的目标虚拟对象的情况下,获取第一渲染参数,该第一渲染参数为该第一参数中与第一角度和第一距离对应的渲染参数,该第一角度为该被控虚拟对象与该目标虚拟对象之间的角度,该第一距离为该被控虚拟对象与该目标虚拟对象之间的距离。基于该第一渲染参数对该第一视频帧中的该目标虚拟对象进行渲染,得到该第二视频帧。

在一种可能的实施方式中,该渲染模块802,用于基于该第一渲染参数,确定该第一视频帧中该目标虚拟对象的多个目标像素点的目标像素值。采用该目标像素值,更新该第一视频帧中该多个目标像素点的像素值,得到该第二视频帧。

在一种可能的实施方式中,该渲染模块802,用于在该第一视频帧显示有该第一终端的目标虚拟对象的情况下,从该第一参数中获取第一渲染参数和第二渲染参数。在该第一视频帧中,采用该第一渲染参数和该第二渲染参数,分别对该目标虚拟对象和该被控虚拟对象进行渲染,得到该第二视频帧。

在一种可能的实施方式中,该渲染模块802,用于在该第一视频帧显示有该第一终端的目标虚拟对象的情况下,确定该被控虚拟对象在该目标虚拟场景中的位置。在该被控虚拟对象处于该目标虚拟场景的目标子场景的情况下,基于该第一参数对该第一视频帧中的该目标虚拟对象进行渲染,得到该第二视频帧。

在一种可能的实施方式中,该渲染模块802,还用于在该被控虚拟对象不处于该目标虚拟场景的目标子场景的情况下,将该第一视频帧确定为该第二视频帧。

在一种可能的实施方式中,该装置还包括:

图像识别模块,用于对该第一视频帧进行图像识别,确定该第一视频帧的类型。

该渲染模块802用于在该类型表示该第一视频帧显示有该目标虚拟对象的情况下,基于该第一参数对该目标虚拟对象进行渲染,得到该第二视频帧。

在一种可能的实施方式中,该图像识别模块,用于采用该目标虚拟对象的模板图像在该

第一视频帧中进行检测。响应于检测到该第一视频帧中存在与该目标虚拟对象的模板图像匹配的区域,将该第一视频帧确定为第一类型,该第一类型表示该第一视频帧显示有该目标虚拟对象。响应于检测到该第一视频帧中不存在与该目标虚拟对象的模板图像匹配的区域,将该第一视频帧确定为第二类型,该第二类型表示该第一视频帧未显示有该目标虚拟对象。

在一种可能的实施方式中,该图像识别模块,用于将该第一视频帧输入图像识别模型,通过该图像识别模型,对该第一视频帧进行特征提取和分类,输出该第一视频帧的类型。

在一种可能的实施方式中,该图像识别模块,用于将该第一视频帧划分为多个图像块,将该多个图像块输入该图像识别模型。通过该图像识别模型,对该多个图像块进行特征提取和全连接处理,得到该多个图像块分别对应的多个概率,该概率为对应图像块包括该目标虚拟对象的概率。响应于该多个概率中的任一概率大于或等于概率阈值,将该第一视频帧确定为第一类型,该第一类型表示该第一视频帧显示有该目标虚拟对象。响应于该多个概率均小于该概率阈值,将该第一视频帧确定为第二类型,该第二类型表示该第一视频帧未显示有该目标虚拟对象。

在一种可能的实施方式中,该装置还包括:

第三视频帧获取模块,用于获取该第一终端对应的第三视频帧,该第三视频帧是在该目标虚拟场景中发生目标事件后,以该被控虚拟对象的视角对该目标虚拟场景进行渲染得到的视频帧,该目标事件为该被控虚拟对象在该目标虚拟场景中击败了第一虚拟对象。

该渲染模块 802,还用于获取与该目标事件对应的第一动画以及与该目标事件对应的第一音频,该第一音频为该第一终端的音频。在该第三视频帧中确定该第一虚拟对象被击败的区域。在该区域中添加与该目标事件对应的第一动画,得到该第四视频帧。

该发送模块 803,还用于将该第四视频帧和该第一音频发送给该第一终端,该第一终端用于在显示该第四视频帧的同时播放该第一音频。

在一种可能的实施方式中,该目标虚拟场景还包括第二虚拟对象,该第二虚拟对象为第二终端控制的虚拟对象,该第二终端与该第一终端为不同终端,该装置还包括:

第五视频帧获取模块,用于获取该第二终端对应的第五视频帧,该第五视频帧是以该第二虚拟对象的视角,对该目标虚拟场景进行渲染得到的视频帧。

该渲染模块 802,还用于在该第五视频帧显示有该目标虚拟对象的情况下,对该第五视频帧进行目标处理,得到第六视频帧。

该发送模块 803,还用于将该第六视频帧发送给该第二终端,该第二终端用于显示该第六视频帧。

在一种可能的实施方式中,该渲染模块 802,还用于执行下述任一项:

在该第五视频帧显示有该目标虚拟对象的情况下,基于该第二终端的第二参数对该第五视频帧中的该目标虚拟对象进行渲染,得到该第六视频帧。

在该第五视频帧显示有该目标虚拟对象的情况下,基于该第一参数对该第五视频帧中的该目标虚拟对象进行渲染,得到该第六视频帧。

在该第五视频帧显示有该目标虚拟对象的情况下,在该第五视频帧中添加与该目标虚拟对象对应的第二动画,得到该第六视频帧。

在一种可能的实施方式中,该被控虚拟对象和该第二虚拟对象同时处于该目标虚拟场景的目标子场景,该渲染模块 802,还用于在该第五视频帧显示有该目标虚拟对象的情况下,比较该被控虚拟对象和该第二虚拟对象的虚拟等级。在该被控虚拟对象的虚拟等级高于该第二虚拟对象的虚拟等级的情况下,基于该第一参数对该第五视频帧中的该目标虚拟对象进行渲染,得到该第六视频帧。在该被控虚拟对象的虚拟等级低于该第二虚拟对象的虚拟等级的情况下,基于该第二终端的第二参数对该第五视频帧中的该目标虚拟对象进行渲染,得到该第六视频帧。在该被控虚拟对象和该第二虚拟对象的虚拟等级相同的情况下,基于第三参数对该第五视频帧中的该目标虚拟对象进行渲染,得到该第六视频帧。

在一种可能的实施方式中,该渲染模块 802,还用于获取该目标虚拟对象对应的第二动

画和第二音频。在该第二视频帧中添加与该目标虚拟对象对应的第二动画,得到第七视频帧。

该发送模块 803,还用于将该第七视频帧和该第二音频发送给该第一终端,该第一终端用于在显示该第七视频帧的同时播放该第二音频。

在一种可能的实施方式中,该装置还包括视频帧集合生成模块,该视频帧集合生成模块用于执行下述任一项:

将多个该第二视频帧聚合为第一视频帧集合,将该第一视频帧集合发送给该第一终端,该第一终端用于向其他终端分享该第一视频帧集合。

将该第二视频帧和该第一视频帧进行拼接,得到拼接视频帧。将多个该拼接视频帧聚合为第二视频帧集合,将该第二视频帧集合发送给该第一终端,该第一终端用于向其他终端分享该第二视频帧集合。

在一种可能的实施方式中,该装置还包括:

参数获取模块,用于从客户端服务器获取该第一参数,该客户端服务器用于基于该第一终端上传的针对该目标虚拟对象的设置参数,创建该目标虚拟对象的三维模型,该第一参数包括该三维模型的多个渲染参数。

需要说明的是:上述实施例提供的视频帧的渲染装置在对视频帧进行二次渲染时,仅以上述各功能模块的划分进行举例说明,实际应用中,可以根据需要而将上述功能分配由不同的功能模块完成,即将服务器的内部结构划分成不同的功能模块,以完成以上描述的全部或者部分功能。另外,上述实施例提供的视频帧的渲染装置与视频帧的渲染方法实施例属于同一构思,其具体实现过程详见方法实施例,这里不再赘述。

通过本申请实施例提供的技术方案,如果想要改变某个虚拟对象的显示效果,可以通过第一终端为该虚拟对象确定第一参数,由服务器基于该第一参数对视频帧进行二次渲染,得到第二视频帧,相较于第一视频帧,第二视频帧中显示的虚拟对象的显示效果也即是第一终端为该虚拟对象配置的显示效果。通过这样的技术方案,在云应用中提供了一种二次渲染虚拟对象的功能,用户能够快捷、高效地调整虚拟场景中某个虚拟对象的显示效果,从而扩展了云应用的功能范围,提高了云应用的个性化,使得云应用传播更加广。

下面对服务器的结构进行介绍:

图 9 是本申请实施例提供的一种服务器的结构示意图,该服务器 900 可因配置或性能不同而产生比较大的差异,可以包括一个或多个处理器 (Central Processing Units, CPU) 901 和一个或多个的存储器 902,其中,所述一个或多个存储器 902 中存储有至少一条计算机程序,所述至少一条计算机程序由所述一个或多个处理器 901 加载并执行以实现上述各个方法实施例提供的方法。当然,该服务器 900 还可以具有有线或无线网络接口、键盘以及输入输出接口等部件,以便进行输入输出,该服务器 900 还可以包括其他用于实现设备功能的部件,在此不做赘述。

在示例性实施例中,还提供了一种计算机可读存储介质,例如包括计算机程序的存储器,上述计算机程序可由处理器执行以完成上述实施例中的视频帧的渲染方法。例如,该计算机可读存储介质可以是只读存储器 (Read-Only Memory, ROM)、随机存取存储器 (Random Access Memory, RAM)、只读光盘 (Compact Disc Read-Only Memory, CD-ROM)、磁带、软盘和光数据存储设备等。

在示例性实施例中,还提供了一种计算机程序产品或计算机程序,该计算机程序产品或计算机程序包括程序代码,该程序代码存储在计算机可读存储介质中,计算机设备的处理器从计算机可读存储介质读取该程序代码,处理器执行该程序代码,使得该计算机设备执行上述视频帧的渲染方法。也即是该程序代码被处理器执行时实现上述视频帧的渲染方法。

在一些实施例中,本申请实施例所涉及的计算机程序可被部署在一个计算机设备上执行,或者在位于一个地点的多个计算机设备上执行,又或者,在分布在多个地点且通过通信网络

互连的多个计算机设备上执行，分布在多个地点且通过通信网络互连的多个计算机设备可以组成区块链系统。

本领域普通技术人员可以理解实现上述实施例的全部或部分步骤可以通过硬件来完成，也可以通过程序来指令相关的硬件完成，该程序可以存储于一种计算机可读存储介质中，上述提到的存储介质可以是只读存储器，磁盘或光盘等。

上述仅为本申请的可选实施例，并不用以限制本申请，凡在本申请的精神和原则之内，所做的任何修改、等同替换、改进等，均应包含在本申请的保护范围之内。

权利要求书

1.一种视频帧的渲染方法,所述方法包括:

服务器获取第一终端对应的第一视频帧,所述第一视频帧是以目标虚拟场景中被控虚拟对象的视角,对所述目标虚拟场景进行渲染得到的视频帧,所述被控虚拟对象为所述第一终端控制的虚拟对象;

在所述第一视频帧显示有所述第一终端的目标虚拟对象的情况下,所述服务器基于所述第一终端的第一参数,对所述第一视频帧中的所述目标虚拟对象进行渲染,得到第二视频帧,所述目标虚拟对象为所述第一终端需要重新进行渲染的虚拟对象;

所述服务器将所述第二视频帧发送给所述第一终端,所述第一终端用于显示所述第二视频帧。

2.根据权利要求1所述的方法,其中,所述在所述第一视频帧显示有所述第一终端的目标虚拟对象的情况下,所述服务器基于所述第一终端的第一参数,对所述第一视频帧中的所述目标虚拟对象进行渲染,得到第二视频帧包括:

在所述第一视频帧显示有所述第一终端的目标虚拟对象的情况下,所述服务器获取第一渲染参数,所述第一渲染参数为所述第一参数中与第一角度和第一距离对应的渲染参数,所述第一角度为所述被控虚拟对象与所述目标虚拟对象之间的角度,所述第一距离为所述被控虚拟对象与所述目标虚拟对象之间的距离;

所述服务器基于所述第一渲染参数,对所述第一视频帧中的所述目标虚拟对象进行渲染,得到所述第二视频帧。

3.根据权利要求2所述的方法,其中,所述服务器基于所述第一渲染参数,对所述第一视频帧中的所述目标虚拟对象进行渲染,得到所述第二视频帧包括:

所述服务器基于所述第一渲染参数,确定所述第一视频帧中所述目标虚拟对象的多个目标像素点的目标像素值;

所述服务器采用所述目标像素值,更新所述第一视频帧中所述多个目标像素点的像素值,得到所述第二视频帧。

4.根据权利要求1所述的方法,其中,所述在所述第一视频帧显示有所述第一终端的目标虚拟对象的情况下,所述服务器基于所述第一终端的第一参数,对所述第一视频帧中的所述目标虚拟对象进行渲染,得到第二视频帧包括:

在所述第一视频帧显示有所述第一终端的目标虚拟对象的情况下,所述服务器从所述第一参数中获取第一渲染参数和第二渲染参数;

所述服务器在所述第一视频帧中,采用所述第一渲染参数和所述第二渲染参数,分别对所述目标虚拟对象和所述被控虚拟对象进行渲染,得到所述第二视频帧。

5.根据权利要求1所述的方法,其中,所述在所述第一视频帧显示有所述第一终端的目标虚拟对象的情况下,所述服务器基于所述第一终端的第一参数,对所述第一视频帧中的所述目标虚拟对象进行渲染,得到第二视频帧包括:

在所述第一视频帧显示有所述第一终端的目标虚拟对象的情况下,所述服务器确定所述被控虚拟对象在所述目标虚拟场景中的位置;

在所述被控虚拟对象处于所述目标虚拟场景的目标子场景的情况下,所述服务器基于所述第一参数对所述第一视频帧中的所述目标虚拟对象进行渲染,得到所述第二视频帧。

6.根据权利要求5所述的方法,其中,所述方法还包括:

在所述被控虚拟对象不处于所述目标虚拟场景的目标子场景的情况下,所述服务器将所述第一视频帧确定为所述第二视频帧。

7.根据权利要求1所述的方法,其中,所述在所述第一视频帧显示有所述第一终端的目

标虚拟对象的情况下,所述服务器基于所述第一终端的第一参数,对所述第一视频帧中的所述目标虚拟对象进行渲染,得到第二视频帧之前,所述方法还包括:

所述服务器对所述第一视频帧进行图像识别,确定所述第一视频帧的类型;

所述在所述第一视频帧显示有所述第一终端的目标虚拟对象的情况下,所述服务器基于第一参数对所述第一视频帧中的所述目标虚拟对象进行渲染,得到第二视频帧包括:

在所述类型表示所述第一视频帧显示有所述目标虚拟对象的情况下,所述服务器基于所述第一参数对所述目标虚拟对象进行渲染,得到所述第二视频帧。

8.根据权利要求1所述的方法,其中,所述方法还包括:

所述服务器获取所述第一终端对应的第三视频帧,所述第三视频帧是在所述目标虚拟场景中发生目标事件后,以所述被控虚拟对象的视角对所述目标虚拟场景进行渲染得到的视频帧,所述目标事件为所述被控虚拟对象在所述目标虚拟场景中击败了第一虚拟对象;

所述服务器获取与所述目标事件对应的第一动画以及与所述目标事件对应的第一音频,所述第一音频为所述第一终端的音频;

所述服务器在所述第三视频帧中确定所述第一虚拟对象被击败的区域;

所述服务器在所述区域中添加与所述目标事件对应的第一动画,得到第四视频帧;

所述服务器将所述第四视频帧和所述第一音频发送给所述第一终端,所述第一终端用于在显示所述第四视频帧的同时播放所述第一音频。

9.根据权利要求1所述的方法,其中,所述目标虚拟场景还包括第二虚拟对象,所述第二虚拟对象为第二终端控制的虚拟对象,所述第二终端与所述第一终端为不同终端,所述方法还包括:

所述服务器获取所述第二终端对应的第五视频帧,所述第五视频帧是所述以所述第二虚拟对象的视角,对所述目标虚拟场景进行渲染得到的视频帧;

在所述第五视频帧显示有所述目标虚拟对象的情况下,所述服务器对所述第五视频帧进行目标处理,得到第六视频帧;

所述服务器将所述第六视频帧发送给所述第二终端,所述第二终端用于显示所述第六视频帧。

10.根据权利要求9所述的方法,其中,所述在所述第五视频帧显示有所述目标虚拟对象的情况下,所述服务器对所述第五视频帧进行目标处理,得到第六视频帧显示有下述任一项:

在所述第五视频帧显示有所述目标虚拟对象的情况下,所述服务器基于所述第二终端的第二参数,对所述第五视频帧中的所述目标虚拟对象进行渲染,得到所述第六视频帧;

在所述第五视频帧显示有所述目标虚拟对象的情况下,所述服务器基于所述第一参数对所述第五视频帧中的所述目标虚拟对象进行渲染,得到所述第六视频帧;

在所述第五视频帧显示有所述目标虚拟对象的情况下,所述服务器在所述第五视频帧中添加与所述目标虚拟对象对应的第二动画,得到所述第六视频帧。

11.根据权利要求9所述的方法,其中,所述被控虚拟对象和所述第二虚拟对象同时处于所述目标虚拟场景的目标子场景,所述在所述第五视频帧显示有所述目标虚拟对象的情况下,所述服务器对所述第五视频帧进行目标处理,得到第六视频帧包括:

在所述第五视频帧显示有所述目标虚拟对象的情况下,所述服务器比较所述被控虚拟对象和所述第二虚拟对象的虚拟等级;

在所述被控虚拟对象的虚拟等级高于所述第二虚拟对象的虚拟等级的情况下,所述服务器基于所述第一参数对所述第五视频帧中的所述目标虚拟对象进行渲染,得到所述第六视频帧;

在所述被控虚拟对象的虚拟等级低于所述第二虚拟对象的虚拟等级的情况下,所述服务器基于所述第二终端的第二参数对所述第五视频帧中的所述目标虚拟对象进行渲染,得到所述第六视频帧;

在所述被控虚拟对象和所述第二虚拟对象的虚拟等级相同的情况下，所述服务器基于第三参数对所述第五视频帧中的所述目标虚拟对象进行渲染，得到所述第六视频帧。

12.根据权利要求1所述的方法，其中，所述在所述第一视频帧显示有所述第一终端的目标虚拟对象的情况下，所述服务器基于所述第一终端的第一参数，对所述第一视频帧中的所述目标虚拟对象进行渲染，得到第二视频帧之后，所述方法还包括：

所述服务器获取所述目标虚拟对象对应的第二动画和第二音频；

所述服务器在所述第二视频帧中添加与所述目标虚拟对象对应的第二动画，得到第七视频帧；

所述服务器将所述第七视频帧和所述第二音频发送给所述第一终端，所述第一终端用于在显示所述第七视频帧的同时播放所述第二音频。

13.根据权利要求1-12任一项所述的方法，其中，所述将所述第二视频帧发送给所述第一终端，所述第一终端用于显示所述第二视频帧之后，所述方法还包括下述任一项：

所述服务器将多个所述第二视频帧聚合为第一视频帧集合，将所述第一视频帧集合发送给所述第一终端，所述第一终端用于向其他终端分享所述第一视频帧集合；

所述服务器将所述第二视频帧和所述第一视频帧进行拼接，得到拼接视频帧；将多个所述拼接视频帧聚合为第二视频帧集合，将所述第二视频帧集合发送给所述第一终端，所述第一终端用于向其他终端分享所述第二视频帧集合。

14.一种视频帧的渲染系统，所述系统包括：

第一终端、第一服务器以及第二服务器，所述第一终端、所述第一服务器以及所述第二服务器之间通讯相连；

所述第一服务器用于以目标虚拟场景中被控虚拟对象的视角对所述目标虚拟场景进行渲染，得到所述第一终端对应的第一视频帧，将所述第一视频帧发送给所述第二服务器，所述被控虚拟对象为所述第一终端控制的虚拟对象；

所述第二服务器用于接收所述第一视频帧；

所述第二服务器还用于在所述第一视频帧显示有所述第一终端的目标虚拟对象的情况下，基于所述第一终端的第一参数，对所述第一视频帧中的所述目标虚拟对象进行渲染，得到第二视频帧；

所述第二服务器还用于向所述第一终端发送所述第二视频帧；

所述第一终端用于响应于接收到所述第二视频帧，显示所述第二视频帧。

15.一种服务器，所述服务器包括一个或多个处理器和一个或多个存储器，所述一个或多个存储器中存储有至少一条计算机程序，所述计算机程序由所述一个或多个处理器加载并执行以实现如权利要求1至权利要求12任一项所述的视频帧的渲染方法。

16.一种计算机可读存储介质，所述计算机可读存储介质中存储有至少一条计算机程序，所述计算机程序由处理器加载并执行以实现如权利要求1至权利要求12任一项所述的视频帧的渲染方法。

17.一种计算机程序产品，包括程序代码，所述程序代码被处理器执行时实现权利要求1至权利要求12任一项所述的视频帧的渲染方法。