

(12) 按照专利合作条约所公布的国际申请

(19) 世界知识产权组织
国际局



(43) 国际公布日
2024年4月25日 (25.04.2024)

(10) 国际公布号
WO 2024/082878 A1

- (51) 国际专利分类号:
A63F 13/77 (2014.01) *G06T 15/00* (2011.01)
G06F 9/50 (2006.01)
- (21) 国际申请号: PCT/CN2023/118545
- (22) 国际申请日: 2023年9月13日 (13.09.2023)
- (25) 申请语言: 中文
- (26) 公布语言: 中文
- (30) 优先权:
202211293540.4 2022年10月21日 (21.10.2022) CN
- (71) 申请人: 腾讯科技(深圳)有限公司 (TENCENT TECHNOLOGY (SHENZHEN) COMPANY LIMITED) [CN/CN]; 中国广东省深圳市南山区高新区科技中一路腾讯大厦35层, Guangdong 518057 (CN)。
- (72) 发明人: 刘京洋 (LIU, Jingyang); 中国广东省深圳市南山区高新区科技中一路腾讯大厦

35层, Guangdong 518057 (CN)。杨衍东 (YANG, Yandong); 中国广东省深圳市南山区高新区科技中一路腾讯大厦35层, Guangdong 518057 (CN)。赵新达 (ZHAO, Xinda); 中国广东省深圳市南山区高新区科技中一路腾讯大厦35层, Guangdong 518057 (CN)。

(74) 代理人: 北京派特恩知识产权代理有限公司 (CHINA PAT INTELLECTUAL PROPERTY OFFICE); 中国北京市海淀区苏州街3号大恒科技大厦南座五层503, Beijing 100080 (CN)。

(81) 指定国(除另有指明, 要求每一种可提供的国家保护): AE, AG, AL, AM, AO, AT, AU, AZ, BA, BB, BG, BH, BN, BR, BW, BY, BZ, CA, CH, CL, CN, CO, CR, CU, CV, CZ, DE, DJ, DK, DM, DO, DZ, EC, EE, EG, ES, FI, GB, GD, GE, GH, GM, GT, HN, HR, HU, ID, IL, IN, IQ, IR, IS, IT, JM, JO, JP, KE, KG, KH, KN, KP, KR, KW, KZ, LA, LC, LK, LR, LS, LU, LY, MA, MD, MG, MK, MN, MU, MW, MX, MY, MZ, NA, NG, NI, NO, NZ, OM, PA,

(54) Title: RENDERING PROCESSING METHOD AND APPARATUS, ELECTRONIC DEVICE, COMPUTER-READABLE STORAGE MEDIUM, AND COMPUTER PROGRAM PRODUCT

(54) 发明名称: 渲染处理方法、装置、电子设备、计算机可读存储介质及计算机程序产品

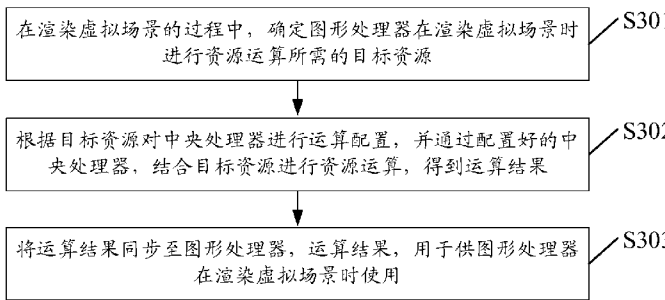


图 3

- S301 In a process of rendering a virtual scene, determine a target resource required by a graphics processing unit to perform resource operation when rendering the virtual scene
- S302 Perform operation configuration on a central processing unit according to the target resource, and perform resource operation by means of the configured central processing unit and according to the target resource, so as to obtain an operation result
- S303 Synchronize the operation result to the graphics processing unit, the operation result being used for the graphics processing unit to use when rendering the virtual scene

(57) Abstract: Embodiments of the present application provide a rendering processing method and apparatus, a device, and a medium. The method comprises: in a process of rendering a virtual scene, determining a target resource required by a graphics processing unit to perform resource operation when rendering the virtual scene; performing operation configuration on a central processing unit according to the target resource, and performing resource operation by means of the configured central processing unit and according to the target resource, so as to obtain an operation result; synchronizing the operation result to the graphics processing unit, the operation result being used for the graphics processing unit to use when rendering the virtual scene.

PE, PG, PH, PL, PT, QA, RO, RS, RU, RW, SA, SC, SD, SE, SG, SK, SL, ST, SV, SY, TH, TJ, TM, TN, TR, TT, TZ, UA, UG, US, UZ, VC, VN, WS, ZA, ZM, ZW。

- (84) 指定国(除另有指明, 要求每一种可提供的地区保护): ARIPO (BW, CV, GH, GM, KE, LR, LS, MW, MZ, NA, RW, SC, SD, SL, ST, SZ, TZ, UG, ZM, ZW), 欧亚 (AM, AZ, BY, KG, KZ, RU, TJ, TM), 欧洲 (AL, AT, BE, BG, CH, CY, CZ, DE, DK, EE, ES, FI, FR, GB, GR, HR, HU, IE, IS, IT, LT, LU, LV, MC, ME, MK, MT, NL, NO, PL, PT, RO, RS, SE, SI, SK, SM, TR), OAPI (BF, BJ, CF, CG, CI, CM, GA, GN, GQ, GW, KM, ML, MR, NE, SN, TD, TG)。

本国际公布:

— 包括国际检索报告(条约第21条(3))。

(57) 摘要: 本申请实施例提供了一种渲染处理方法、装置、设备及介质, 其中的方法包括: 在渲染虚拟场景的过程中, 确定图形处理器在渲染虚拟场景时进行资源运算所需的目标资源; 根据目标资源对中央处理器进行运算配置, 并通过配置好的中央处理器, 结合目标资源进行资源运算, 得到运算结果; 将运算结果同步至图形处理器, 运算结果, 用于供图形处理器在渲染虚拟场景时使用。

渲染处理方法、装置、电子设备、计算机可读存储介质及计算机程序产品

相关申请的交叉引用

本申请实施例基于申请号为 202211293540.4、申请日为 2022 年 10 月 21 日的中国专利申请提出，并要求该中国专利申请的优先权，该中国专利申请的全部内容在此引入本申请实施例作为参考。

技术领域

本申请涉及计算机技术领域，尤其涉及一种渲染处理方法、装置、电子设备、计算机可读存储介质以及计算机程序产品。

背景技术

10 随着计算机技术的快速发展，各种类型的应用程序层出不穷；如应用程序为游戏类型的应用程序，可包括：游戏渲染和计算均在云端服务器的云游戏。

计算机设备主要是依赖于设备中部署的图形处理器（Graphics Processing Unit, GPU）和中央处理器（Central Processing Unit, CPU），来实现针对游戏应用程序的游戏画面渲染的。经实践发现，在游戏运行过程中，游戏渲染占用图形处理器资源的占用率往往高于中央处理器资源的占用率，如在图形处理器资源不足的时候，中央处理器往往相对空闲，导致图形处理器的运行压力较大，从而可能降低游戏的运行效率。

发明内容

本申请实施例提供一种渲染处理方法、装置、设备及介质，能够减小图形处理器的运行压力，从而提升虚拟场景的运行效率。

20 本申请实施例提供了一种渲染处理方法，该方法包括：

在渲染虚拟场景的过程中，确定图形处理器在渲染所述虚拟场景时进行资源运算所需的目标资源；

根据所述目标资源对中央处理器进行运算配置，并通过配置好的所述中央处理器，结合所述目标资源进行资源运算，得到运算结果；

25 将所述运算结果同步至所述图形处理器，所述运算结果，用于供所述图形处理器在渲染所述虚拟场景时使用。

本申请实施例提供了一种渲染处理装置，该装置包括：

获取单元，配置为在渲染虚拟场景的过程中，确定图形处理器在渲染所述虚拟场景时进行资源运算所需的目标资源；

30 处理单元，配置为根据所述目标资源对中央处理器进行运算配置，并通过配置好的所述中央处理器，结合所述目标资源进行资源运算，得到运算结果；

所述处理单元，还配置为将所述运算结果同步至所述图形处理器，所述运算结果，用于供所述图形处理器在渲染所述虚拟场景时使用。

本申请实施例提供一种计算机设备，该设备包括：

35 处理器，适于执行计算机程序；

计算机可读存储介质，计算机可读存储介质中存储有计算机程序，计算机程序被处理器执行时，实现如上述的渲染处理方法。

本申请实施例提供一种计算机可读存储介质，该计算机可读存储介质存储有计算机程序，计算机程序适于由处理器加载并执行如上述的渲染处理方法。

本申请实施例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计算机指令，该计算机指令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计算机指令，处理器执行该计算机指令，使得该计算机设备执行上述的渲染处理方法。

5 本申请实施例中，首先通过图形处理器以及中央处理器协同处理，从而在不增加硬件成本的情况下，提升虚拟场景的运行效率，提高目标对象在虚拟场景中的体验；其次，在图形处理器渲染虚拟场景的过程中，当图形处理器的负载率较高时，将图形处理器中的计算着色器进行资源运算时所需的目标资源拷贝至中央处理器，从而通过中央处理器代替图形处理器，执行对目标资源运算得到运算结果；再由中央处理器将运算结果同步至图形处理器中，确保
10 数据的一致性，以便于图形处理器基于运算结果继续进行虚拟场景的渲染处理。如此，在图形处理器的负载率较高时，将目标资源转移至负载率较低的中央处理器进行处理，通过这样削峰填谷方式，在减小图形处理器的运行压力的同时，还能确保图形处理器和中央处理器均能以较高的吞吐运行虚拟场景，从而进一步提升虚拟场景的运行效率，提高目标对象在虚拟场景中的体验。

15

附图说明

图 1 是本申请一个示例性实施例提供的一种渲染流程的示意图；

图 2A 是本申请一个示例性实施例提供的一种客户端游戏的游戏场景示意图；

图 2B 是本申请一个示例性实施例提供的一种云游戏的游戏场景示意图；

20 图 3 是本申请一个示例性实施例提供的一种渲染处理方法的流程示意图；

图 4 是本申请一个示例性实施例提供的一种根据目标资源对中央处理器进行运算配置的流程示意图；

图 5 是本申请一个示例性实施例提供的一种渲染处理方法的完整技术流程；

图 6 是本申请一个示例性实施例提供的另一种渲染处理方法的流程示意图；

25 图 7 是本申请一个示例性实施例提供的一种将纹理资源从图形处理器拷贝至中央处理器的示意图；

图 8 是本申请一个示例性实施例提供的一种渲染处理装置的结构示意图；

图 9 是本申请一个示例性实施例提供的一种计算机设备的结构示意图。

30 具体实施方式

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

35 本申请实施例涉及计算机图形学 (Computer Graphics, CG)。计算机图形学是一种使用数学算法将二维或三维图像，转化为计算机显示器的栅格形式的科学；可以是研究如何在计算机中表示图像，以及利用计算机进行图像计算、处理和显示的相关原理和算法。计算机图形学涉及的一道重要工具是渲染技术；所谓渲染是指将计算机中抽象的模型转换为直观可见的图像的过程；渲染作为计算机图形学中的最后一道工具，可实现将模型转换为图像，并最终呈现在计算机屏幕上。其中，上述提及的模型可是指用语言或数据结构进行严格定义的三维物体或虚拟场景，可以包括但不限于：几何（如物体形状等）、视点（即图像中摄像机的光心）、纹理（如物理表面呈现凹凸不平的沟纹或图案等）及照明（如通过光线实现图像中的阴影效果等）等信息。

40

随着计算机图形学的不断深入发展，渲染技术的应用越来越广泛。例如，在计算机设备（如智能终端或智能电脑等设备）中运行目标应用程序时，往往需要将应用程序提供的服务页面渲染显示到显示屏幕上，以便于目标对象（如持有计算机设备的任一对象）能够直观获取服务页面中显示的内容。此处的目标应用程序是指为完成某项或多项特定工作的计算机程序；按照目标应用程序的功能可分为游戏类型的应用程序、文本类型的应用程序或通信类型的应用程序等；按照目标应用程序的运行方式可分为客户端、小程序、网页应用等；本申请实施例对目标应用程序的类型不作限定。为便于阐述，后续以目标应用程序为游戏类型的应用程序（可称为游戏应用程序），如待渲染显示的图像为游戏应用程序提供的虚拟场景（如任一游戏）的游戏画面为例进行阐述，特在此说明。

在实际应用中，计算机设备依赖于其部署的硬件“中央处理器（Central Processing Unit, CPU）”和“图形处理器（Graphics Processing Unit, GPU）”，来实现渲染显示图像（或页面、画面（如游戏画面）等）。计算机设备依赖于中央处理器和图形处理器执行渲染处理的流程可至少包括：应用阶段→几何阶段→光栅化阶段。其中：①应用阶段由中央处理器主导，是开发者进行场景开发的阶段；在应用阶段可以设置图像中的摄像机以及灯光等，形成渲染图像所需的点、线、面以及纹理等数据（或信息）。在应用阶段得到渲染图像所需的数据后，可将数据从中央处理器转移至图形处理器，由图形处理器基于数据执行几何阶段。

②几何阶段是由图形处理器主导，图形处理器在几何阶段可将中央处理器在应用阶段发来的数据进行进一步处理。如图1所示，图形处理器中包含渲染管线，渲染管线是图形处理器中负责给图像配上颜色的一组通道；渲染管线可被划分为几个阶段，每个阶段会把前一个阶段的输出作为当前阶段的输入，使得渲染管线类似于流水线一样执行工作。由于渲染管线上各个阶段具有并行执行的特性，因此显卡（包含图形处理器的设备）中的小处理器可以在图形处理器上为每一个阶段运行各自的小程序，从而在渲染管线中快速处理数据，这些小程序可以叫做着色器（shader），不同着色器可实现不同功能；如图1所示的渲染管线中包括顶点着色器及几何着色器等，当前着色器的输入为前一个着色器的输出。在显卡上运行的小程序（即着色器）是由DirectX（DirecteXtension）提供的；DirectX是一种应用程序接口，可作为Windows（操作系统）下的渲染接口（Application Programming Interface, API），可包括但不限于：DirectX9、DirectX10、DirectX11或DirectX12，不同的DirectX对应不同的库文件（Dynamic-Link Libraries, DLL）。

图形处理器中还包独立于渲染管线中的单独管线，该单独管线包括计算着色器（Compute Shader）。计算着色器是运行在图形处理器中的通用计算逻辑，在渲染流程中对渲染管线的任一阶段进行读取操作，以执行较为纯粹的渲染计算任务。图形处理器有了计算着色器的计算能力后，就不再是光栅化中的“模拟”计算了，而是专业级的计算能力，计算速度更快且更加的灵活，速度更快。

③光栅化阶段也是由图形处理器主导，在此阶段可接收上一阶段（即几何阶段）传输的数据，并基于接收到的数据产生屏幕上的像素，并渲染出最终的图像（或画面）；光栅化阶段的主要任务是确定每个渲染图元（即图像中的元素）中的哪些像素应该被绘制在显示屏幕上，以实现将数据或信息渲染显示为终端屏幕中的图像。

通过上述描述的渲染流程不难发现，图形处理器在整个渲染流程中承担图像渲染的海量数学运算，然而中央处理器在整个渲染流程中只承担了应用阶段的数据准备。因而在很多图像渲染场景中，图形处理器的占用率（或使用率、负载率）往往高于中央处理器的占用率；图形处理器的占用率可是指图形处理器被使用资源（如内存空间）与总资源的比值，同理中央处理器的占用率可是指中央处理器被使用资源与总资源的比值；这导致图形处理器和中央处理器的资源使用不均衡，如图形处理器资源不足时，可能中央处理器处于较空闲状态。基于此，本申请实施例提出一种渲染处理方案，该方案支持将图形处理器所负责的部分工作转移至中央处理器，由中央处理器来代替图形处理器执行这部分工作；通过部分算力的转移，

使得中央处理器承担部分图形处理器的工作，在一定程度上降低图形处理器的运行压力，从而使图像渲染得到更好的表现。

在实际实施时，参见前述描述的渲染流程可知，计算着色器在渲染流程中负责的是较为纯粹的计算任务，该计算任务可以在虚拟场景运行的任意时间进行启动或停止，而不会影响虚拟场景的运行逻辑。因此，本申请实施例支持将图形处理器包含的渲染管线拆分为两部分：计算着色器和正常管线（如顶点着色器或几何着色器等）；然后，将图形处理器中计算着色器这部分算力转移至中央处理器，也就是说，从图形处理器中拆分出来的计算着色器可耗费中央处理器资源执行计算任务。在中央处理器执行完成计算着色器运算后，再将计算结果（或运算结果）返回至图形处理器中；计算结果拷贝至图形处理器后，即是拷贝至图形处理器管线后，图形处理器就像计算着色器已经在图形处理器中执行完成一样，得到了期望结果，从而图形处理器可以在渲染虚拟场景时使用该计算结果。

本申请实施例提供的渲染处理方案的主要原理可包括：在图形处理器渲染虚拟场景的过程中，当检测到图形处理器负载率较大，而中央处理器负载率较小时，确定可以启动计算迁移，此时可截断虚拟场景的计算着色器相关的调用，并确定图形处理器渲染虚拟场景时进行资源运算所需的目标资源；该目标资源可包括图形处理器中任一着色器或其他装置进行资源运算所需的资源，可包括图形处理器中的计算着色器进行资源运算所需的资源（如纹理资源、几何资源及顶点资源等）。然后，根据该目标资源对中央处理器进行运算配置，如在中央处理器中创建用于进行计算着色器计算的渲染设备。再将计算着色器的相关资源（如目标资源）转移至中央处理器中创建的渲染设备等；并利用渲染设备在中央处理器中运行的特点，将渲染工作从图形处理器转移到中央处理器，即使用配置好的中央处理器对目标资源进行运算，得到运算结果。最后，在渲染设备的计算着色器渲染结束后，需将运算结果同步至图形处理器中，保证数据的一致性，以使图形处理器在渲染虚拟场景时使用该运算结果。

通过本申请实施例，首先通过图形处理器以及中央处理器协同处理，从而在不增加硬件成本的情况下，提升虚拟场景的运行效率，提高目标对象在虚拟场景中的体验；其次，在图形处理器渲染虚拟场景的过程中，当图形处理器的负载率较高时，将图形处理器中的计算着色器进行资源运算时所需的目标资源拷贝至中央处理器，从而通过中央处理器代替图形处理器，执行对目标资源运算得到运算结果；再由中央处理器将运算结果同步至图形处理器中，确保数据的一致性，以便于图形处理器基于运算结果继续进行虚拟场景的渲染处理。如此，在图形处理器的负载率较高时，将目标资源转移至负载率较低的中央处理器进行处理，通过这样削峰填谷方式，在减小图形处理器的运行压力的同时，还能确保图形处理器和中央处理器均能以较高的吞吐运行虚拟场景，从而进一步提升虚拟场景的运行效率，提高目标对象在虚拟场景中的体验。

需要说明的是，本申请实施例是以迁移图形处理器中计算着色器这部分算力为例进行阐述的；但在实际场景中，若图形处理器中还包括其他可以迁移的算力，那么也可以迁移其他算力，本申请实施例对此不作限定。

本申请实施例提供的渲染处理方案可应用于不同应用场景；根据虚拟场景的运行方式不同，应用场景并不相同。虚拟场景可按照不同运行方式进行分类，可包括虚拟游戏场景，如本地游戏（或称为客户端游戏）和云游戏，虚拟购物场景或者音视频应用场景。

在实际应用中，不管虚拟场景是虚拟游戏场景，还是虚拟购物场景，或者是音视频应用场景，对于目标对象而言，只需在其持有的终端设备中部署集成了渲染处理方案的本地软件，就可以从本地软件拉起虚拟场景，虚拟场景就会自动判断是否需要启动计算着色器从图形处理器到中央处理器的计算迁移，即执行本申请实施例提供的计算迁移的方案。其中，该本地软件中集成了渲染处理方案的整体执行逻辑和数据库等信息；并且，该本地软件的打开和运行无需网络流量要求，目标对象可以在需要时随时使用该本地软件提供的计算迁移，在一定程度上提升渲染处理方案的普适性。

下面以虚拟场景为虚拟游戏场景，如本地游戏和云游戏为例，对本申请实施例涉及的游戏应用场景进行示例性介绍，其中：

1) 本地游戏可是指：通过下载游戏安装包至终端设备，并在终端设备的本地运行的游戏。本地游戏的游戏画面渲染过程是由计算机设备来执行的，也就是说，计算机设备不仅负责渲染本地游戏的游戏画面，还负责显示本地游戏的游戏画面。在虚拟场景为本地游戏的游戏应用场景下，上述提及的执行渲染处理方案的计算机设备可包括与目标对象（如游戏玩家）进行交互的终端设备，终端设备可包括但不限于：智能手机（如部署安卓（Android）系统的智能手机，或部署互联网操作系统（Internetworking Operating System, IOS）的智能手机）、平板电脑、便携式个人计算机、移动互联网设备（Mobile Internet Devices, MID）、智能电视、车载设备、头戴设备等携带显示屏幕的智能终端。

如图 2A 所示，在虚拟场景为终端设备 201 中部署的本地游戏的场景中，终端设备 201 通过与终端设备中部署的虚拟场景对应的服务器 202 进行数据交互，以实现运行虚拟场景；如终端设备 201 接收服务器 202 发送的游戏数据，并由终端设备 201 基于接收到的游戏数据进行游戏画面的渲染显示。此游戏应用场景中的服务器可以包括但不限于：数据处理服务器、全球广域网（World Wide Web, Web）服务器、应用服务器等等具有复杂计算能力的设备；或者，服务器可以是独立的物理服务器，也可以是由多个物理服务器构成的服务器集群或者分布式系统。

在虚拟场景为本地游戏的由此场景下（如客户端独立运行虚拟游戏场景的模式下），通过本申请实施例提供的计算迁移过程，可在保持资源运算的计算量不减少的情况调整计算资源的组成，将计算着色器从图形处理器渲染管线中拆分出来使用中央处理器进行计算，从而降低对计算机设备（即终端设备）硬件配置的要求，实现使用较低配置的显卡就可以运行较高渲染质量的虚拟游戏场景，在一定程度上可提高终端设备运行大型游戏的速率，使得虚拟游戏场景的运行效果更好。

2) 云游戏又可称为游戏点播（gaming on demand），是指在计算机设备中运行的游戏，此时的计算机设备包括云服务器（或称为云端服务器）。如图 2B 所示，在虚拟场景为云游戏的游戏应用场景下，虚拟场景可将外设设备发送的数据（如输入指令或信号）发送至云服务器，由云游戏服务器根据数据负责渲染游戏画面，并将渲染完毕的游戏画面压缩后，通过网络传送给操作对象所使用的终端设备，此时终端设备仅做显示游戏画面的操作。

云游戏的这种运行模式，使得游戏玩家持有的终端设备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（如显示游戏画面的能力）、人机交互能力（获取操作对象的输入操作的能力），以及数据传输能力（如发送指令给云服务器的能力）即可。

在虚拟场景为云游戏的由此场景下（即纯服务端计算的云游戏模式），本申请实施例提供的渲染处理方案，相比于传统的云游戏渲染而言具有明显优势。这里，在传统云游戏模式下，针对某个云游戏的运行中央处理器和图形处理器是合理配比的，两者的计算能力刚好匹配该云游戏；如果运行中的云游戏，由于更新修改中央处理器和图形处理器的配比模型，例如图形处理器占用忽然大幅度增大，就会导致运行该云游戏的云服务器可以运行的路数（即运行云游戏的数量）变低，而该云服务器的中央处理器资源就会被大量浪费。然而，通过本申请实施例提供的渲染处理方案（即计算迁移的方案），可实现在特定的硬件环境下，动态的调整中央处理器和图形处理器算力的分布，同一款硬件的中央处理器资源不会被大幅度浪费，会用来分担一部分图形处理器算力，使得服务端的硬件搭配更加灵活，从而达到更高的运行路数，节省硬件成本。并且，使得同一硬件配置可以支持多种不同需求的游戏计算需求，提高硬件方案的适应能力，如同一个硬件方案可以用于更多的游戏情况，提高硬件方案的寿命。

本申请实施例还需说明如下两点：①本申请实施例对终端设备中运行的虚拟游戏场景的实际类型不作限定；如虚拟游戏场景可以为本地游戏，此时执行渲染处理方案的计算机设备为目标对象持有的终端设备，即由终端设备执行渲染处理方案，前述提及的中央处理器和图

形处理器均是指部署于终端设备中的装置。同理，如虚拟游戏场景为云游戏，此时执行渲染处理方案的计算机设备为云服务器，即由云服务器执行渲染处理方案，前述提及的中央处理器和图形处理器均是指部署于云服务器中的装置。②本申请实施例运用到实际的游戏产品或技术中时，如目标应用程序获取目标对象操作虚拟游戏场景时的游戏数据时，需要获得目标对象的许可或者同意；且相关数据的收集、使用和处理需要遵守相关地区的相关法律法规和标准，如目标应用程序提供的游戏装备需要遵循相关地区的相关法律法规和标准。③本申请实施例是以渲染处理方案的应用场景为游戏应用场景为例进行阐述；但在实际应用中，本申请实施例提供的渲染处理方案的应用场景并不仅限于游戏应用场景。例如，本申请实施例提供的渲染处理方案还支持应用于音视频应用场景；如在直播应用场景中，可以将图形处理器渲染直播画面的资源运算转移至中央处理器，由中央处理器代替图形处理器进行资源运算，从而使得直播设备使用较低配置的显卡，也能获得较高质量的直播画面，提升直播体验；又或者，本申请实施例提供的渲染处理方案还支持应用于网络购物应用场景，如在网络购物应用场景中，可以将图形处理器渲染购物画面的资源运算转移至中央处理器，由中央处理器代替图形处理器进行资源运算，从而使得直播设备使用较低配置的显卡，也能获得较高质量的购物画面，提升用户的购物体验。

基于上述描述的渲染处理方案，本申请实施例提出更为详细的渲染处理方法，下面将结合附图对本申请实施例提出的渲染处理方法进行详细介绍。请参见图 3，图 3 示出了本申请一个示例性实施例提供的一种渲染处理方法的流程示意图；该渲染处理方法可以由前述提及的计算机设备执行，该方法可包括步骤 S301-S303：

S301，在渲染虚拟场景的过程中，确定图形处理器在渲染虚拟场景时进行资源运算所需的目标资源。

目标资源可是指渲染虚拟场景时渲染管线所需使用的资源，包括但不限于：纹理、网格模型、着色器常量（如入射光方向或入射光颜色等）、以及帧缓存等。正如前述所描述的，本申请实施例主要是将部分图形处理器算力迁移至中央处理器，由中央处理器代替图形处理器承担被迁移的部分算力，那么根据待迁移算力的不同，需要迁移的目标资源也不相同。考虑到计算着色器在渲染流程中负责的是较为纯粹的计算任务，该计算任务可以在虚拟场景运行的任意时间进行启动或停止，而不会影响虚拟场景的运行逻辑；因此，本申请实施例以待迁移算力为计算着色器的算力为例，也就是说，让计算着色器耗费中央处理器进行资源运算；此实现方式下，目标资源可包括：在图形处理器渲染虚拟场景时，图形处理器中的计算着色器进行资源运算所需的资源，可包括缓冲区（如顶点、索引与常量）或纹理等类型的资源。

为实现算力直接从图形处理器移动至中央处理器，需要在计算迁移启动后，截断虚拟场景对于计算着色器相关的调用，以避免继续通过图形处理器中的计算着色器执行资源运算的操作。其中，计算着色器相关的调用可是指：计算着色器执行资源运算所需条件的调用，包括但不限于：资源准备（为计算着色器执行资源运算准备资源）、管线分配（如设置计算着色器）和分发（如执行运算）等。本申请实施例支持采用库文件（Dynamic-Link Libraries, DLL）替换的方式，来实现上述提及的截断虚拟场景对于计算着色器相关的调用的操作。其中，库文件是动态链接库文件，又称为应用程序扩展，是软件文件类型；在 Windows 中许多应用程序并不是完整的可执行文件，而是被分割为一些相互独立的 DLL 文件，放置于系统中；当执行某个程序时相应 DLL 文件就会被调用。换句话说，DLL 文件是一种可执行文件，它允许程序共享执行特殊任务所必需的代码和其他资源，即 DLL 文件中包含了允许基于 Windows 的程序在 Windows 环境下操作的许多函数和资源。

基于上述对库文件的相关介绍，采用库文件替换，实现截断虚拟场景针对计算着色器相关的调用的实现过程可包括：

首先，获取本申请实施例提供的第二库文件，该第二库文件用于指示在渲染虚拟场景时通过中央处理器进行资源运算；也就是说，该第二库文件中包含通过中央处理器执行资源运

算所需的相关代码,通过运行该相关代码,可实现耗费中央处理器资源针对目标资源的资源运算。第二库文件可以是开发对象预先编写好,存放于本地软件中的;当需要采用库文件替换,实现截断虚拟场景针对计算着色器相关的调用时,虚拟场景可从本地软件中获取到该第二库文件。其中,关于本地软件的相关介绍可参见前述相关描述,在此不作赘述。

5 然后,采用第二库文件替换图形处理器中的原生库文件(在本申请实施例中可称为第一库文件),该第一库文件用于指示在渲染虚拟场景时通过图形处理器进行资源运算,该第一库文件中包含有目标函数,第一库文件可调用该目标函数获取到图形处理器在渲染虚拟场景时进行资源运算所需的目标资源。换言之,本申请实施例通过将虚拟场景运行所需要加载的第一库文件替换为第二库文件,可实现截断第一库文件对目标函数的函数调用,由替换后
10 的第二库文件实现对目标函数的函数调用;也就是说,将第一库文件对目标函数的函数调用,替换为第二库文件对目标函数的函数调用。这样可由替换后的第二库文件调用函数执行后续处理,如图形处理器通过该第二库文件调用目标函数,获取图形处理器在渲染虚拟场景时进行资源运算所需的目标资源,从而达到截断图形处理器通过原生库文件调用目标函数执行资源运算的操作。

15 最后,通过图形处理器调用第二库文件,以确定出图形处理器在渲染虚拟场景时进行资源运算所需的目标资源;可以通过第二库文件调用目标函数时,可获取到图形处理器在渲染虚拟场景时进行资源运算所需的目标资源(如纹理资源及顶点资源等)。

通过上述描述的库文件替换的实施过程,可实现在图形处理器中截断原生库文件的函数调用,截断的内容由本申请实施例提供的第二库文件来处理,如由第二库文件执行函数调用,
20 从而实现将原本的耗费图形处理器进行计算着色器的相关运算逻辑,迁移至中央处理器,即通过耗费中央处理器进行计算着色器的相关运算。通过这种库文件替换方式,可实现快速截断虚拟场景对计算着色器相关的调用,且操作简单快捷。

S302,根据目标资源对中央处理器进行运算配置,并通过配置好的中央处理器,结合目标资源进行资源运算,得到运算结果。

25 由前述描述可知,本申请实施例支持将部分图形处理器算力迁移至中央处理器,如从图形处理器渲染管线中拆分出来的计算着色器,可耗费中央处理器进行计算;那么需要在中央处理器中创建渲染设备,以实现由中央处理器中创建的渲染设备承担与计算着色器相关的资源运算。其中,渲染设备是一种具备完整渲染功能的设备,本申请实施例提供的渲染设备可包括图形变换装置(warp设备)。warp设备(warpdevice)是一种高效的中央处理器渲染设备,可以模拟DirectX特性进行渲染处理;可以是使用中央处理器渲染的DirectX管线,从而
30 实现使用中央处理器渲染完整的游戏渲染需求。当然,在实际应用中,在中央处理器中创建的渲染设备的类型可以发生变化,本申请实施例对此不作限定。

在实际应用中,在中央处理器中创建使用中央处理器进行渲染计算的图形变换装置,该图形变换装置中包括上下文(或称为设备上下文),可用于记录图形变换装置中的渲染管线的
35 的管线状态(如绑定新的资源、绑定新的着色器、修改固定功能阶段设置等)。然后,将在图形处理器渲染虚拟场景时进行资源运算所需的目标资源,配置到中央处理器中的图形变换装置,即是配置到图形变换装置的上下文中,以实现根据目标资源对中央处理器进行的运算配置。

在实际实施时,在对中央处理器进行运算配置结束后,可通过配置好的中央处理器,结合
40 目标资源进行资源运算,即是通过在中央处理器中配置好的图形变换装置,结合目标资源进行资源运算,得到运算结果。需要说明的是,根据目标资源的类型不同,对目标资源执行的运算过程也不相同;例如,目标资源包括纹理资源,对纹理资源进行的运算可包括纹理模糊运算;本申请实施例对针对目标资源的运算的实施过程不作限定,特在此说明。

上述根据目标资源对中央处理器进行运算配置的流程示意图可参见图4;如图4所示,
45 计算机设备中运行的虚拟场景依赖于计算机设备中部署的中央处理器和图形处理器进行画面渲染显示。这里,在图形处理器渲染虚拟场景的过程中,若确定启动计算迁移(即算力从

图形处理器转移至中央处理器),则可以确定图形处理器在渲染虚拟场景时进行资源运算所需的目标资源,即是与计算着色器相关的目标资源(或计算着色器进行资源运算所需的目标资源)。然后,在中央处理器中创建使用中央处理器进行渲染计算的图形变换装置。最后,将目标资源从图形处理器配置到中央处理器,即是将目标资源配置到中央处理器中的图形变换装置管线中的上下文中,以便于配置有目标资源的图形变换装置代替图形处理器执行资源运算等相关处理。

还需说明的是,传统的 Windows 管线支持根据需要对计算着色器的运算结果进行缓存,这使得在 DirectX 的计算着色器是异步运行时,也不需要单独的 flush 和阻塞等待计算着色器的运算结果,只需根据需要从缓存中获取相应运算结果即可。其中,DirectX 的计算着色器异步运算可简单理解为,计算着色器的计算是混杂在普通的渲染指令中的,如在普通执行任一个或多个普通渲染指令时,均可以执行计算着色器运行。但是,在本申请实施例涉及的拆分模式下,也就是将图形处理器中的计算着色器拆分至中央处理器进行资源运算的模式下,是期望图形处理器中计算着色器的运算结果可以立即同步到图形处理器中(可以是图形处理器上运行的管线中)。因此,在图形处理器渲染虚拟场景的过程中,当检测到在图形处理器中调用计算着色器分发与虚拟场景相关的资源运算时,就需要将上述图形处理器中的资源运算,截断为图形处理器调用 warp 设备(即图形变换装置)中的上下文进行分发计算着色器;此时,还需要对 warp 设备进行刷新(flush)和阻塞等待计算着色器的运算结果,以确保能够及时获取到计算着色器的运算结果,从而快捷的将图形处理器中计算着色器的运算结果同步至图形处理器中。

S303,将运算结果同步至图形处理器,运算结果,用于供图形处理器在渲染虚拟场景时使用。

基于前述步骤实现在中央处理器中对目标资源进行运算,得到运算结果后,还需将运算结果同步回图形处理器中,以保持中央处理器和图形处理器两端的数据一致性;使得图形处理器接收到该运算结果后类似于是图形处理器自身执行资源运算得到的运算结果,从而图形处理器可基于该运算结果执行后续渲染处理。

其中,DirectX 在计算 Compute Shader 的过程中,需要用到的目标资源可包括目标资源视图对应的资源,目标资源视图是渲染管线工作过程中所使用的一类资源的视图。在某一个或一类资源需要配置到渲染管线的不同阶段时,如纹理资源需要配置到像素着色器,还需配置到片云着色器时,需要为资源创建资源视图;该资源视图中可保存该资源的显存地址,这样通过将资源视图配置到渲染管线上,渲染管线上相应阶段就可以通过资源视图(可以是资源视图中保存的资源的显存地址)间接访问该资源。

在本申请实施例中,需要从中央处理器拷贝至图形处理器的目标资源关联的目标资源视图可至少包括:着色器资源视图(ShaderResourceView)和无序访问视图(UnorderedAccessView)。其中:①着色器资源视图可引用缓冲区或纹理资源,绑定到计算着色器阶段。着色器资源视图对于计算着色器的执行过程是只读的,即是图形变换装置只具有对着色器资源视图的读取权限,而不具有对着色器资源视图的写入权限;但考虑到每次调用计算着色器的时候,着色器资源视图对应的同一份资源的内容(即资源包含的子资源或数据)是可能发生改变。基于此,本申请实施例仍然设计将着色器资源视图对应的资源,从图形处理器拷贝至中央处理器的流程;这种将着色器资源视图对应的资源拷贝至中央处理器的拷贝成本是可以接受的,因为计算着色器是计算密集型的,也就是说资源的拷贝比起计算的开销更小,在一定程度上可节省成本。②无序访问视图可引用缓冲区或纹理资源,绑定到计算着色器阶段或片元着色器阶段。无序访问资源视图对于计算着色器的执行过程是可读写的,即是图形变换装置具有对无序访问资源视图的读写权限,读写权限包括:读取权限和写入权限。

在 DirectX 计算 Compute Shader 的过程中,还涉及的目标资源视图可包括以下至少一个:取样视图(Samplers)或着色器视图(Shader)。其中,取样视图和着色器视图对于计算着色

器的执行过程是只读的，即是图形变换装置只具有对着色器资源视图的读取权限，而不具有对着色器资源视图的写入权限。并且，取样视图对应的资源和着色器视图对应的资源是固定不变的小资源（即资源容量（如资源所能承载的最大数据量）较小），这样取样视图对应的资源和着色器视图对应的资源，可同时存在于中央处理器中和图形处理器中，且在中央处理器中和图形处理器中同时使用也是没有成本的。因此，在图形处理器中存在取样视图对应的资源和着色器视图对应的资源的情况下，取样视图对应的资源和着色器视图对应的资源，也可以不从图形处理器拷贝至中央处理器，此实现方式，需要从中央处理器拷贝至图形处理器的目标资源就包括：着色器资源视图对应的资源以及无序访问视图对应的资源，这在一定程度上可减少资源拷贝成本。

基于上述对目标资源以及目标资源视图的相关描述可知，在计算着色器的执行过程中，上述提及的四种目标资源视图中存在三种目标资源视图（着色器资源视图、取样视图以及着色器视图）均是只读的，只有一种目标资源视图（无序访问视图）是可读写的；这使得在 Compute Shader 下，上述四种目标资源视图中，只有无序访问资源视图对应的一个或多个资源可能被修改；换句话说，中央处理器执行完 Compute Shader 需要同步到图形处理器的结果只有无序资源视图对应的资源。基于此，采用配置好的图形处理器对目标资源进行运算后，待同步至图形处理器管线的运算结果包括：对无序访问视图对应的资源进行运算得到的资源结果；此时可将无序访问视图对应的资源进行运算得到的资源结果，从中央处理器同步至图形处理器管线中；图形处理器在接收到该资源结果后，就像计算着色器已经在图形处理器中执行完成一样，得到了期望的资源结果，从而图形处理器可以在渲染虚拟场景过程中使用该资源结果。

本申请实施例中，首先通过图形处理器以及中央处理器协同处理，从而在不增加硬件成本的情况下，提升虚拟场景的运行效率，提高目标对象在虚拟场景中的体验；其次，在图形处理器渲染虚拟场景的过程中，当图形处理器的负载率较高时，将图形处理器中的计算着色器进行资源运算时所需的目标资源拷贝至中央处理器，从而通过中央处理器代替图形处理器，执行对目标资源运算得到运算结果；再由中央处理器将运算结果同步至图形处理器中，确保数据的一致性，以便于图形处理器基于运算结果继续进行虚拟场景的渲染处理。如此，在图形处理器的负载率较高时，将目标资源转移至负载率较低的中央处理器进行处理，通过这样削峰填谷方式，在减小图形处理器的运行压力的同时，还能确保图形处理器和中央处理器均能以较高的吞吐运行虚拟场景，从而进一步提升虚拟场景的运行效率，提高目标对象在虚拟场景中的体验。另外，部分算力从图形处理器移动到中央处理器的方式，可通过增加中央处理器的开销来降低图形处理器的开销，从而在客户端虚拟场景下可降低终端设备的显卡要求，以及，在纯服务器虚拟场景下可合理化服务器的硬件配比（即中央处理器资源和图形处理器资源的配比）。

本申请实施例提供的渲染处理方法的完整技术流程可参见图 5。如图 5 所示，本申请实施例提供的渲染处理方法由步骤 501 至步骤 506 所实现，其中，步骤 502 分为步骤 5021 以及步骤 5022，即在计算机设备运行虚拟场景的过程中，虚拟场景自动判断是否需要启动计算迁移，即启动计算着色器从图形处理器到中央处理器的计算迁移。若判断不要启动计算迁移，则可以按照图形处理器进行完整的虚拟场景渲染，即由图形处理器执行计算着色器相关资源运算；若判断需要启动计算迁移，则截断虚拟场景针对计算着色器相关的资源准备、管线分配和分发的调用。在截断针对计算着色器相关的调用后，将针对计算着色器相关的调用配置到中央处理器中，即是配置到图形处理器中的 warp 设备。在中央处理器调用 warp 设备的上下文进行资源运算时，对 warp 设备进行 flush 和阻塞等待，并使用配置好的中央处理器对目标资源进行运算，得到运算结果。然后，可将运算结果从中央处理器中拷贝回图形处理器中，即是拷贝回图形处理器中运行的管线上，以便于图形处理器使用该运算结果继续渲染虚拟场景。

基于图 5 所示的完整的渲染处理方法的技术流程,下面结合图 6 对图 5 所示技术流程进行更为详细的阐述。图 6 示出了本申请一个示例性实施例提供的一种渲染处理方法的流程示意图;该渲染处理方法可以由前述提及的计算机设备执行,该方法可包括步骤 S601-S605:

S601,检测图形处理器和中央处理器是否满足计算迁移启动条件。

5 正如前述所描述的,目标对象所持有的终端设备中部署有集成了渲染处理方案的本地软件,那么目标对象可以通过该本地软件拉起虚拟场景;在虚拟场景运行的过程中,虚拟场景就可以通过调用本地软件所包含的库,自动判断是否需要启动计算着色器从图形处理器到中央处理器的计算迁移。通过虚拟场景自动调用本地软件自动判断的方式,无需人工干预,不仅确保图形处理器和中央处理器均能以较高的吞吐运行虚拟场景,而且可以提升虚拟场景的
10 运行效率,提高目标对象在虚拟场景中的体验。

在实际应用中,在虚拟场景的运行过程中,可实时或轮询检测图形处理器和中央处理器是否满足计算迁移启动条件,即是检测图形处理器的负载率(或称为占用率、使用率,是指图形处理器中被使用的资源(如内存空间或处理器等)与总资源的比值)和中央处理器的负载率是否满足计算迁移条件,以便于确定是否启动或取消计算迁移。其中,图形处理器和
15 中央处理器满足计算迁移条件可包括:中央处理器的负载率大于第一负载阈值,且中央处理器的负载率小于第二负载率;第一负载阈值和第二负载阈值的具体取值可是根据经验或测试得到的,如第一负载阈值的取值可为 90%,第二负载阈值的取值可为 50%,本申请实施例对第一负载阈值和第二负载阈值的具体取值不作限定。

S602,若检测到图形处理器和中央处理器满足计算迁移启动条件,则确定图形处理器在
20 渲染虚拟场景时进行资源运算所需的目标资源。

S603,根据目标资源对中央处理器进行运算配置,并通过配置好的中央处理器,结合目标资源进行资源运算,得到运算结果。

步骤 S602-S603 中,如果在运行虚拟场景的过程中,检测到图形处理器的负载率大于第一负载阈值,且中央处理器的负载率小于第二负载阈值,表示当前图形处理器承担较大的运行压力,运算效率会有所下降,且中央处理器相对较为空闲,则可以启动计算迁移,以实现
25 通过增加中央处理器的负载降低图形处理器的运行压力。其中,启动计算迁移后的部分操作,可包括步骤 S602 所示的截断虚拟场景对计算着色器相关的调用,以及,步骤 S603 所示的目标资源从图形处理器到中央处理器的转移。需要说明的是,步骤 S602-S603 所示的实施过程,可参见前述图 3 所示的步骤 S301-S302 所示的实施过程的相关描述,在此不作赘述。

30 如前述图 3 所示实施例中步骤 S302 中描述的,根据目标资源对中央处理器进行运算配置的实施过程中包括:将目标资源从图形处理器配置到中央处理器中,即是配置到中央处理器中的 warp 设备的上下文中。针对此步骤还需补充的是,考虑到需要配置到 warp 设备的上下文中的目标资源可能是位于图形处理器中的,如无序访问视图对应的资源,或着色器资源
35 视图对应的资源均是位于中央处理器中的,而 warp 设备是在中央处理器中创建的,即 warp 设备是位于中央处理器中的;因此在计算迁移过程中,管线配置的过程(即前述提及的目标资源配置到 warp 设备的过程),会对应目标资源从图形处理器到中央处理器的拷贝的过程。也就是说,目标资源的配置过程(或称为管线配置的过程),其实质是对应目标资源从图形处理器到中央处理器的拷贝过程。

可以理解的是,中央处理器和图形处理器所能读写的资源并不完全相同,因此上述提及
40 的资源拷贝过程并非是简单的直接拷贝,而是需要一个中间数据格式作为中转实现间隔的拷贝。下面以目标资源属于第一资源类型,且图形处理器具有针对第一资源类型的目标资源的读写权限,但中央处理器不具有针对第一资源类型的目标资源的读写权限,而具有针对第二资源类型的资源的读写权限,为例对目标资源从图形处理器到中央处理器的拷贝的实施过程进行介绍,其中,第一资源类型与第二资源类型不同。

45 在实际应用中,考虑到图形处理器虽然具有对属于第一资源类型的目标资源的读写权限,即是目标资源中包括有目标数据(或称为子资源),图形处理器具有对目标资源包括的目标

数据的读写权限；但中央处理器不具有对属于第一资源类型的目标资源的读写权限，因此，属于第一资源类型的目标资源不能被中央处理器从图形处理器中读取到。基于此，需要先在图形处理器中创建能够被中央处理器读取的属于第二资源类型的第一参考资源，即中央处理器具有对属于第二资源类型的资源的读写权限；创建的属于第二资源类型的第一参考资源的大小（或称为资源容量）与目标资源的大小相同。然后，先在图形处理器中将目标资源所包含的目标数据，拷贝至第一参考资源中，得到更新后的第一参考资源，该更新后的第一参考资源包括目标数据。最后，由于第一参考资源属于第二资源类型，该第二资源类型能被中央处理器读取，因此，可实现从更新后的第一参考资源中，将目标数据拷贝至中央处理器中。

在实际实施时，虽然中央处理器具有对第二资源类型的目标资源的读取权限，但中央处理器中创建的图形变换装置所使用的接口仍然是 DirectX 的接口（API），即图形变换装置所能读写的资源的资源类型仍然是第一资源类型；基于此，还需在图形变换装置（可以是图形变换装置的上下文）中创建属于第二资源类型的第三参考资源。然后，将图形处理器中属于第二资源类型的第一参考资源中的目标数据，拷贝至图形变换装置中的第三参考资源中，得到更新后的第三参考资源，该更新后的第三参考资源中包括目标数据。最后，将更新后的第三参考资源中的目标数据，拷贝至第二参考资源，以实现将目标数据拷贝至中央处理器中的图形变换装置；其中，图形变换装置本身包含属于第一资源类型的第二参考资源。

为便于更好地理解上述给出的资源拷贝的过程，下面对上述描述的资源拷贝过程进行举例说明。如图 7 所示，以目标资源包括纹理资源为例，图形处理器使用的纹理资源的数据格式的类型为 D3D11_USAGE_DEFAULT 类型，D3D11_USAGE_DEFAULT 类型的资源能够被中央处理器读写，但不能被中央处理器读写，因此要将属于 D3D11_USAGE_DEFAULT 类型的目标资源（可以是目标资源所包含的目标数据）从图形处理器拷贝至中央处理器时，需要在图形处理器中创建一个与目标资源大小相同，且能够被中央处理器读写的第二资源类型的第一参考资源，如能够被中央处理器读写的第二资源类型为 D3D11_USAGE_STAGING 类型。然后，先将目标资源所包含的目标数据拷贝至创建的 D3D11_USAGE_STAGING 类型的第一参考资源中；再从 D3D11_USAGE_STAGING 类型的第一参考资源中，将目标资源拷贝至中央处理器中，即是拷贝至中央处理器中的 warp 设备。

同理，考虑到 warp 设备支持读写的资源类型为 D3D11_USAGE_DEFAULT 类型，即 warp 设备本身包含属于 D3D11_USAGE_DEFAULT 类型的第二参考资源，但中央处理器支持读取的资源类型为 D3D11_USAGE_STAGING 类型；因此，为了成功将目标资源包含的目标数据从图形处理器拷贝至中央处理器中的 warp 设备的上下文中，则还需 warp 设备中的上下文创建一个属于 D3D11_USAGE_STAGING 类型的第三参考资源，这样就可以将目标数据从第三参考资源中拷贝至第二参考资源，以实现 warp 设备从支持读取的 D3D11_USAGE_DEFAULT 类型的第二参考资源中读取到目标数据，从而根据目标数据执行资源运算等相关处理。

上述过程需特别说明的是，①考虑到 D3D11_USAGE_STAGING 类型的目标资源可以直接将虚拟地址（map）存储到内存中，那么就可以从图形处理器下的 D3D11_USAGE_STAGING 类型资源的虚拟地址，拷贝到中央处理器下的 D3D11_USAGE_STAGING 类型资源的虚拟地址。如图 7 所示，即是属于 D3D11_USAGE_STAGING 类型的第一参考资源包含的目标数据的虚拟地址存储至内存空间中，那么中央处理器可从内存空间中将该目标数据的虚拟地址拷贝至图形变换装置中属于 D3D11_USAGE_STAGING 类型的第三参考资源中；也就是说，可通过拷贝目标数据的虚拟地址的方式实现对目标数据的拷贝，这样根据虚拟地址即可获取到该虚拟地址指向的目标数据。

②上述拷贝过程中提到的属于第二资源类型（如 D3D11_USAGE_STAGING）的第一参考资源和第三参考资源均支持资源复用。所谓资源复用可简单理解为在一次虚拟场景的虚拟

画面渲染过程中，创建了第一参考资源和第三参考资源后，在后续渲染其他虚拟画面也需进行目标资源转移时，可直接基于已创建的第一参考资源和第三资源进行资源转移，而无需重复创建；也就是说，D3D11_USAGE_STAGING 类型的资源（如第一参考资源和第三参考资源）对每一个计算着色器用到的资源都只需要创建一次，在后续渲染过程中就可以直接复用已创建的资源；这样可避免每次资源拷贝过程中，重复创建多个资源，不仅避免了资源创建的浪费，还在一定程度上可提高资源拷贝速率，从而提升虚拟场景的渲染速度和效果。

举例来说，若在渲染虚拟场景的第一虚拟画面时，检测到需要进行计算迁移，即将关于第一虚拟画面的目标资源（如纹理资源）从图形处理器转移至中央处理器，那么需要在渲染第一虚拟画面过程中，在图形处理器中创建属于第二资源类型的第一参考资源，以及在图形变换装置中创建属于第二资源类型的第三参考资源，以实现目标资源的转移；该第一虚拟画面是虚拟场景中的任一虚拟画面。同时，若在渲染虚拟场景中的第二虚拟画面时，检测到也需要启动计算迁移，即需要将渲染第二虚拟画面时进行资源运算所需的目标资源，转移至中央处理器进行资源运算，那么考虑到在渲染第一虚拟画面时已创建计算着色器所需使用的目标资源的第一参考资源和第三参考资源，则可以直接在图形处理器中将目标资源包含的目标数据拷贝至已创建的第一参考资源，得到更新后的第一参考资源；其中，该第二虚拟画面是虚拟场景中的任一虚拟画面，且第二虚拟画面与第一虚拟画面不同。同理，可继续从更新后的第一参考资源中，将目标数据直接拷贝至中央处理器中已创建的第三参考资源。通过对第一参考资源和第三参考资源的资源复用，可避免每次虚拟画面渲染均进行相应资源的创建，在节省创建成本的情况下，还可以在在一定程度上提升迁移速度，从而提高画面渲染效率。

③中央处理器能够读取的资源类型并不仅限于上述提及的 D3D11_USAGE_DEFAULT 类型，以及图形处理器能够读取的资源类型并不仅限于上述提及的 D3D11_USAGE_STAGING 类型；上述只是以 D3D11_USAGE_DEFAULT 类型和 D3D11_USAGE_STAGING 类型为例，对资源拷贝过程进行示例性介绍，并不会对本申请实施例产生限定，特在此说明。

S604，将运算结果同步至图形处理器，运算结果，用于供图形处理器在渲染虚拟场景时使用。

需要说明的是，采用配置好的中央处理器将运算结果同步至图形处理器的过程，就是将运算结果从中央处理器拷贝回图形处理器的过程，此处的运算结果可包括无序访问视图对应的资源。与前述步骤 S603 所示的，目标资源从图形处理器拷贝至中央处理器的拷贝过程类似的，将运算结果从中央处理器拷贝回图形处理器的拷贝过程并不是简单的直接拷贝，也是需要需要一个中间数据格式作为中转实现间接的拷贝。为避免赘述，本申请实施例在此对运算结果从中央处理器到图形处理器的拷贝过程不作详细描述，实施过程可参见前述步骤 S603 所示的实施过程的相关描述。

S605，在采用中央处理器代替图形处理器进行资源运算的过程中，若检测到中央处理器满足计算迁移取消条件，则取消中央处理器进行资源运算。

在执行上述步骤 S602-S604 中的任一步骤的过程中，即在启动了计算迁移后采用中央处理器代替图形处理器进行资源运算的过程中，若检测到中央处理器满足计算迁移取消条件，则可以取消采用中央处理器进行资源运算；也就是说，停止采用中央处理器代替图形处理器执行资源运算的操作，以确保不会影响中央处理器本身的功能，即利用相对空闲状态的中央处理器来帮助图形处理器承担部分工作，但在中央处理器的负载率较高的情况下，需及时停止中央处理器执行的资源运算，以保持中央处理器正常的工作。其中，中央处理器满足计算迁移取消条件可包括：中央处理器的负载率大于第三负载阈值；与前述描述第一负载阈值和第二负载阈值类似的，第三负载阈值的具体取值可以根据经验或测试得到的，如第三负载阈值的具体取值为 90%，本申请实施例对第三负载阈值的具体取值不作限定，特在此说明。

本申请实施例中，首先通过图形处理器以及中央处理器协同处理，从而在不增加硬件成本的情况下，提升虚拟场景的运行效率，提高目标对象在虚拟场景中的体验；其次，在图形

处理器渲染虚拟场景的过程中，当图形处理器的负载率较高时，将图形处理器中的计算着色器进行资源运算时所需的目标资源拷贝至中央处理器，从而通过中央处理器代替图形处理器，执行对目标资源运算得到运算结果；再由中央处理器将运算结果同步至图形处理器中，确保数据的一致性，以便于图形处理器基于运算结果继续进行虚拟场景的渲染处理。如此，在图形处理器的负载率较高时，将目标资源转移至负载率较低的中央处理器进行处理，通过这样削峰填谷方式，在减小图形处理器的运行压力的同时，还能确保图形处理器和中央处理器均能以较高的吞吐运行虚拟场景，从而进一步提升虚拟场景的运行效率，提高目标对象在虚拟场景中的体验。另外，部分算力从图形处理器移动到中央处理器的方式，可通过增加中央处理器的开销来降低图形处理器的开销，从而在客户端虚拟场景下可降低终端设备的显卡要求，以及，在纯服务器虚拟场景下可合理化服务器的硬件配比（即中央处理器资源和图形处理器资源的配比）。

上述详细阐述了本申请实施例的渲染处理方法，为了便于更好地实施本申请实施例的上述方案，相应地，下面提供了本申请实施例的装置。

图 8 示出了本申请一个示例性实施例提供的一种渲染处理装置的结构示意图；该渲染处理装置可以是运行于计算机设备中的一个计算机程序（包括程序代码）；该渲染处理装置可以用于执行图 3 或图 6 所示的方法实施例中的部分或全部步骤。请参见图 8，该渲染处理装置包括如下单元：

获取单元 801，配置为在渲染虚拟场景的过程中，确定图形处理器在渲染虚拟场景时进行资源运算所需的目标资源；

处理单元 802，配置为根据目标资源对中央处理器进行运算配置，并通过配置好的中央处理器，结合目标资源进行资源运算，得到运算结果；

处理单元 802，还配置为将运算结果同步至图形处理器，运算结果，用于供图形处理器在渲染虚拟场景时使用。

在一种实现方式中，图形处理器中包含第一库文件，第一库文件用于指示在渲染虚拟场景时通过图形处理器进行资源运算；处理单元 802，还配置为：获取第二库文件，第二库文件用于指示在渲染虚拟场景时通过中央处理器进行资源运算；

采用第二库文件替换图形处理器中的第一库文件；

通过图形处理器调用第二库文件，其中，调用第二库文件用于确定图形处理器在渲染虚拟场景时进行资源运算所需的目标资源。

在一种实现方式中，第一库文件中包括目标函数，第一库文件通过调用目标函数，获取图形处理器在渲染虚拟场景时进行资源运算所需的目标资源；处理单元 802，还配置为：

将第一库文件对目标函数的函数调用，替换为第二库文件对目标函数的函数调用；第二库文件通过调用目标函数，获取图形处理器在渲染虚拟场景时进行资源运算所需的目标资源。

在一种实现方式中，处理单元 802，还配置为：

在中央处理器中创建图形变换装置，图形变换装置使用中央处理器进行渲染计算；

将目标资源配置到图形变换装置。

在一种实现方式中，目标资源属于第一资源类型，目标资源包括属于第一资源类型的目标数据；中央处理器不具有对属于第一资源类型的资源的读写权限，且中央处理器具有对属于第二资源类型的资源的读写权限；处理单元 802，还配置为：

在图形处理器中创建属于第二资源类型的第一参考资源，第一参考资源的资源容量与目标资源的资源容量相同；

在图形处理器中将目标数据拷贝至第一参考资源，得到更新后的第一参考资源；

从更新后的第一参考资源中，将目标数据拷贝至中央处理器中的图形变换装置。

在一种实现方式中，图形变换装置中包括属于第一资源类型的第二参考资源；处理单元 802，还配置为：

在图形变换装置中创建属于第二资源类型的第三参考资源；

将图形处理器中第一参考资源中的目标数据，拷贝至图形变换装置中的第三参考资源，得到更新后的第三参考资源；

将更新后的第三参考资源中的目标数据，拷贝至第二参考资源。

5 在一种实现方式中，处理单元 802，还配置为：

将图形处理器中第一参考资源中的目标数据的虚拟地址，拷贝至图形变换装置中的第三参考资源。

10 在一种实现方式中，若在渲染虚拟场景的第一虚拟画面时，在图形处理器中创建了属于第二资源类型的第一参考资源，以及，在图形变换装置中创建了属于第二资源类型的第三参考资源，第一虚拟画面是虚拟场景中的任一虚拟画面，则处理单元 802，还配置为：

若需要将渲染虚拟场景的第二虚拟画面时进行资源运算所需的目标资源，转移至中央处理器进行资源运算，则在图形处理器中将目标资源包括的目标数据拷贝至已创建的第一参考资源，得到更新后的第一参考资源；第二虚拟画面是虚拟场景中的任一虚拟画面，且第二虚拟画面与第一虚拟画面不同；

15 从更新后的第一参考资源中，将目标数据拷贝至中央处理器中已创建的第三参考资源。

在一种实现方式中，目标资源包括目标资源视图对应的资源，目标资源视图至少包括着色器资源视图和无序访问视图，目标资源视图还包括以下至少一个：取样视图或着色器视图；

20 其中，图形变换装置具有对着色器资源视图、取样视图和着色器视图的读取权限，不具有对着色器资源视图、取样视图和着色器视图的写入权限；图形变换装置具有对无序访问视图的读写权限，读写权限包括：读取权限和写入权限；

运算结果包括：对无序访问视图对应的资源进行运算得到的资源结果；处理单元 802，还配置为：

将资源结果从图形变换装置，拷贝回图形处理器中。

在一种实现方式中，处理单元 802，还配置为：

25 若检测到图形处理器和中央处理器满足计算迁移启动条件，则触发执行确定图形处理器在渲染虚拟场景时进行资源运算所需的目标资源的步骤；图形处理器和中央处理器满足计算迁移启动条件包括：图形处理器的负载率大于第一负载阈值，且中央处理器的负载率小于第二负载阈值；

30 在采用中央处理器代替图形处理器进行资源运算的过程中，若检测到中央处理器满足计算迁移取消条件，则取消中央处理器进行的资源运算；中央处理器满足计算迁移取消条件包括：中央处理器的负载率大于第三负载阈值。

在一种实现方式中，虚拟场景包括：虚拟游戏场景、虚拟购物场景、音视频应用场景。

35 根据本申请的一个实施例，图 8 所示的渲染处理装置中的各个单元可以分别或全部合并为一个或若干个另外的单元来构成，或者其中的某个（些）单元还可以再拆分为功能上更小的多个单元来构成，这可以实现同样的操作，而不影响本申请的实施例的技术效果的实现。上述单元是基于逻辑功能划分的，在实际应用中，一个单元的功能也可以由多个单元来实现，或者多个单元的功能由一个单元实现。在本申请的其它实施例中，该渲染处理装置也可以包括其它单元，在实际应用中，这些功能也可以由其它单元协助实现，并且可以由多个单元协作实现。根据本申请的另一个实施例，可以通过在包括中央处理单元（CPU）、随机存取存储介质（RAM）、只读存储介质（ROM）等处理元件和存储元件的例如计算机的通用计算设备上运行能够执行如图 3 或图 6 所示的相应方法所涉及的各步骤的计算机程序（包括程序代码），来构造如图 8 中所示的渲染处理装置，以及来实现本申请实施例的渲染处理方法。计算机程序可以记载于例如计算机可读记录介质上，并通过计算机可读记录介质装载于上述计算设备中，并在其中运行。

45 本申请实施例中，首先通过图形处理器以及中央处理器协同处理，从而在不增加硬件成本的情况下，提升虚拟场景的运行效率，提高目标对象在虚拟场景中的体验；其次，在图形

处理器渲染虚拟场景的过程中，当图形处理器的负载率较高时，将图形处理器中的计算着色器进行资源运算时所需的目标资源拷贝至中央处理器，从而通过中央处理器代替图形处理器，执行对目标资源运算得到运算结果；再由中央处理器将运算结果同步至图形处理器中，确保数据的一致性，以便于图形处理器基于运算结果继续进行虚拟场景的渲染处理。如此，在图形处理器的负载率较高时，将目标资源转移至负载率较低的中央处理器进行处理，通过这样削峰填谷方式，在减小图形处理器的运行压力的同时，还能确保图形处理器和中央处理器均能以较高的吞吐运行虚拟场景，从而进一步提升虚拟场景的运行效率，提高目标对象在虚拟场景中的体验。另外，部分算力从图形处理器移动到中央处理器的方式，可通过增加中央处理器的开销来降低图形处理器的开销，从而在客户端虚拟场景下可降低终端设备的显卡要求，以及，在纯服务器虚拟场景下可合理化服务器的硬件配比（即中央处理器资源和图形处理器资源的配比）。

图 9 示出了本申请一个示例性实施例提供的一种计算机设备的结构示意图。请参见图 9，该计算机设备，包括处理器 901、通信接口 902 以及计算机可读存储介质 903。其中，处理器 901、通信接口 902 以及计算机可读存储介质 903 可通过总线或者其它方式连接。其中，通信接口 902 配置为接收和发送数据。计算机可读存储介质 903 可以存储在计算机设备的存储器中，计算机可读存储介质 903 配置为存储计算机程序，计算机程序包括程序指令，处理器 901 配置为执行计算机可读存储介质 903 存储的程序指令。处理器 901（或称 CPU（Central Processing Unit，中央处理器））是计算机设备的计算核心以及控制核心，其适于实现一条或多条指令，适于加载并执行一条或多条指令从而实现相应方法流程或相应功能。

本申请实施例还提供了一种计算机可读存储介质（Memory），计算机可读存储介质是计算机设备中的记忆设备，配置为存放程序和数据。可以理解的是，此处的计算机可读存储介质既可以包括计算机设备中的内置存储介质，当然也可以包括计算机设备所支持的扩展存储介质。计算机可读存储介质提供存储空间，该存储空间存储了计算机设备的处理系统。并且，在该存储空间中还存放了适于被处理器 901 加载并执行的一条或多条的指令，这些指令可以是一个或多个的计算机程序（包括程序代码）。需要说明的是，此处的计算机可读存储介质可以是高速 RAM 存储器，也可以是非不稳定的存储器（non-volatile memory），例如至少一个磁盘存储器；可选的，还可以是至少一个位于远离前述处理器的计算机可读存储介质。

在一个实施例中，该计算机设备可以是前述实施例提到的目标应用程序；该计算机可读存储介质中存储有一条或多条指令；由处理器 901 加载并执行计算机可读存储介质中存放的一条或多条指令，以实现上述渲染处理方法实施例中的相应步骤；在实际应用中，计算机可读存储介质中的一条或多条指令由处理器 901 加载并执行如下步骤：

在渲染虚拟场景的过程中，确定图形处理器在渲染所述虚拟场景时进行资源运算所需的目标资源；

根据所述目标资源对中央处理器进行运算配置，并通过配置好的所述中央处理器，结合所述目标资源进行资源运算，得到运算结果；

将所述运算结果同步至所述图形处理器，所述运算结果，用于供所述图形处理器在渲染所述虚拟场景时使用。

在一种实现方式中，图形处理器中包含第一库文件，第一库文件用于指示在渲染虚拟场景时通过图形处理器进行资源运算；计算机可读存储介质中的一条或多条指令由处理器 901 加载并在执行，确定图形处理器在渲染虚拟场景时进行资源运算所需的目标资源时，执行如下步骤：

获取第二库文件，第二库文件用于指示在渲染虚拟场景时通过中央处理器进行资源运算；采用第二库文件替换图形处理器中的第一库文件；

通过图形处理器调用第二库文件，其中，调用第二库文件用于确定图形处理器在渲染虚拟场景时进行资源运算所需的目标资源。

在一种实现方式中，第一库文件中包括目标函数，第一库文件通过调用目标函数，获取图形处理器在渲染虚拟场景时进行资源运算所需的目标资源；计算机可读存储介质中的一条或多条指令由处理器 901 加载并在执行采用第二库文件替换图形处理器中的第一库文件时，执行如下步骤：

- 5 将第一库文件对目标函数的函数调用，替换为第二库文件对目标函数的函数调用；第二库文件通过调用目标函数，获取图形处理器在渲染虚拟场景时进行资源运算所需的目标资源。

在一种实现方式中，计算机可读存储介质中的一条或多条指令由处理器 901 加载并在执行根据目标资源对中央处理器进行运算配置时，执行如下步骤：

在中央处理器中创建图形变换装置，图形变换装置使用中央处理器进行渲染计算；

- 10 将目标资源配置到图形变换装置。

在一种实现方式中，目标资源属于第一资源类型，目标资源包括属于第一资源类型的目标数据；中央处理器不具有对属于第一资源类型的资源的读写权限，且中央处理器具有对属于第二资源类型的资源的读写权限；计算机可读存储介质中的一条或多条指令由处理器 901 加载并在执行将目标资源配置到图形变换装置时，执行如下步骤：

- 15 在图形处理器中创建属于第二资源类型的第一参考资源，第一参考资源的资源容量与目标资源的资源容量相同；

在图形处理器中将目标数据拷贝至第一参考资源，得到更新后的第一参考资源；

从更新后的第一参考资源中，将目标数据拷贝至中央处理器中的图形变换装置。

- 20 在一种实现方式中，图形变换装置中包括属于第一资源类型的第二参考资源；计算机可读存储介质中的一条或多条指令由处理器 901 加载并在执行从更新后的第一参考资源中，将目标数据拷贝至中央处理器中的图形变换装置时，执行如下步骤：

在图形变换装置中创建属于第二资源类型的第三参考资源；

将图形处理器中第一参考资源中的目标数据，拷贝至图形变换装置中的第三参考资源，得到更新后的第三参考资源；

- 25 将更新后的第三参考资源中的目标数据，拷贝至第二参考资源。

在一种实现方式中，计算机可读存储介质中的一条或多条指令由处理器 901 加载并在执行将中央处理器中第一参考资源中的目标资源，拷贝至图形变换装置中的第三参考资源，得到更新后的第三参考资源时，执行如下步骤：

- 30 将图形处理器中第一参考资源中的目标数据的虚拟地址，拷贝至图形变换装置中的第三参考资源。

在一种实现方式中，若在渲染虚拟场景的第一虚拟画面时，在图形处理器中创建了属于第二资源类型的第一参考资源，以及，在图形变换装置中创建了属于第二资源类型的第三参考资源，第一游戏画面是虚拟场景中的任一虚拟画面，则计算机可读存储介质中的一条或多条指令由处理器 901 加载并还执行如下步骤：

- 35 若需要将渲染虚拟场景的第二虚拟画面时进行资源运算所需的目标资源，转移至中央处理器进行资源运算，则在图形处理器中将目标资源包括的目标数据拷贝至已创建的第一参考资源，得到更新后的第一参考资源；第二虚拟画面是虚拟场景中的任一虚拟画面，且第二虚拟画面与第一虚拟画面不同；

从更新后的第一参考资源中，将目标数据拷贝至中央处理器中已创建的第三参考资源。

- 40 在一种实现方式中，目标资源包括目标资源视图对应的资源，目标资源视图至少包括着色器资源视图和无序访问视图，目标资源视图还包括以下至少一个：取样视图或着色器视图；

其中，图形变换装置具有对着色器资源视图、取样视图和着色器视图的读取权限，不具有对着色器资源视图、取样视图和着色器视图的写入权限；图形变换装置具有对无序访问视图的读写权限，读写权限包括：读取权限和写入权限；

- 45 运算结果包括：对无序访问视图对应的资源进行运算得到的资源结果；计算机可读存储介质中的一条或多条指令由处理器 901 加载并在执行将配置好的中央处理器中的运算结果

同步至图形处理器时，执行如下步骤：

将资源结果从图形变换装置，拷贝回图形处理器中。

在一种实现方式中，计算机可读存储介质中的一条或多条指令由处理器 901 加载并还执行如下步骤：

- 5 若检测到图形处理器和中央处理器满足计算迁移启动条件，则触发执行确定图形处理器在渲染虚拟场景时进行资源运算所需的目标资源的步骤；图形处理器和中央处理器满足计算迁移启动条件包括：图形处理器的负载率大于第一负载阈值，且中央处理器的负载率小于第二负载阈值；

10 在采用中央处理器代替图形处理器进行资源运算的过程中，若检测到中央处理器满足计算迁移取消条件，则取消中央处理器进行的资源运算；中央处理器满足计算迁移取消条件包括：中央处理器的负载率大于第三负载阈值。

在一种实现方式中，虚拟场景包括：虚拟游戏场景、虚拟购物场景、音视频应用场景。

15 本申请实施例中，首先通过图形处理器以及中央处理器协同处理，从而在不增加硬件成本的情况下，提升虚拟场景的运行效率，提高目标对象在虚拟场景中的体验；其次，在图形处理器渲染虚拟场景的过程中，当图形处理器的负载率较高时，将图形处理器中的计算着色器进行资源运算时所需的目标资源拷贝至中央处理器，从而通过中央处理器代替图形处理器，执行对目标资源运算得到运算结果；再由中央处理器将运算结果同步至图形处理器中，确保数据的一致性，以便于图形处理器基于运算结果继续进行虚拟场景的渲染处理。如此，在图形处理器的负载率较高时，将目标资源转移至负载率较低的中央处理器进行处理，通过这样
20 削峰填谷方式，在减小图形处理器的运行压力的同时，还能确保图形处理器和中央处理器均能以较高的吞吐运行虚拟场景，从而进一步提升虚拟场景的运行效率，提高目标对象在虚拟场景中的体验。另外，部分算力从图形处理器移动到中央处理器的方式，可通过增加中央处理器的开销来降低图形处理器的开销，从而在客户端虚拟场景下可降低终端设备的显卡要求，以及，在纯服务器虚拟场景下可合理化服务器的硬件配比（即中央处理器资源和图形处理器资源的配比）。

25 本申请实施例还提供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计算机可执行指令，该计算机可执行指令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计算机可执行指令，处理器执行该计算机可执行指令，使得该计算机设备执行上述渲染处理方法。

30 本领域普通技术人员可以意识到，结合本申请中所公开的实施例描述的各示例的单元及算法步骤，能够以电子硬件、或者计算机软件和电子硬件的结合来实现。这些功能究竟以硬件还是软件方式来执行，取决于技术方案的特定应用和设计约束条件。专业技术人员可以对每个特定的应用，使用不同方法来实现所描述的功能，但是这种实现不应认为超出本申请的范围。

35 在上述实施例中，可以全部或部分地通过软件、硬件、固件或者其任意组合来实现。当使用软件实现时，可以全部或部分地以计算机程序产品的形式实现。计算机程序产品包括一个或多个计算机指令。在计算机上加载和执行计算机程序指令时，全部或部分地产生按照本发明实施例的流程或功能。计算机可以是通用计算机、专用计算机、计算机网络、或者其他可编程设备。计算机指令可以存储在计算机可读存储介质中，或者通过计算机可读存储介质
40 进行传输。计算机指令可以从一个网站站点、计算机、服务器或数据中心通过有线（例如，同轴电缆、光纤、数字用户线（DSL））或无线（例如，红外、无线、微波等）方式向另一个网站站点、计算机、服务器或数据中心进行传输。计算机可读存储介质可以是计算机能够存取的任何可用介质或者是包含一个或多个可用介质集成的服务器、数据中心等数据存储设备。可用介质可以是磁性介质（例如，软盘、硬盘、磁带）、光介质（例如，DVD）、或者半导体介质
45 （例如，固态硬盘（Solid State Disk，SSD））等。

前述内容，仅为本申请的具体实施方式，但本申请的保护范围并不局限于此，任何熟悉本技术领域的技术人员在本发明揭露的技术范围内，可轻易想到变化或替换，都应涵盖在本申请的保护范围之内。因此，本申请的保护范围应以权利要求的保护范围为准。

权利要求书

1、一种渲染处理方法，所述方法由电子设备执行，包括：

在渲染虚拟场景的过程中，确定图形处理器在渲染所述虚拟场景时进行资源运算所需的目标资源；

根据所述目标资源对中央处理器进行运算配置，并通过配置好的所述中央处理器，结合所述目标资源进行资源运算，得到运算结果；

将所述运算结果同步至所述图形处理器，所述运算结果，用于供所述图形处理器在渲染所述虚拟场景时使用。

2、如权利要求 1 所述的方法，其中，所述图形处理器中包含第一库文件，所述第一库文件用于指示在渲染所述虚拟场景时通过图形处理器进行资源运算；所述确定所述图形处理器在渲染所述虚拟场景时进行资源运算所需的目标资源，包括：

获取第二库文件，所述第二库文件用于指示在渲染所述虚拟场景时通过中央处理器进行资源运算；

采用所述第二库文件替换所述图形处理器中的所述第一库文件；

通过所述图形处理器调用所述第二库文件，其中，调用所述第二库文件用于确定所述图形处理器在渲染所述虚拟场景时进行资源运算所需的目标资源。

3、如权利要求 2 所述的方法，其中，所述第一库文件中包括目标函数，所述第一库文件通过调用所述目标函数，获取所述图形处理器在渲染所述虚拟场景时进行资源运算所需的目标资源；

所述采用所述第二库文件替换所述图形处理器中的所述第一库文件，包括：

将所述第一库文件对所述目标函数的函数调用，替换为所述第二库文件对所述目标函数的函数调用；所述第二库文件通过调用所述目标函数，获取所述图形处理器在渲染所述虚拟场景时进行资源运算所需的目标资源。

4、如权利要求 1-3 所述的方法，其中，所述根据所述目标资源对中央处理器进行运算配置，包括：

在中央处理器中创建图形变换装置，所述图形变换装置使用所述中央处理器进行渲染计算；

将所述目标资源配置到所述图形变换装置。

5、如权利要求 4 所述的方法，其中，所述目标资源属于第一资源类型，所述目标资源包括属于所述第一资源类型的目标数据；所述中央处理器不具有对属于所述第一资源类型的资源的读写权限，且所述中央处理器具有对属于第二资源类型的资源的读写权限；所述将所述目标资源配置到所述图形变换装置，包括：

在所述图形处理器中创建属于所述第二资源类型的第一参考资源，所述第一参考资源的资源容量与所述目标资源的资源容量相同；

在所述图形处理器中将所述目标数据拷贝至所述第一参考资源，得到更新后的第一参考资源；

从更新后的第一参考资源中，将所述目标数据拷贝至所述中央处理器中的所述图形变换装置。

6、如权利要求 5 所述的方法，其中，所述图形变换装置中包括属于所述第一资源类型的第二参考资源；所述从更新后的第一参考资源中，将所述目标数据拷贝至所述中央处理器中的所述图形变换装置，包括：

在所述图形变换装置中创建属于所述第二资源类型的第三参考资源；

将所述图形处理器中所述第一参考资源中的目标数据，拷贝至所述图形变换装置中的所述第三参考资源，得到更新后的第三参考资源；

将所述更新后的第三参考资源中的目标数据，拷贝至所述第二参考资源。

7、如权利要求 6 所述的方法，其中，所述将所述中央处理器中所述第一参考资源中的目标数据，拷贝至所述图形变换装置中的所述第三参考资源，得到更新后的第三参考资源，包括：

将所述图形处理器中所述第一参考资源中的目标数据的虚拟地址，拷贝至所述图形变换装置中的所述第三参考资源。

8、如权利要求 5-7 任一项所述的方法，其中，若在渲染所述虚拟场景的第一虚拟画面时，在图形处理器中创建了属于第二资源类型的第一参考资源，以及，在图形变换装置中创建了属于第二资源类型的第三参考资源，所述第一虚拟画面是所述虚拟场景中的任一虚拟画面，则所述方法还包括：

若需要将渲染第二虚拟画面时进行资源运算所需的目标资源，转移至所述中央处理器进行资源运算，则在所述图形处理器中将所述目标资源包括的目标数据，拷贝至已创建的所述第一参考资源，得到更新后的第一参考资源；所述第二虚拟画面是所述虚拟场景中的任一虚拟画面，且所述第二虚拟画面与所述第一虚拟画面不同；

从更新后的第一参考资源中，将所述目标数据拷贝至所述中央处理器中已创建的所述第三参考资源。

9、如权利要求 1-8 所述的方法，其中，所述目标资源包括目标资源视图对应的资源，所述目标资源视图至少包括着色器资源视图和无序访问视图，所述目标资源视图还包括以下至少一个：取样视图或着色器视图；

其中，图形变换装置具有对所述着色器资源视图、所述取样视图和所述着色器视图的读取权限，不具有对所述着色器资源视图、所述取样视图和所述着色器视图的写入权限；图形变换装置具有对所述无序访问视图的读写权限，所述读写权限包括：读取权限和写入权限；

所述运算结果包括：对所述无序访问视图对应的资源进行运算得到的资源结果；所述将所述运算结果同步至所述图形处理器，包括：

将所述资源结果从所述图形变换装置，拷贝回所述图形处理器中。

10、如权利要求 1-9 所述的方法，其中，所述方法还包括：

若检测到图形处理器和中央处理器满足计算迁移启动条件，则触发执行所述确定所述图形处理器在渲染所述虚拟场景时进行资源运算所需的目标资源的步骤；所述图形处理器和所述中央处理器满足计算迁移启动条件包括：图形处理器的负载率大于第一负载阈值，且中央处理器的负载率小于第二负载阈值；

在采用所述中央处理器代替所述图形处理器进行资源运算的过程中，若检测到所述中央处理器满足计算迁移取消条件，则取消所述中央处理器进行的资源运算；所述中央处理器满足计算迁移取消条件包括：中央处理器的负载率大于第三负载阈值。

11、如权利要求 1-10 所述的方法，其中，所述虚拟场景包括：虚拟游戏场景、虚拟购物场景、音视频应用场景。

12、一种渲染处理装置，包括：

获取单元，配置为在渲染虚拟场景的过程中，确定图形处理器在渲染所述虚拟场景时进行资源运算所需的目标资源；

处理单元，配置为根据所述目标资源对中央处理器进行运算配置，并通过配置好的所述中央处理器，结合所述目标资源进行资源运算，得到运算结果；

所述处理单元，还配置为将所述运算结果同步至所述图形处理器，所述运算结果，用于供所述图形处理器在渲染所述虚拟场景时使用。

13、一种电子设备，包括：

存储器，配置为存储计算机可执行指令；

处理器，配置为执行所述存储器中存储的计算机可执行指令时，实现权利要求 1 至 11 任一项所述的方法。

14、一种计算机可读存储介质，存储有计算机可执行指令，所述计算机可执行指令被处

理器执行时，实现权利要求 1 至 11 任一项所述的方法。

15、一种计算机程序产品，包括计算机程序或计算机可执行指令，所述计算机程序或计算机可执行指令被处理器执行时，实现权利要求 1 至 11 任一项所述的方法。

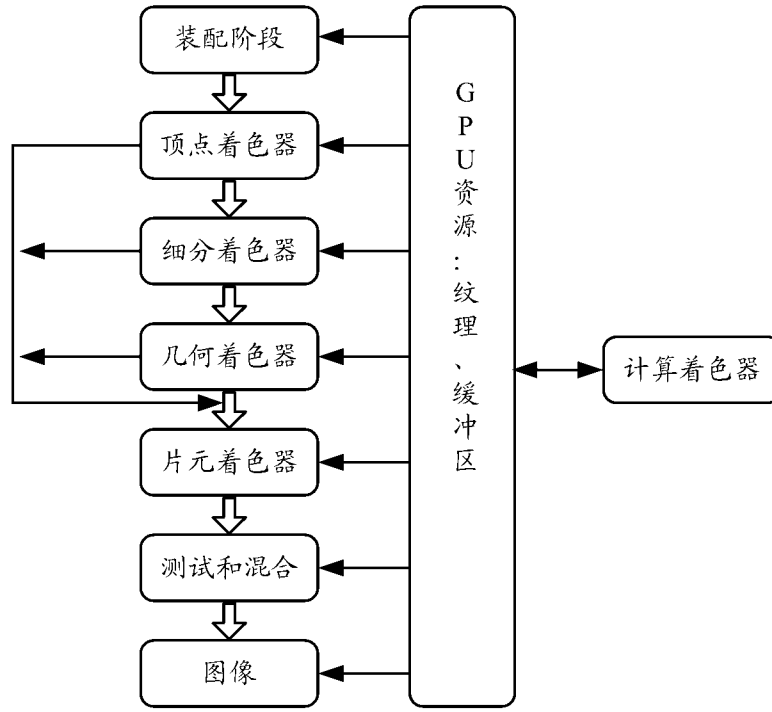


图 1

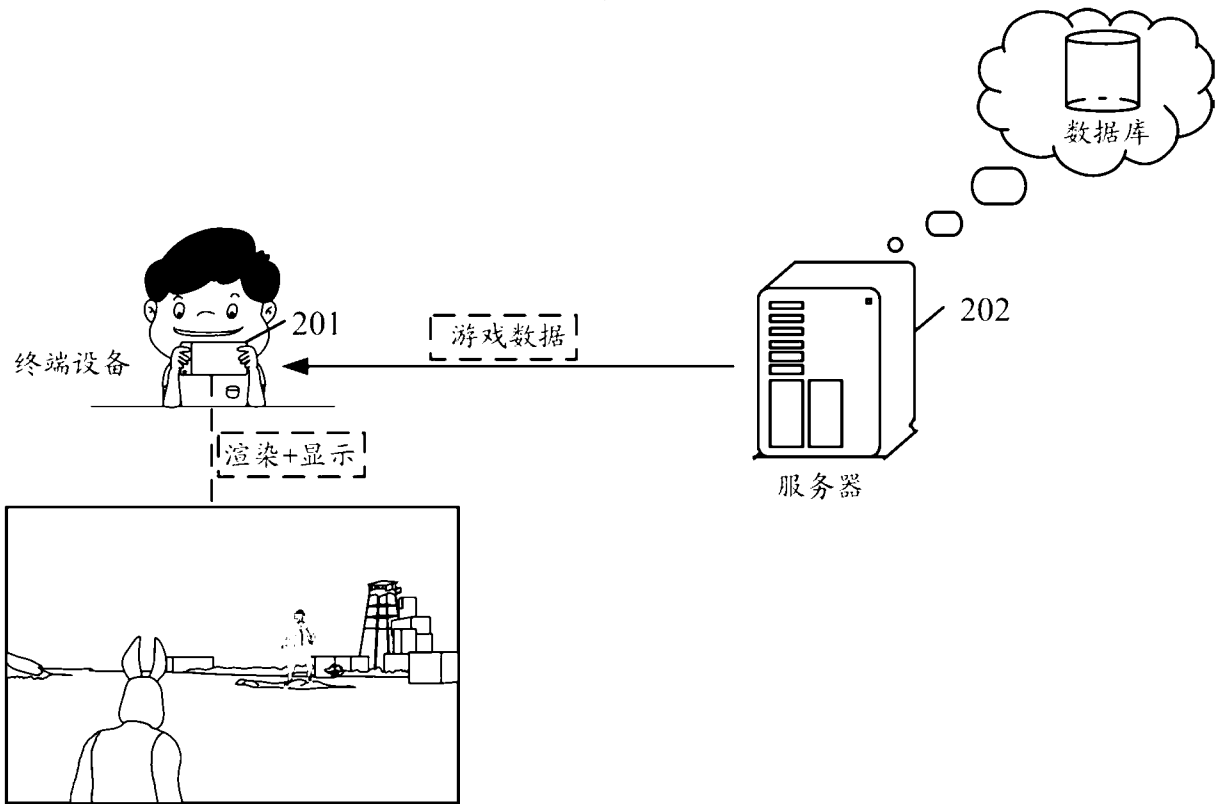


图 2A

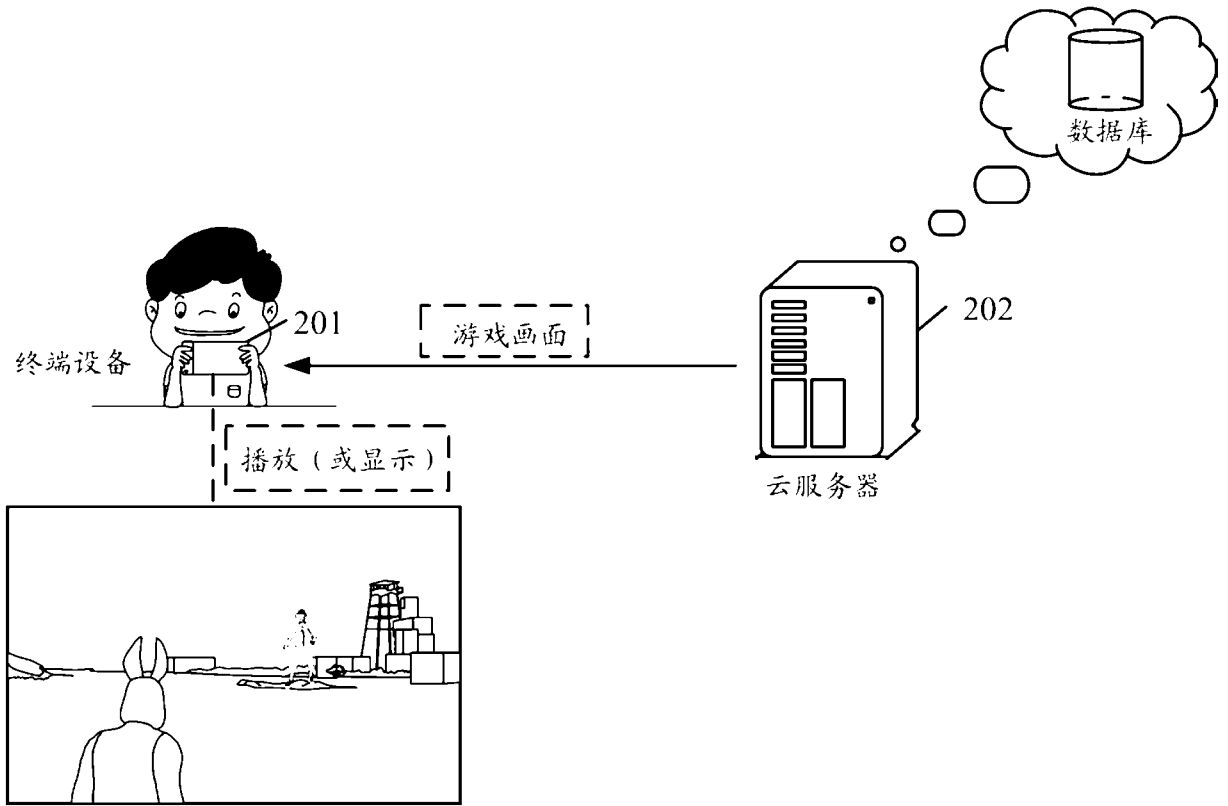


图 2B

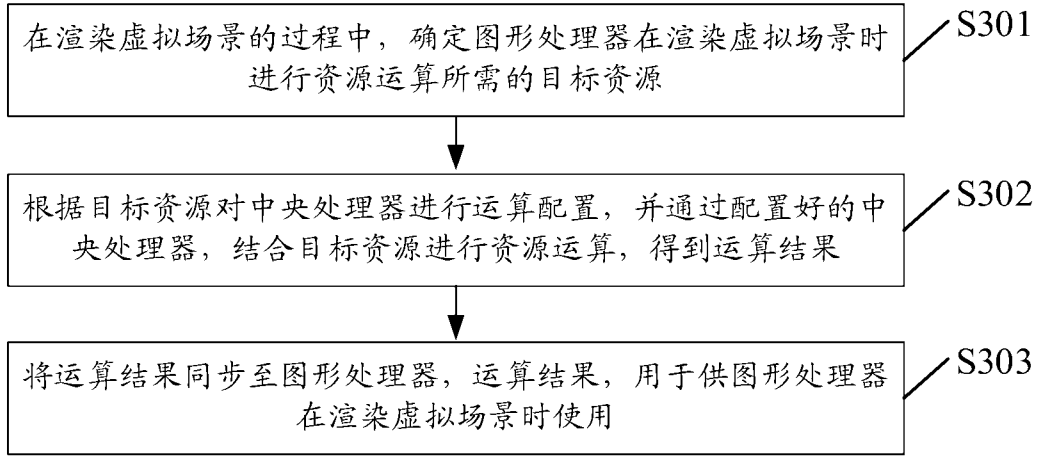


图 3

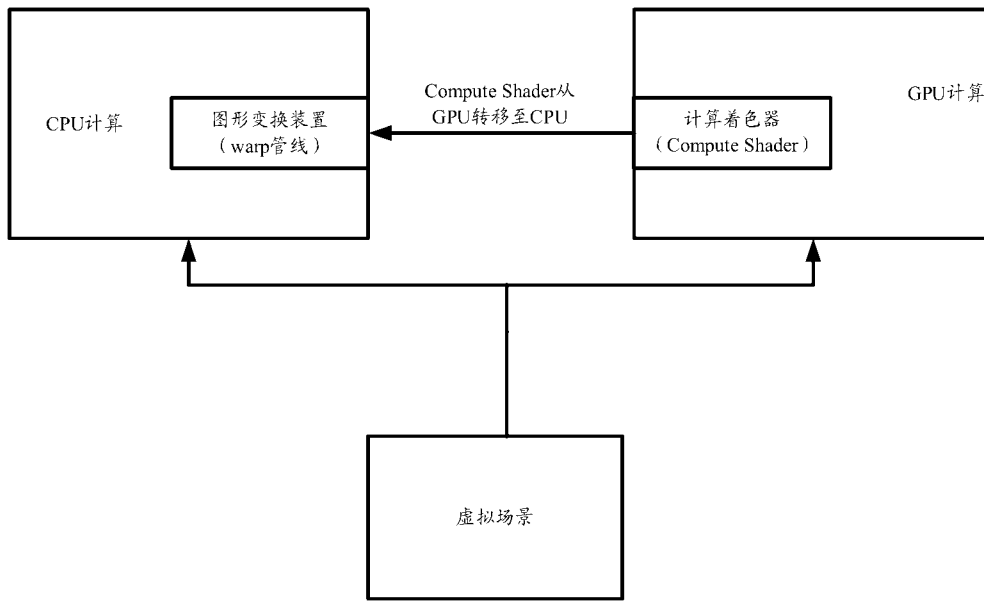


图 4

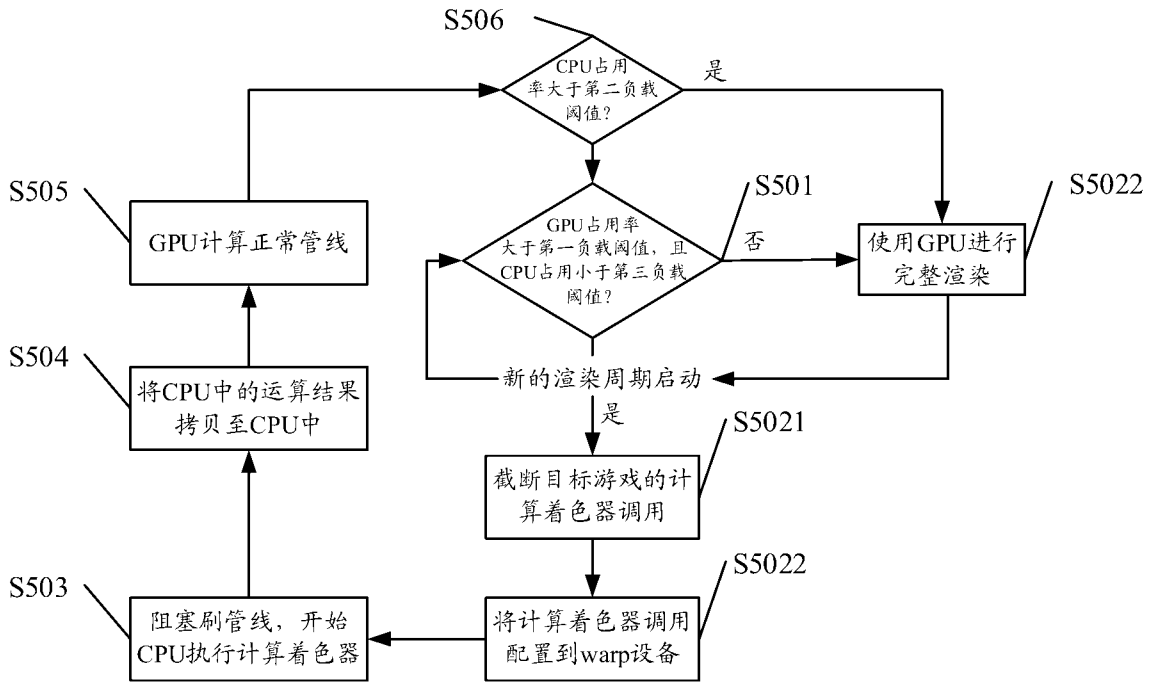


图 5

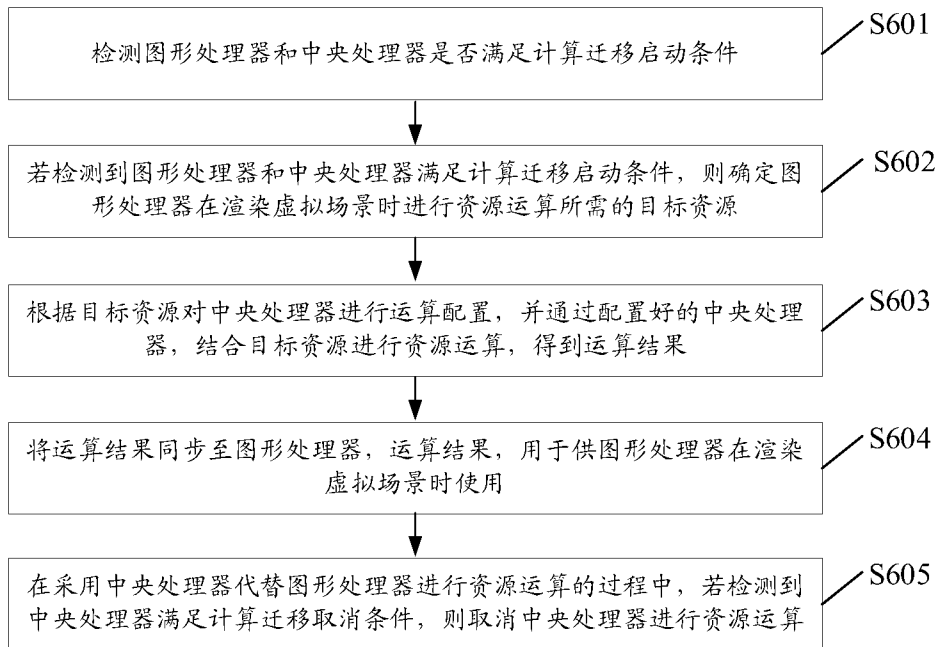
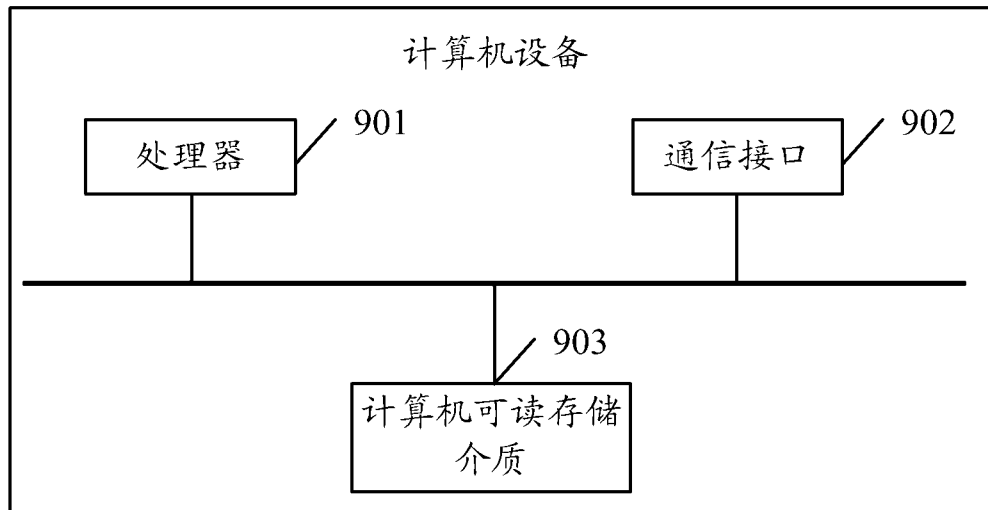
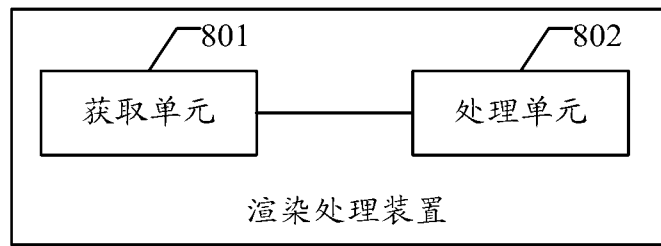
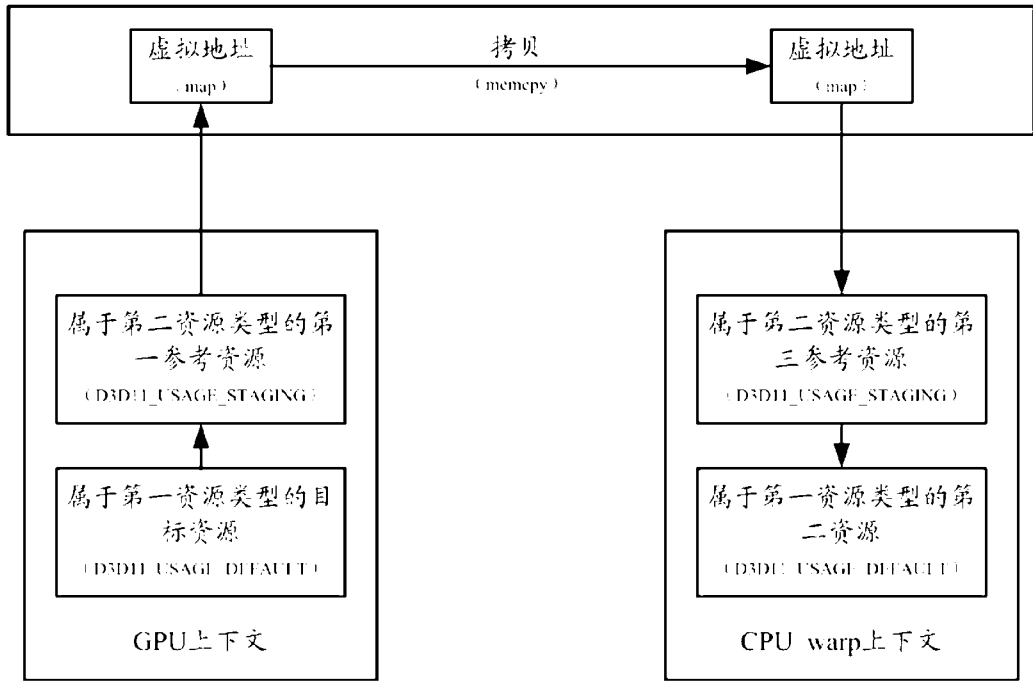


图 6



INTERNATIONAL SEARCH REPORT

International application No.

PCT/CN2023/118545

A. CLASSIFICATION OF SUBJECT MATTER		
A63F13/77(2014.01)i; G06F9/50(2006.01)i; G06T15/00(2011.01)i		
According to International Patent Classification (IPC) or to both national classification and IPC		
B. FIELDS SEARCHED		
Minimum documentation searched (classification system followed by classification symbols)		
IPC:A63F,G06F,G06T		
Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched		
Electronic data base consulted during the international search (name of data base and, where practicable, search terms used)		
VEN, CNABS, CNTXT, WOTXT, EPTXT, USTXT, CNKI, IEEE: 传输, 发送, 负担, 负载, 迁移, 转移, 图形处理器, 中央处理器, 渲染, transmit, transfer, burden, load, move, send, GPU, CPU		
C. DOCUMENTS CONSIDERED TO BE RELEVANT		
Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
PX	CN 115350479 A (TENCENT TECHNOLOGY (SHENZHEN) CO., LTD.) 18 November 2022 (2022-11-18) description, paragraphs 0043-0116	1-15
X	CN 111754381 A (HUAWEI TECHNOLOGIES CO., LTD.) 09 October 2020 (2020-10-09) description, paragraphs, 0262-0296, 0302, and 0332-0361, and figure 12	1-4, 9-15
A	CN 104157004 A (CHANGZHOU ZANYUN SOFTWARE TECHNOLOGY CO., LTD.) 19 November 2014 (2014-11-19) entire document	1-15
A	CN 106952216 A (CHENGDU AERONAUTIC POLYTECHNIC) 14 July 2017 (2017-07-14) entire document	1-15
A	WO 2022089592 A1 (HUAWEI TECHNOLOGIES CO., LTD.) 05 May 2022 (2022-05-05) entire document	1-15
<input type="checkbox"/> Further documents are listed in the continuation of Box C. <input checked="" type="checkbox"/> See patent family annex.		
* Special categories of cited documents: "A" document defining the general state of the art which is not considered to be of particular relevance "D" document cited by the applicant in the international application "E" earlier application or patent but published on or after the international filing date "L" document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified) "O" document referring to an oral disclosure, use, exhibition or other means "P" document published prior to the international filing date but later than the priority date claimed "T" later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention "X" document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone "Y" document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art "&" document member of the same patent family		
Date of the actual completion of the international search		Date of mailing of the international search report
09 November 2023		23 November 2023
Name and mailing address of the ISA/CN		Authorized officer
China National Intellectual Property Administration (ISA/CN) China No. 6, Xitucheng Road, Jimenqiao, Haidian District, Beijing 100088		Telephone No.

INTERNATIONAL SEARCH REPORT
Information on patent family members

International application No.

PCT/CN2023/118545

Patent document cited in search report			Publication date (day/month/year)	Patent family member(s)	Publication date (day/month/year)
CN	115350479	A	18 November 2022	None	
CN	111754381	A	09 October 2020	EP 3937118 A1	12 January 2022
				WO 2020192608 A1	01 October 2020
				US 2022012842 A1	13 January 2022
CN	104157004	A	19 November 2014	None	
CN	106952216	A	14 July 2017	None	
WO	2022089592	A1	05 May 2022	EP 4231242 A1	23 August 2023
				CN 114529658 A	24 May 2022

<p>A. 主题的分类</p> <p>A63F13/77(2014.01)i; G06F9/50(2006.01)i; G06T15/00(2011.01)i</p> <p>按照国际专利分类(IPC)或者同时按照国家分类和IPC两种分类</p>																				
<p>B. 检索领域</p> <p>检索的最低限度文献(标明分类系统和分类号)</p> <p>IPC:A63F,G06F,G06T</p> <p>包含在检索领域中的除最低限度文献以外的检索文献</p> <p>在国际检索时查阅的电子数据库(数据库的名称, 和使用的检索词(如使用))</p> <p>VEN, CNABS, CNTXT, WOTXT, EPTXT, USTXT, CNKI, IEEE: 传输, 发送, 负担, 负载, 迁移, 转移, 图形处理器, 中央处理器, 渲染, transmit, transfer, burden, load, move, send, GPU, CPU</p>																				
<p>C. 相关文件</p> <table border="1"> <thead> <tr> <th>类型*</th> <th>引用文件, 必要时, 指明相关段落</th> <th>相关的权利要求</th> </tr> </thead> <tbody> <tr> <td>PX</td> <td>CN 115350479 A (腾讯科技(深圳)有限公司) 2022年11月18日 (2022 - 11 - 18) 说明书0043-0116段</td> <td>1-15</td> </tr> <tr> <td>X</td> <td>CN 111754381 A (华为技术有限公司) 2020年10月9日 (2020 - 10 - 09) 说明书0262-0296、0302、0332-0361段, 图12</td> <td>1-4、9-15</td> </tr> <tr> <td>A</td> <td>CN 104157004 A (常州赞云软件科技有限公司) 2014年11月19日 (2014 - 11 - 19) 全文</td> <td>1-15</td> </tr> <tr> <td>A</td> <td>CN 106952216 A (成都航空职业技术学院) 2017年7月14日 (2017 - 07 - 14) 全文</td> <td>1-15</td> </tr> <tr> <td>A</td> <td>WO 2022089592 A1 (HUAWEI TECHNOLOGIES CO., LTD.) 2022年5月5日 (2022 - 05 - 05) 全文</td> <td>1-15</td> </tr> </tbody> </table> <p><input type="checkbox"/> 其余文件在C栏的续页中列出。 <input checked="" type="checkbox"/> 见同族专利附件。</p> <p>* 引用文件的具体类型: “A” 认为不特别相关的表示了现有技术一般状态的文件 “D” 申请人在国际申请中引证的文件 “E” 在国际申请日的当天或之后公布的在先申请或专利 “L” 可能对优先权要求构成怀疑的文件, 或为确定另一篇引用文件的公布日而引用的或者因其他特殊理由而引用的文件(如具体说明的) “O” 涉及口头公开、使用、展览或其他方式公开的文件 “P” 公布日先于国际申请日但迟于所要求的优先权日的文件 “T” 在申请日或优先权日之后公布, 与申请不相抵触, 但为了理解发明之理论或原理的在后文件 “X” 特别相关的文件, 单独考虑该文件, 认定要求保护的发明不是新颖的或不具有创造性 “Y” 特别相关的文件, 当该文件与另一篇或者多篇该类文件结合并且这种结合对于本领域技术人员为显而易见时, 要求保护的发明不具有创造性 “&” 同族专利的文件</p>			类型*	引用文件, 必要时, 指明相关段落	相关的权利要求	PX	CN 115350479 A (腾讯科技(深圳)有限公司) 2022年11月18日 (2022 - 11 - 18) 说明书0043-0116段	1-15	X	CN 111754381 A (华为技术有限公司) 2020年10月9日 (2020 - 10 - 09) 说明书0262-0296、0302、0332-0361段, 图12	1-4、9-15	A	CN 104157004 A (常州赞云软件科技有限公司) 2014年11月19日 (2014 - 11 - 19) 全文	1-15	A	CN 106952216 A (成都航空职业技术学院) 2017年7月14日 (2017 - 07 - 14) 全文	1-15	A	WO 2022089592 A1 (HUAWEI TECHNOLOGIES CO., LTD.) 2022年5月5日 (2022 - 05 - 05) 全文	1-15
类型*	引用文件, 必要时, 指明相关段落	相关的权利要求																		
PX	CN 115350479 A (腾讯科技(深圳)有限公司) 2022年11月18日 (2022 - 11 - 18) 说明书0043-0116段	1-15																		
X	CN 111754381 A (华为技术有限公司) 2020年10月9日 (2020 - 10 - 09) 说明书0262-0296、0302、0332-0361段, 图12	1-4、9-15																		
A	CN 104157004 A (常州赞云软件科技有限公司) 2014年11月19日 (2014 - 11 - 19) 全文	1-15																		
A	CN 106952216 A (成都航空职业技术学院) 2017年7月14日 (2017 - 07 - 14) 全文	1-15																		
A	WO 2022089592 A1 (HUAWEI TECHNOLOGIES CO., LTD.) 2022年5月5日 (2022 - 05 - 05) 全文	1-15																		
国际检索实际完成的日期	2023年11月9日	国际检索报告邮寄日期	2023年11月23日																	
ISA/CN的名称和邮寄地址	中国国家知识产权局 中国北京市海淀区蓟门桥西土城路6号 100088	授权官员	宿渊源 电话号码 (+86) 010-53961376																	

国际检索报告
关于同族专利的信息

国际申请号

PCT/CN2023/118545

检索报告引用的专利文件			公布日 (年/月/日)	同族专利			公布日 (年/月/日)
CN	115350479	A	2022年11月18日	无			
CN	111754381	A	2020年10月9日	EP	3937118	A1	2022年1月12日
				WO	2020192608	A1	2020年10月1日
				US	2022012842	A1	2022年1月13日
CN	104157004	A	2014年11月19日	无			
CN	106952216	A	2017年7月14日	无			
WO	2022089592	A1	2022年5月5日	EP	4231242	A1	2023年8月23日
				CN	114529658	A	2022年5月24日