



(12) 发明专利

(10) 授权公告号 CN 113034629 B

(45) 授权公告日 2021.08.20

(21) 申请号 202110536966.7

G06F 9/50 (2006.01)

(22) 申请日 2021.05.17

(56) 对比文件

(65) 同一申请的已公布的文献号

CN 108711182 A, 2018.10.26

申请公布号 CN 113034629 A

审查员 陈伊娜

(43) 申请公布日 2021.06.25

(73) 专利权人 腾讯科技(深圳)有限公司

地址 518057 广东省深圳市南山区高新区

科技中一路腾讯大厦35层

(72) 发明人 袁志强 赵新达 杨衍东

(74) 专利代理机构 广州三环专利商标代理有限

公司 44202

代理人 熊永强 杜维

(51) Int. Cl.

G06T 9/00 (2006.01)

A63F 13/52 (2014.01)

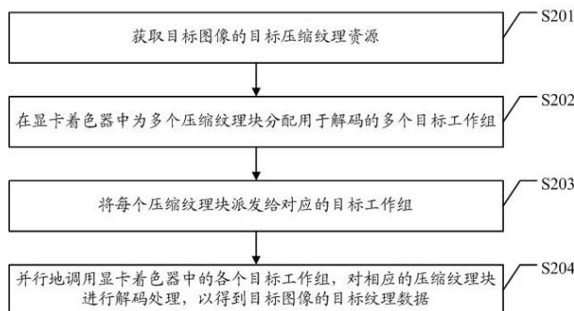
权利要求书3页 说明书20页 附图7页

(54) 发明名称

图像处理方法和装置、计算机设备及存储介质

(57) 摘要

本申请实施例公开了一种图像处理方法和装置、计算机设备及存储介质,其中方法包括:获取目标图像的目标压缩纹理资源,所述目标压缩纹理资源包括多个压缩纹理块;在显卡着色器中为所述多个压缩纹理块分配用于解码的多个目标工作组,一个目标工作组被分配一个或多个压缩纹理块;以及,将每个压缩纹理块派发给对应的目标工作组;并行地调用所述显卡着色器中的各个目标工作组,对相应的压缩纹理块进行解码处理,以得到所述目标图像的目标纹理数据,所述目标纹理数据包括所述每个压缩纹理块对应的解码数据。本申请实施例在实现解码目标压缩纹理资源的同时,可有效提升解码效率。



1. 一种图像处理方法,其特征在于,包括:

获取目标图像的目标压缩纹理资源,所述目标压缩纹理资源是采用压缩纹理格式对所述目标图像进行编码得到的,所述目标压缩纹理资源包括多个压缩纹理块;

在显卡着色器中为所述多个压缩纹理块分配用于解码的多个目标工作组,一个目标工作组被分配一个或多个压缩纹理块;以及,将每个压缩纹理块派发给对应的目标工作组;工作组是指所述显卡着色器中具有解码处理能力的子处理单元,各个工作组之间相互独立;且任一目标工作组是采用用于解码所述目标压缩纹理资源的算法指令,对相应的压缩纹理块进行解码处理的;

并行地调用所述显卡着色器中的各个目标工作组,对相应的压缩纹理块进行解码处理,以得到所述目标图像的目标纹理数据,所述目标纹理数据包括所述每个压缩纹理块对应的解码数据;

其中,所述目标图像为目标游戏中的目标游戏画面图像,所述目标游戏的离线解码数据集中包括:所述目标游戏中的一帧或多帧游戏画面图像的图像标识,以及每个图像标识所关联的纹理数据;任一图像标识所关联的纹理数据,是对所述任一图像标识所指示的游戏画面图像的压缩纹理资源进行离线解码得到的;

所述在显卡着色器中为所述多个压缩纹理块分配用于解码的多个目标工作组这一步骤,是在所述离线解码数据集中不存在所述目标图像的图像标识的情况下执行的;当所述离线解码数据集中存在所述目标图像的图像标识时,所述目标图像的目标纹理数据从所述离线解码数据集中加载得到。

2. 如权利要求1所述的方法,其特征在于,所述显卡着色器设置于计算机设备中,所述计算机设备还包括渲染管线和解码管线,所述显卡着色器按照所述解码管线规定的流程进行工作;

所述方法还包括:所述渲染管线根据所述目标纹理数据进行图像渲染,得到所述目标图像。

3. 如权利要求2所述的方法,其特征在于,所述渲染管线具有虚拟的第一显卡地址空间,所述解码管线具有虚拟的第二显卡地址空间;

其中,所述第一显卡地址空间和所述第二显卡地址空间,与同一个纹理存储空间相映射。

4. 如权利要求3所述的方法,其特征在于,所述方法还包括:

所述显卡着色器中的任一目标工作组对相应的压缩纹理块进行解码处理后,根据所述第二显卡地址空间和所述纹理存储空间之间的映射关系,将所述任一目标工作组解码得到的解码数据缓存至所述纹理存储空间中;

所述渲染管线根据所述目标纹理数据进行图像渲染,得到所述目标图像,包括:

在各个压缩纹理块对应的解码数据均被存储至所述纹理存储空间后,所述渲染管线根据所述第一显卡地址空间和所述纹理存储空间之间的映射关系,在所述纹理存储空间中根据所述目标纹理数据进行图像渲染,得到所述目标图像。

5. 如权利要求3所述的方法,其特征在于,所述方法还包括:

所述渲染管线在所述计算机设备的本地空间中,为所述目标图像分配所述纹理存储空间;

所述渲染管线将所述第一显卡地址空间映射到所述纹理存储空间,并将所述纹理存储空间的空间标识传递给所述解码管线;

所述解码管线根据所述空间标识,将所述第二显卡地址空间映射到所述纹理存储空间。

6.如权利要求2所述的方法,其特征在于,所述获取目标图像的目标压缩纹理资源包括:所述渲染管线从所述计算机设备的本地空间中,加载目标图像的目标压缩纹理资源;

所述方法还包括:

所述渲染管线将所述目标压缩纹理资源派发给所述解码管线;

所述解码管线在成功接收到所述目标压缩纹理资源后,向所述显卡着色器发送解码通知,以通知所述显卡着色器执行在显卡着色器中为所述多个压缩纹理块分配用于解码的多个目标工作组的步骤。

7.如权利要求6所述的方法,其特征在于,所述计算机设备具有显卡解码模式,所述显卡解码模式是指通过所述显卡着色器进行解码处理的模式;所述方法还包括:

所述渲染管线检测所述显卡解码模式所处的模式状态,所述模式状态包括模式开启状态或模式关闭状态;

若所述模式状态为所述模式开启状态,则所述渲染管线执行将所述目标压缩纹理资源派发给所述解码管线的步骤;

若所述模式状态为所述模式关闭状态,则所述渲染管线调用显卡驱动中央处理器对所述目标压缩纹理资源中的各个压缩纹理块进行解码处理,以得到所述目标图像的目标纹理数据。

8.如权利要求7所述的方法,其特征在于,所述目标图像为目标游戏中的目标游戏画面图像;所述方法还包括:

所述渲染管线获取所述目标游戏的离线解码数据集;

所述渲染管线采用所述目标图像的图像标识对所述离线解码数据集中的图像标识进行标识命中处理;

若未成功命中,则所述渲染管线执行检测所述显卡解码模式所处的模式状态的步骤;

若成功命中,则所述渲染管线从所述离线解码数据集中,加载被命中的图像标识所关联的纹理数据,并将加载到的纹理数据作为所述目标图像的目标纹理数据。

9.如权利要求8所述的方法,其特征在于,所述方法还包括:

所述渲染管线检测所述显卡驱动中央处理器关于所述目标压缩纹理资源的解码能力;

若检测到所述显卡驱动中央处理器具有所述解码能力,则所述渲染管线执行获取所述目标游戏的离线解码数据集的步骤;

若检测到所述显卡驱动中央处理器不具备所述解码能力,则所述渲染管线调用所述目标游戏的引擎中央处理器,对所述目标压缩纹理资源中的各个压缩纹理块进行解码处理,以得到所述目标图像的目标纹理数据。

10.如权利要求1所述的方法,其特征在于,所述方法还包括:

若所述目标压缩纹理资源中的全部或部分压缩纹理块被所述显卡着色器解码失败,则调用显卡驱动中央处理器对所述目标压缩纹理资源中的各个压缩纹理块进行解码处理,以得到所述目标图像的目标纹理数据。

11. 如权利要求1所述的方法,其特征在于,所述显卡着色器中包括M个工作组,M为大于1的整数;所述在显卡着色器中为所述多个压缩纹理块分配用于解码的多个目标工作组,包括:

遍历所述多个压缩纹理块,并确定当前遍历的当前压缩纹理块;

若所述显卡着色器中的M个工作组中存在未被分配的一个或多个剩余工作组,则在所述一个或多个剩余工作组中为所述当前压缩纹理块分配一个用于解码的目标工作组;

若所述M个工作组中的各个工作组均已被分配一个或多个压缩纹理块,则在所述M个工作组中为所述当前压缩纹理块分配一个用于解码的目标工作组。

12. 如权利要求1所述的方法,其特征在于,所述显卡着色器所执行的解码流程被独立封装在一个软件工具包中,以使在需解码任一图像的压缩纹理资源时,通过调用所述软件工具包执行解码处理。

13. 一种图像处理装置,其特征在于,包括:

获取单元,用于获取目标图像的目标压缩纹理资源,所述目标压缩纹理资源是采用压缩纹理格式对所述目标图像进行编码得到的,所述目标压缩纹理资源包括多个压缩纹理块;

处理单元,用于在显卡着色器中为所述多个压缩纹理块分配用于解码的多个目标工作组,一个目标工作组被分配一个或多个压缩纹理块;以及,将每个压缩纹理块派发给对应的目标工作组;工作组是指所述显卡着色器中具有解码处理能力的子处理单元,各个工作组之间相互独立;且任一目标工作组是采用用于解码所述目标压缩纹理资源的算法指令,对相应的压缩纹理块进行解码处理的;

所述处理单元,还用于并行地调用所述显卡着色器中的各个目标工作组,对相应的压缩纹理块进行解码处理,以得到所述目标图像的目标纹理数据,所述目标纹理数据包括所述每个压缩纹理块对应的解码数据;

其中,所述目标图像为目标游戏中的目标游戏画面图像,所述目标游戏的离线解码数据集中包括:所述目标游戏中的一帧或多帧游戏画面图像的图像标识,以及每个图像标识所关联的纹理数据;任一图像标识所关联的纹理数据,是对所述任一图像标识所指示的游戏画面图像的压缩纹理资源进行离线解码得到的;

所述在显卡着色器中为所述多个压缩纹理块分配用于解码的多个目标工作组这一步骤,是在所述离线解码数据集中不存在所述目标图像的图像标识的情况下执行的;当所述离线解码数据集中存在所述目标图像的图像标识时,所述目标图像的目标纹理数据从所述离线解码数据集中加载得到。

14. 一种计算机设备,包括输入接口和输出接口,其特征在于,还包括:显卡以及计算机存储介质,所述显卡至少包括显卡着色器;

所述计算机存储介质存储有一条或多条指令,且所述一条或多条指令适于由所述显卡加载并执行如权利要求1-12任一项所述的图像处理方法。

15. 一种计算机存储介质,所述计算机存储介质存储有一条或多条指令,所述一条或多条指令适于显卡加载并执行如权利要求1-12任一项所述的图像处理方法。

图像处理方法、装置、计算机设备及存储介质

技术领域

[0001] 本申请涉及互联网技术领域,具体涉及图像处理技术领域,尤其涉及一种图像处理方法、一种图像处理装置、一种计算机设备及一种计算机存储介质。

背景技术

[0002] 目前,图像制作者在制作图像的过程中,为了节省图像的占用内存以及传输图像所需的网络资源,通常会采用压缩纹理格式对图像进行压缩编码,得到该图像的压缩纹理资源,从而向外界发布该压缩纹理资源。那么相应的,外界的任一设备在获取到该图像的压缩纹理资源后,通常需对该压缩纹理资源进行解码处理;基于此,如何解码压缩纹理资源成为了研究热点。

发明内容

[0003] 本申请实施例提供了一种图像处理方法、装置、计算机设备及存储介质,可以在实现解码目标压缩纹理资源的同时,有效提升解码效率。

[0004] 一方面,本申请实施例提供了一种图像处理方法,所述方法包括:

[0005] 获取目标图像的目标压缩纹理资源,所述目标压缩纹理资源是采用压缩纹理格式对所述目标图像进行编码得到的,所述目标压缩纹理资源包括多个压缩纹理块;

[0006] 在显卡着色器中为所述多个压缩纹理块分配用于解码的多个目标工作组,一个目标工作组被分配一个或多个压缩纹理块;以及,将每个压缩纹理块派发给对应的目标工作组;

[0007] 并行地调用所述显卡着色器中的各个目标工作组,对相应的压缩纹理块进行解码处理,以得到所述目标图像的目标纹理数据,所述目标纹理数据包括所述每个压缩纹理块对应的解码数据。

[0008] 另一方面,本申请实施例提供了一种图像处理装置,所述装置包括:

[0009] 获取单元,用于获取目标图像的目标压缩纹理资源,所述目标压缩纹理资源是采用压缩纹理格式对所述目标图像进行编码得到的,所述目标压缩纹理资源包括多个压缩纹理块;

[0010] 处理单元,用于在显卡着色器中为所述多个压缩纹理块分配用于解码的多个目标工作组,一个目标工作组被分配一个或多个压缩纹理块;以及,将每个压缩纹理块派发给对应的目标工作组;

[0011] 所述处理单元,还用于并行地调用所述显卡着色器中的各个目标工作组,对相应的压缩纹理块进行解码处理,以得到所述目标图像的目标纹理数据,所述目标纹理数据包括所述每个压缩纹理块对应的解码数据。

[0012] 再一方面,本申请实施例提供了一种计算机设备,所述计算机设备包括输入接口和输出接口,所述计算机设备还包括:显卡着色器以及计算机存储介质;

[0013] 所述计算机存储介质存储有一条或多条指令,所述一条或多条指令适于执行如下

步骤：

[0014] 获取目标图像的目标压缩纹理资源，所述目标压缩纹理资源是采用压缩纹理格式对所述目标图像进行编码得到的，所述目标压缩纹理资源包括多个压缩纹理块；

[0015] 在显卡着色器中为所述多个压缩纹理块分配用于解码的多个目标工作组，一个目标工作组被分配一个或多个压缩纹理块；以及，将每个压缩纹理块派发给对应的目标工作组；

[0016] 并行地调用所述显卡着色器中的各个目标工作组，对相应的压缩纹理块进行解码处理，以得到所述目标图像的目标纹理数据，所述目标纹理数据包括所述每个压缩纹理块对应的解码数据。

[0017] 再一方面，本申请实施例提供了一种计算机存储介质，所述计算机存储介质存储有一条或多条指令，所述一条或多条指令适于执行如下步骤：

[0018] 获取目标图像的目标压缩纹理资源，所述目标压缩纹理资源是采用压缩纹理格式对所述目标图像进行编码得到的，所述目标压缩纹理资源包括多个压缩纹理块；

[0019] 在显卡着色器中为所述多个压缩纹理块分配用于解码的多个目标工作组，一个目标工作组被分配一个或多个压缩纹理块；以及，将每个压缩纹理块派发给对应的目标工作组；

[0020] 并行地调用所述显卡着色器中的各个目标工作组，对相应的压缩纹理块进行解码处理，以得到所述目标图像的目标纹理数据，所述目标纹理数据包括所述每个压缩纹理块对应的解码数据。

[0021] 本申请实施例通过引入显卡着色器，并在显卡着色器中预先部署多个相互独立的工作组；使得在获取到目标图像的目标压缩纹理资源后，可在显卡着色器中为目标压缩纹理资源中的每个压缩纹理块分别分配一个用于解码的目标工作组，并将每个压缩纹理块派发给对应的目标工作组。然后，可通过并行地调用显卡着色器中的各个目标工作组的方式，对相应的压缩纹理块进行解码处理；这样的处理方式，不仅可实现目标压缩纹理资源的解码处理，还可实现每个压缩纹理块的并行解码，从而有效提升解码效率。另外，通过显卡着色器实现整个解码过程，使得整个解码过程无需中央处理器的参与，这样可有效节省中央处理器的资源，从而有效减少中央处理器的资源占用率，提升计算机设备的运行效率。

附图说明

[0022] 为了更清楚地说明本申请实施例技术方案，下面将对实施例描述中所需要使用的附图作简单地介绍，显而易见地，下面描述中的附图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。

[0023] 图1a是本申请实施例提供的一种图像处理方案的解码流程示意图；

[0024] 图1b是本申请实施例提供的一种显卡着色器的解码流程示意图；

[0025] 图2是本申请实施例提供的一种图像处理方法的流程示意图；

[0026] 图3a是本申请实施例提供的一种区块链的结构示意图；

[0027] 图3b是本申请实施例提供的一种将目标纹理数据存储至区块链的示意图；

[0028] 图4是本申请另一实施例提供的一种图像处理方法的流程示意图；

[0029] 图5是本申请实施例提供的一种空间映射的示意图；

- [0030] 图6a是本申请实施例提供的一种云游戏服务器的结构示意图；
- [0031] 图6b是本申请另一实施例提供的一种图像处理方法的流程示意图；
- [0032] 图7是本申请实施例提供的一种软件工具包的调用示意图；
- [0033] 图8是本申请实施例提供的一种图像处理装置的结构示意图；
- [0034] 图9是本申请实施例提供的一种计算机设备的结构示意图。

具体实施方式

[0035] 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。

[0036] 随着互联网技术的不断发展,人工智能(Artificial Intelligence, AI)技术也随之得到更好的发展。所谓的人工智能技术是指利用数字计算机或者数字计算机控制的机器模拟、延伸和扩展人的智能,感知环境、获取知识并使用知识获得最佳结果的理论、方法、技术及应用系统。换句话说,人工智能是计算机科学的一个综合技术;其主要通过了解智能的实质,生产出一种新的能以人类智能相似的方式做出反应的智能机器,使得智能机器具有感知、推理与决策等多种功能。相应的, AI技术是一门综合学科,其主要包括计算机视觉技术(Computer Vision, CV)、语音处理技术、自然语言处理技术以及机器学习(Machine Learning, ML)/深度学习等几大方向。

[0037] 其中,计算机视觉技术是一门研究如何使机器“看”的科学,更进一步的说,就是指用摄影机和电脑代替人眼对目标进行识别、跟踪和测量等机器视觉,并进一步做图形处理,使电脑处理成为更适合人眼观察或传送给仪器检测的图像。作为一个科学学科,计算机视觉研究相关的理论和技术,试图建立能够从图像或者多维数据中获取信息的人工智能系统。计算机视觉技术通常包括图像处理、图像识别、图像语义理解、图像检索、OCR(Optical Character Recognition, 光学字符识别)、视频处理、视频语义理解、视频内容/行为识别、三维物体重建、3D(三维)技术、虚拟现实、增强现实、同步定位与地图构建等技术,还包括常见的人脸识别、指纹识别等生物特征识别技术。

[0038] 基于上述所提及的计算机视觉技术中的图像处理技术,本申请实施例提出了一种图像处理方案,以实现任一图像的压缩纹理资源进行解码处理;所谓的压缩纹理资源是指:采用压缩纹理格式对图像进行编码得到的编码结果。其中,压缩纹理格式可以是ASTC(Adaptive Scalable Texture Compression)格式、或者ETC格式(一种压缩纹理格式),等等;ASTC和ETC通常是以压缩纹理块(简称块(block))为单位进行像素压缩存储的,不同的block压缩的像素个数的范围可从 4×4 到 12×12 ,且block之间相互独立。也就是说,采用压缩纹理格式对图像进行编码所得到的压缩纹理资源通常可包括多个压缩纹理块;需说明的是,本申请实施例所提及的“多个”的含义均是指至少两个。

[0039] 在具体实现中,该图像处理方案可在一个计算机设备中执行,该计算机设备可以是终端或者服务器;此处所提及的终端可以包括但不限于:智能手机、平板电脑、笔记本电脑、台式计算机、智能手表、智能电视等;终端内可运行各式各样的客户端(application, APP),如游戏客户端、多媒体客户端、社交客户端,等等。此处所提及的服务器可以是独立的物理服务器,也可以是多个物理服务器构成的服务器集群或者分布式系统,还可以是提供云服务、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、CDN(Content Delivery Network,内容分发网络)、以及大数据和人工智能平台等基础云计算服务的云服务器,等等。

[0040] 另外,该计算机设备可以位于区块链网络外,也可以位于区块链网络内,对此不作限定;所谓的区块链网络是一种由点对点网络(P2P网络)和区块链所构成的网络,而区块链则是指一种分布式数据存储、点对点传输、共识机制、加密算法等计算机技术的新型应用模式,其本质上是一个去中心化的数据库,是一串使用密码学方法相关联产生的数据块(或称为区块)。当计算机设备位于区块链网络内,或者与区块链网络存在通信连接时,计算机设备可将内部数据上传至区块链网络的区块链中进行存储,以防止计算机设备的内部数据被篡改,从而提升内部数据的安全性。

[0041] 具体的,该计算机设备中可至少包括:图形处理器(Graphics Processing Unit, GPU)、显卡计算着色器(后续简称为显卡着色器)以及计算着色器解码管线(后续简称解码管线),等等。其中:①显卡是指以GPU为中心,且用于承担输出显示图像的任务的器件。②显卡着色器是指在显卡中运行的处理单元,其可与用于数据输入的输入显卡着色器缓存块以及用于数据输出的输出显卡着色器缓存块相连接;显卡着色器中可包括M个具有解码处理能力的工作组(work group),所谓的工作组可理解成是显卡着色器中具体解码处理能力的子处理单元;各个工作组之间相互独立,M为大于1的整数。③解码管线是一种用于指示显卡着色器的解码流程的处理单元,即本申请实施例中的显卡着色器按照解码管线规定的流程进行工作。进一步的,计算机设备中还可包括渲染管线;所谓的渲染管线又可称为渲染流水线,是显卡内部处理图形信号相互独立的并行处理单元。若计算机设备中包括渲染管线,则在通过图像处理方案解码得到任一图像的纹理数据后,还可进一步通过该渲染管线根据解码得到的纹理数据实现该任一图像的渲染显示。

[0042] 本申请实施例所提出的图像处理方案,主要是利用计算机设备中的显卡着色器来解码任一图像的压缩纹理资源的。参见图1a所示,该图像处理方案的解码流程大致如下:首先,可对计算机设备中的解码管线进行初始化,并加载着色器解码文件(shader解码文件);在成功加载着色器解码文件后,可将该着色器解码文件传输至计算机设备的显卡着色器中。当将任一图像的压缩纹理资源和该压缩纹理资源的相关解码参数,通过输入着色器缓存块输入至计算机设备中的显卡着色器后,显卡着色器可分别为该压缩纹理资源中的每个压缩纹理块分配一个目标工作组(即被分配了压缩纹理块的工作组)。

[0043] 然后,可将各个压缩纹理块分别派发给对应的目标工作组;由各个目标工作组并行地根据相关解码参数和着色器解码文件对相应的压缩纹理块进行解码处理,从而实现该任一图像的压缩纹理资源的解码处理,得到该任一图像的纹理数据。例如以 4×4 的block(压缩纹理块)为例,并采用“B”表示每个压缩纹理块,参见图1b所示:可将压缩纹理资源中的压缩纹理块11派发给显卡着色器中的目标工作组21进行解码处理,从而得到采用31所标识的解码数据;可将压缩纹理资源中的压缩纹理块12派发给显卡着色器中的目标工作组22进行解码处理,从而得到采用32所标识的解码数据,以此类推。在得到任一图像的纹理数据后,显卡着色器还可通过输出着色器缓存块输出该任一图像的纹理数据;可选的,若显卡着色器中在解码压缩纹理资源的过程中,存在一个或多个压缩纹理块被解码失败的情况,则显卡着色器还可通过输出着色器缓存块输出解码过程中所涉及的错误数据,以使相关技术人员及时根据该错误数据分析解码失败的原因。

[0044] 经实践表明,本申请实施例所提出的图像处理方案可至少具有如下有益效果:①通过引入显卡着色器实现整个解码过程,使得整个解码过程无需中央处理器的参与,这样可有效节省中央处理器的资源,从而有效减少中央处理器的资源占用率,提升计算机设备的运行效率。②通过并行地调用显卡着色器中的各个目标工作组的方式,可实现每个压缩纹理块的并行解码,从而有效提升解码效率。

[0045] 基于上述的描述,本申请实施例提出一种图像处理方法,该图像处理方法可以在上述所提及的计算机设备中执行;该计算机设备中可包括渲染管线、解码管线和显卡着色器,且该显卡着色器按照解码管线规定的流程进行工作。请参见图2,该图像处理方法可包括以下步骤S201-S204:

[0046] S201,获取目标图像的目标压缩纹理资源。

[0047] 其中,目标图像可以是以下任一种图像:目标游戏中的目标游戏画面图像、任一影视剧的视频图像、任一用户的自拍图像、任一景点的风景图像,等等。目标压缩纹理资源是采用压缩纹理格式对目标图像进行编码得到的,该目标压缩纹理资源包括多个压缩纹理块。需说明的是,目标图像包括的是纹理格式的纹理数据,此处的纹理格式可以例如是RGBA格式;其中,RGBA中的R表示红色通道,G表示绿色通道,B表示蓝色通道,A表示透明通道。采用压缩纹理格式对目标图像进行编码的过程,可理解成将目标图像的纹理数据从纹理格式转换成压缩纹理格式的过程;那么相应的,对目标压缩纹理资源进行解码的本质,可理解成将目标压缩纹理资源从压缩纹理格式转换成纹理格式的处理。

[0048] 在具体实现中,若计算机设备的本地空间中未预先存储该目标图像的目标压缩纹理资源,则步骤S201的具体实施方式可以是:向存储有该目标压缩纹理资源的其他设备发送资源获取请求,以请求其他设备返回该目标压缩纹理资源;然后,计算机设备可接收该目标压缩纹理资源。若计算机设备的本地空间中预先存储有该目标图像的目标压缩纹理资源,则计算机设备可通过内部的渲染管线从本地空间中加载该目标压缩纹理资源,以实现目标压缩纹理资源的获取;即步骤S201的具体实施方式可以是:渲染管线从计算机设备的本地空间中,加载该目标图像的目标压缩纹理资源。

[0049] S202,在显卡着色器中为多个压缩纹理块分配用于解码的多个目标工作组。

[0050] 由前述可知,显卡着色器中可包括M个工作组;每个工作组均具有解码处理能力,且每个工作组具有一个序号标识。那么在具体实现中,可遍历多个压缩纹理块,并确定当前遍历的当前压缩纹理块。考虑到显卡着色器中的工作组的数量和压缩纹理块的数量可能相同,也可能不同;且当显卡着色器的工作组的数量少于压缩纹理块的数量时,可能会导致在为当前压缩纹理块分配目标工作组时,显卡着色器中不存在未被分配的工作组。基于此,为了合理地为当前遍历的当前压缩纹理块分配目标工作组,可先检测显卡着色器中的M个工作组中是否存在未被分配的剩余工作组,从而根据检测结果采用不同的分配逻辑为当前压缩纹理块分配目标工作组,具体如下:

[0051] 若显卡着色器中的M个工作组中存在未被分配的一个或多个剩余工作组,则可在一个或多个剩余工作组中为当前压缩纹理块分配一个用于解码的目标工作组。在一种实施方式中,可在一个或多个剩余工作组中随机选取一个剩余工作组,作为用于解码当前压缩纹理块的目标工作组。另一种实施方式中,也可按照序号标识从小到大的顺序,在一个或多个剩余工作组中选取序号标识最小的剩余工作组,作为用于解码当前压缩纹理块的目标工

工作组。另一种实施方式中,若显卡着色器中的各个工作组的解码能力存在差异,则还可根据各个剩余工作组的解码能力,从一个或多个剩余工作组中选取解码能力最高的剩余工作组,或者选取解码能力大于能力阈值的剩余工作组,作为用于解码当前压缩纹理块的目标工作组。

[0052] 若该M个工作组中的各个工作组均已被分配一个或多个压缩纹理块,则可在M个工作组中为当前压缩纹理块分配一个用于解码的目标工作组。在一种实施方式中,可在M个工作组中随机选取一个工作组,作为用于解码当前压缩纹理块的目标工作组。另一种实施方式中,可根据各个工作组被分配的压缩纹理块的纹理块数量,在M个工作组中选取纹理块数量最少的工作组,作为用于解码当前压缩纹理块的目标工作组;通过此实施方式,可有效均衡各个工作组的纹理块数量,减少当前压缩纹理块等待解码的时长,从而提升解码效率。另一种实施方式中,若显卡着色器中的各个工作组的解码能力存在差异,则还可根据各个工作组的解码能力,从M个工作组中选取解码能力最高的工作组,或者选取解码能力大于能力阈值的工作组,作为用于解码当前压缩纹理块的目标工作组。

[0053] 需要说明的是:①基于上述描述可知,一个目标工作组可被分配一个或多个压缩纹理块。其中,当显卡着色器中的工作组的数量(即M)大于或等于目标压缩纹理资源中的压缩纹理块的数量时,一个目标工作组可被分配一个压缩纹理块;当显卡着色器中的工作组的数量(即M)小于目标压缩纹理资源中的压缩纹理块的数量时,存在部分或全部目标工作组被分配至少两个压缩纹理块。②基于上述描述可知,本申请实施例是依次遍历目标压缩纹理资源中的各个压缩纹理块,从而依次为各个压缩纹理块分配目标工作组的;但在其他具体实现中,也可采用批量分配的方式,同时为各个压缩纹理块分配目标工作组,对此步骤限定。

[0054] 例如,设目标压缩纹理资源中的压缩纹理块的数量为N,N为大于1的整数;当M(显卡着色器中的工作组的数量)大于或等于N时,可在显卡着色器的M个工作组中批量选取N个用于解码的目标工作组,然后按照一个目标工作组对应一个压缩纹理块的原则,将N个目标工作组分配给N个压缩纹理块。当M小于N时,可将M个工作组均作为目标工作组,并在目标压缩纹理资源中批量选取M个压缩纹理块,然后按照一个目标工作组对应一个压缩纹理块的原则,将M个目标工作组分配给被选取的M个压缩纹理块。若N个压缩纹理块中未被选取的压缩纹理块的数量(即N-M)小于M,则可在M个工作组中选取N-M个目标工作组,然后按照一个目标工作组对应一个压缩纹理块的原则,将N-M个目标工作组分配给N-M个未被选取的压缩纹理块;若N-M大于或等于M,则可将M个工作组均作为目标工作组,并在N-M个未被选取的压缩纹理块中批量选取M个压缩纹理块,然后按照一个目标工作组对应一个压缩纹理块的原则,将M个目标工作组分配给被选取的M个压缩纹理块;以此类推,直至为每个压缩纹理块均分配了目标工作组为止。

[0055] S203,将每个压缩纹理块派发给对应的目标工作组。

[0056] S204,并行地调用显卡着色器中的各个目标工作组,对相应的压缩纹理块进行解码处理,以得到目标图像的目标纹理数据,目标纹理数据包括每个压缩纹理块对应的解码数据。

[0057] 在具体实现中,任一目标工作组对相应的压缩纹理块进行解码处理的实施方式如下:获取着色器解码文件,以及获取关于目标压缩纹理资源的目标解码参数。其中,该着色

器解码文件中可包括：用于实现对不同压缩纹理格式下的压缩纹理资源进行解码的多条算法指令；目标压缩纹理资源的目标解码参数可用于指示：目标图像所采用的压缩纹理格式。然后，可根据目标解码参数从着色器解码文件中获取用于解码目标压缩纹理资源的算法指令，并采用获取到的算法指令对相应的压缩纹理块进行解码处理，得到该相应的压缩纹理块所对应的解码数据。

[0058] 举例来说，着色器解码文件中可包括：用于实现对ASTC格式下的压缩纹理资源进行解码的第一算法指令，用于实现对ETC格式下的压缩纹理资源进行解码的第二算法指令，等等。若目标压缩纹理资源的目标解码参数用于指示：目标图像所采用的压缩纹理格式为ASTC格式，则可根据该目标解码参数从着色器文件中获取第一算法指令，并采用该第一算法指令对相应的压缩纹理块进行解码处理。若目标压缩纹理资源的目标解码参数用于指示：目标图像所采用的压缩纹理格式为ETC格式，则可根据该目标解码参数从着色器文件中获取第二算法指令，并采用该第二算法指令对相应的压缩纹理块进行解码处理。

[0059] 可选的，为便于解码处理，可将上述所提及的解码处理的具体流程封装成一个解码管线中的解码接口（glDispatchCompute），使得任一目标工作组可直接调用该解码接口对相应的压缩纹理块进行解码处理。

[0060] 可选的，计算机设备在得到目标纹理数据后，还可将该目标纹理数据上传至区块链，以防止目标纹理数据被篡改。其中，区块链由多个区块组成，如图3a所示；创始块中包括区块头和区块主体，区块头中存储有输入信息特征值、版本号、时间戳和难度值，区块主体中存储有输入信息；创始块的下一区块以创始块为父区块，下一区块中同样包括区块头和区块主体，区块头中存储有当前区块的输入信息特征值、父区块的区块头特征值、版本号、时间戳和难度值，并以此类推，使得区块链中每个区块中存储的区块数据均与父区块中存储的区块数据存在关联，保证了区块中输入信息的安全性。基于此，将目标纹理数据上传至区块链中的具体实施方式可以是：

[0061] 若计算机设备位于区块链网络内，则先可将目标纹理数据添加至目标区块的区块主体中，并对区块主体中的目标纹理数据进行哈希运算，得到默克尔哈希值。其次，可采用随机算法生成一个随机数，并采用计算得到的默克尔哈希值、随机数、版本号、上一区块哈希值、当前时间戳以及当前难度值组成目标区块的区块头部，如图3b所示。其中，版本号是指区块链中相关区块协议的版本信息；上一区块哈希值是指上一区块的区块头部的特征值；当前时间戳是指组成区块头部时的系统时间；当前难度值是指计算的难度值，该难度值在固定时间段内为定值，并在超出固定时间段后再次进行确定。然后，可采用特征值算法（如SHA256算法）对区块头部所包含的内容进行一次或多次哈希运算，得到目标区块的区块头部的特征值；此处的哈希运算的次数可根据计算难度确定，计算难度越大，哈希运算的次数越多。在基于上述步骤得到目标区块后，可将该目标区块广播给区块链网络中的各个共识节点进行共识处理；在通过共识处理后，将目标区块添加至区块链上。应理解的是，若计算机设备未位于区块链网络内，则可将目标纹理数据发送给区块链网络中的任一共识节点，由该任一共识节点执行上述步骤，以将该目标纹理数据存储至区块链中。

[0062] 可选的，计算机设备还可采用云技术对目标纹理数据进行云存储处理。其中，云技术（Cloud technology）是指在广域网或局域网内将硬件、软件、网络等系列资源统一起来，实现数据的计算、储存、处理和共享的一种托管技术。云技术基于云计算商业模式应用的网

络技术、信息技术、整合技术、管理平台技术、应用技术等的总称,可以组成资源池,按需所用,灵活便利。云计算技术将变成重要支撑。相应的,云存储(cloud storage)是在云计算概念上延伸和发展出来的一个新的概念,分布式云存储系统(以下简称存储系统)是指通过集群应用、网格技术以及分布存储文件系统等功能,将网络中大量各种不同类型的存储设备(存储设备也称之为存储节点)通过应用软件或应用接口集合起来协同工作,共同对外提供数据存储和业务访问功能的一个存储系统。

[0063] 本申请实施例通过引入显卡着色器,并在显卡着色器中预先部署多个相互独立的工作组;使得在获取到目标图像的目标压缩纹理资源后,可在显卡着色器中为目标压缩纹理资源中的每个压缩纹理块分别分配一个用于解码的目标工作组,并将每个压缩纹理块派发给对应的目标工作组。然后,可通过并行地调用显卡着色器中的各个目标工作组的方式,对相应的压缩纹理块进行解码处理;这样的处理方式,不仅可实现目标压缩纹理资源的解码处理,还可实现每个压缩纹理块的并行解码,从而有效提升解码效率。另外,通过显卡着色器实现整个解码过程,使得整个解码过程无需中央处理器的参与,这样可有效节省中央处理器的资源,从而有效减少中央处理器的资源占用率,提升计算机设备的运行效率。

[0064] 请参见图4,是本申请实施例提供的另一种图像处理方法的流程示意图,该图像处理方法可以在上述所提及的计算机设备中执行;该计算机设备中可包括渲染管线、解码管线和显卡着色器,且该显卡着色器按照解码管线规定的流程进行工作。请参见图4,该图像处理方法可包括以下步骤S401-S407:

[0065] S401,渲染管线从计算机设备的本地空间中,加载目标图像的目标压缩纹理资源。在具体实施过程中,该渲染管线可以通过调用纹理资源加载接口(例如glTexCompressedImage2D(一种加载压缩纹理资源的接口)),从计算机设备的本地空间中,加载目标图像的目标压缩纹理资源。

[0066] S402,渲染管线将目标压缩纹理资源派发给解码管线。

[0067] 在一种实施方式中,渲染管线可在成功加载到目标压缩纹理资源后,可直接将目标压缩纹理资源派发给解码管线。另一种实施方式中,可为计算机设备设置一个显卡解码模式(或称为显卡解码功能),所谓的显卡解码模式是指通过显卡着色器进行解码处理的模式;当显卡解码模式处于开启状态时,表明此时的显卡着色器支持解码处理,当显卡解码模式处于关闭状态时,表明此时的显卡着色器不支持解码处理,此时可通过显卡驱动中央处理器(简称为显卡驱动CPU)进行解码。可见,通过设置该显卡解码模式,可实现用户根据自身需求来开启或关闭该显卡解码模式,从而灵活选择是否需通过显卡着色器进行解码处理。基于此,当计算机设备具有显卡解码模式时,渲染管线在执行步骤S402之前,可先检测显卡解码模式所处的模式状态,该模式状态包括模式开启状态或模式关闭状态。若模式状态为模式开启状态,则渲染管线执行将目标压缩纹理资源派发给解码管线的步骤,即执行步骤S402。若模式状态为模式关闭状态,则渲染管线可调用显卡驱动中央处理器对目标压缩纹理资源中的各个压缩纹理块进行解码处理,以得到目标图像的目标纹理数据;即此情况下,在通过显卡驱动中央处理器得到目标纹理数据后,渲染管线可直接执行步骤S407,即此情况下无需执行步骤S402-S406。

[0068] S403,解码管线在成功接收到目标压缩纹理资源后,向显卡着色器发送解码通知,以通知显卡着色器执行步骤S404。

[0069] S404,在显卡着色器中为多个压缩纹理块分配用于解码的多个目标工作组,一个目标工作组被分配一个或多个压缩纹理块。

[0070] S405,显卡着色器将每个压缩纹理块派发给对应的目标工作组。

[0071] S406,显卡着色器并行地调用显卡着色器中的各个目标工作组,对相应的压缩纹理块进行解码处理,以得到目标图像的目标纹理数据,该目标纹理数据包括每个压缩纹理块对应的解码数据。

[0072] S407,渲染管线根据目标纹理数据进行图像渲染,得到目标图像。

[0073] 由前述可知,目标纹理数据可以是由显卡着色器通过步骤S404-S406解码得到的,也可以是由渲染管线调用显卡驱动中央处理器进行解码处理得到的。若目标纹理数据是由渲染管线调用显卡驱动中央处理器解码得到的,则步骤S407的具体实施方式可以是:渲染管线直接调用渲染接口(如glDraw(一种用于图像渲染的接口))根据目标纹理数据进行图像渲染,得到目标图像。若目标纹理数据是由显卡着色器进行解码处理得到的,则步骤S407的具体实施方式可参见下述描述:

[0074] 在一种具体实现中,渲染管线可具有物理的第一纹理显存空间,解码管线可具有物理的第二纹理显存空间。其中,第一纹理显存空间是指用于存储渲染管线即将处理的数据,或经渲染管线处理得到的数据的显存空间;第二纹理显存空间是指用于存储解码管线即将处理的数据,或经解码管线处理得到的数据的显存空间;所谓的显存空间是指:位于显卡中的存储空间。在此具体实现下,显卡着色器中的任一目标工作组对相应的压缩纹理块进行解码处理后,可将解码得到的解码数据缓存至该第二纹理显存空间中;在各个压缩纹理块对应的解码数据均被缓存至第二纹理显存空间后,即目标图像的目标纹理数据被缓存至第二纹理显存空间后,解码管线可通过中央处理器(central processing unit,CPU),将该目标图像的目标纹理数据从第二纹理显存空间拷贝到第一纹理显存空间中。或者,在目标图像的目标纹理数据被缓存至第二纹理显存空间后,解码管线可通知渲染管线,由渲染管线通过CPU将目标图像的目标纹理数据从第二纹理显存空间中拷贝到第一纹理显存空间中。那么此情况下,步骤S407的具体实施方式可以是:渲染管线在第一纹理显存空间中根据目标纹理数据进行图像渲染,得到目标图像。

[0075] 进一步的,经实践表明:在通过CPU拷贝目标纹理数据时,可能会造成1-5ms的性能损失;那么为了可以充分发挥解码管线的优势,减少性能损失,本申请实施例针对解码管线和渲染管线,提出了物理空间相同但虚拟空间不同的共用方式,使得在渲染管线在将同一个物理的纹理存储空间用于存储压缩纹理资源以及其他数据的同时,解码管线可将该纹理存储空间用于缓存显卡着色器通过解码处理所得到的目标纹理数据,这样就可以避免因为CPU拷贝目标纹理数据所造成的性能损失。基于此,本申请实施例提出了另一种具体实现方式,具体参见下述描述:

[0076] 另一种具体实现中,渲染管线可具有虚拟的第一显卡地址空间,解码管线可具有虚拟的第二显卡地址空间;其中,第一显卡地址空间和第二显卡地址空间,与同一个纹理存储空间相映射,该纹理存储空间本质上是一个物理的存储空间(即真实存在的存储空间)。在此具体实现下,显卡着色器中的任一目标工作组对相应的压缩纹理块进行解码处理后,可根据第二显卡地址空间和纹理存储空间之间的映射关系,将任一目标工作组解码得到的解码数据缓存至纹理存储空间中。那么此情况下,步骤S407的具体实施方式可以是:在各个

压缩纹理块对应的解码数据均被存储至纹理存储空间后,渲染管线根据第一显卡地址空间和纹理存储空间之间的映射关系,在纹理存储空间中根据目标纹理数据进行图像渲染,得到目标图像。可选的,为了使得渲染管线可及时进行图像渲染,在显卡着色器将各个压缩纹理块对应的解码数据均存储至纹理存储空间后,解码管线可向渲染管线发送一个解码结束通知,以通知渲染管线在纹理存储空间中根据纹理数据进行图像渲染。

[0077] 其中,将第一显卡地址空间和第二显卡地址空间映射到同一个物理的纹理存储空间的具体方式如下:渲染管线在计算机设备的本地空间中,为目标图像分配纹理存储空间;渲染管线将第一显卡地址空间映射到纹理存储空间,并将纹理存储空间的空间标识(如纹理存储空间的空间地址、空间序列号等)传递给解码管线;解码管线根据该空间标识,将第二显卡地址空间映射到纹理存储空间。需说明的是,本申请实施例对渲染管线和解码管线执行空间映射操作的时间点不作限定。例如,渲染管线可以在加载目标图像的目标压缩纹理资源的过程中,执行纹理存储空间的分配和空间映射操作,也可以在加载目标纹理资源之前或之后,执行纹理存储空间的分配和映射操作。又如,解码管线可以是在接收到渲染管线传递的空间标识后,立即执行空间映射操作;也可以在接收到渲染管线派发的目标压缩纹理资源后,再执行空间映射操作;进一步的,空间标识和目标压缩纹理资源可以由渲染管线同时发送给解码管线的,也可以是分别发送给解码管线的,对此不作限定。

[0078] 下面结合图5所示的空间映射的示意图,对将第一显卡地址空间和第二显卡地址空间映射到同一个物理的纹理存储空间的具体方式作进一步示意性说明;参见图5所示,渲染管线可在通过glTexCompressedImage2D(一种加载压缩纹理资源的接口)加载目标图像的目标压缩纹理资源的过程中,分配纹理存储空间(Physical Pages),并将该纹理存储空间与渲染管线的第一显卡地址空间进行映射;以及向解码管线发送纹理存储空间的空间标识,以触发解码管线将目标压缩纹理资源解码到该纹理存储空间。解码管线可接收渲染管线发送的空间标识,并通过astc_cs_decode(一种解码准备接口)将空间标识所指示的纹理存储空间与解码管线的第二显卡地址空间进行映射。

[0079] 可选的,若目标压缩纹理资源中的全部或部分压缩纹理块被显卡着色器解码失败,则可调用显卡驱动中央处理器(即显卡驱动CPU)对目标压缩纹理资源中的各个压缩纹理块进行解码处理,以得到目标图像的目标纹理数据。

[0080] 本申请实施例通过显卡着色器实现整个解码过程,使得整个解码过程无需中央处理器的参与,这样可有效节省中央处理器的资源,从而有效减少中央处理器的资源占用率,提升计算机设备的运行效率。进一步的,通过并行地调用显卡着色器中的各个目标工作组的方式,对相应的压缩纹理块进行解码处理;可有效提升解码效率。另外,通过解码管线和渲染管线共用同一物理的纹理存储空间的方式,还可有效减少因CPU拷贝数据所造成的性能损失,且提升图像渲染的及时性,以及整个图像处理过程的处理效率。

[0081] 进一步的,在上述图4所示的方法实施例中,当目标图像为目标游戏中的目标游戏画面图像时,由于目标游戏通常具有游戏资源包,该游戏资源包用于存储目标游戏在运行过程中所需的各种数据资源;因此,游戏开发者在采用压缩纹理格式对目标游戏中的各帧游戏画面图像进行编码,得到各帧游戏画面图像的压缩纹理资源后,也可预先对目标游戏中的一帧或多帧游戏画面的压缩纹理资源进行离线解码,并采用解码得到的各纹理数据和相应的游戏画面图像的图像标识构建目标游戏的离线解码数据集,以及将该离线解码数据

集添加至目标游戏的游戏资源包中。

[0082] 在此情况下,渲染管线在执行步骤S402之前,也可先获取目标游戏的离线解码数据集;具体的,可从目标游戏的游戏资源包中获取该目标游戏的离线解码数据集,该离线解码数据集中包括:目标游戏中的一帧或多帧游戏画面图像的图像标识,以及每个图像标识所关联的纹理数据;任一图像标识所关联的纹理数据,是对任一图像标识所指示的游戏画面图像的压缩纹理资源进行离线解码得到的。然后,渲染管线采用目标图像的图像标识对离线解码数据集中的图像标识进行标识命中处理,以检测该离线解码数据集中是否包括目标游戏画面图像的目标纹理数据。若未成功命中,则渲染管线执行步骤S402。若成功命中,则渲染管线从离线解码数据集中,加载被命中的图像标识所关联的纹理数据,并将加载到的纹理数据作为目标图像的目标纹理数据,这样可有效缩短目标纹理数据的获取时长,提升解码效率。

[0083] 可选的,由上述步骤S402的相关描述可知,渲染管线在执行步骤S402之前,可先执行检测显卡解码模式所处的模式状态的步骤;那么在此情况下,渲染管线可以在检测显卡解码模式所处的模式状态之前或之后,执行标识命中处理的操作。以在检测显卡解码模式所处的模式状态之前,执行标识命中处理的操作为例,渲染管线的处理逻辑如下:渲染管线采用目标图像的图像标识对离线解码数据集中的图像标识进行标识命中处理;若未成功命中,则渲染管线执行检测显卡解码模式所处的模式状态的步骤;若成功命中,则渲染管线从离线解码数据集中,加载被命中的图像标识所关联的纹理数据,并将加载到的纹理数据作为目标图像的目标纹理数据。

[0084] 由前述可知,在图像处理过程中,可能会存在渲染管线调用显卡驱动中央处理器对目标压缩纹理资源中的各个压缩纹理块进行解码处理的情况;例如,渲染管线在检测到显卡解码模式所处的模式状态为关闭状态后,可调用显卡驱动中央处理器对目标压缩纹理资源中的各个压缩纹理块进行解码处理;又如,当目标压缩纹理资源中的全部或部分压缩纹理块被显卡着色器解码失败时,渲染管线可调用显卡驱动中央处理器对目标压缩纹理资源中的各个压缩纹理块进行解码处理。那么进一步的,考虑到显卡驱动中央处理器可能具有解码目标压缩纹理资源的解码能力,也可能不具有该解码能力;当显卡驱动中央处理器不具有该解码能力时,若仍调用显卡驱动中央处理器对目标压缩纹理资源进行解码处理,则会导致目标压缩纹理资源被解码失败。因此,当显卡驱动中央处理器不具有该解码能力时,渲染管线则可直接调用目标游戏的游戏引擎中央处理器(即游戏引擎CPU)对目标压缩纹理资源进行解码。这样的处理方式,一方面可保证目标压缩纹理资源被成功解码,从而保证目标图像被成功渲染显示;另一方面,可避免因显卡驱动中央处理器执行无用的解码处理,而浪费处理资源的问题。

[0085] 基于此,渲染管线在调用显卡驱动中央处理器对目标压缩纹理资源中的各个压缩纹理块进行解码处理之前,还可检测显卡驱动中央处理器(即显卡驱动CPU)关于目标压缩纹理资源的解码能力;若检测到显卡驱动中央处理器具有该解码能力,则渲染管线执行调用显卡驱动中央处理器对目标压缩纹理资源中的各个压缩纹理块进行解码处理的步骤;若检测到显卡驱动中央处理器不具备解码能力,则渲染管线调用目标游戏的游戏引擎中央处理器,对目标压缩纹理资源中的各个压缩纹理块进行解码处理,以得到目标图像的目标纹理数据。需说明的是,本申请实施例对渲染管线执行检测显卡驱动中央处理器关于目标压

缩纹理资源的解码能力的具体时间点不作限定。例如在其他实施例中,渲染管线也可是在获取目标游戏的离线解码数据集之前,执行解码能力检测的步骤;在此情况下,渲染管线的处理逻辑如下:检测显卡驱动中央处理器(即显卡驱动CPU)关于目标压缩纹理资源的解码能力;若检测到显卡驱动中央处理器具有该解码能力,则渲染管线执行获取目标游戏的离线解码数据集的步骤;若检测到显卡驱动中央处理器不具备解码能力,则渲染管线调用目标游戏的游戏引擎中央处理器,对目标压缩纹理资源中的各个压缩纹理块进行解码处理,以得到目标图像的目标纹理数据。

[0086] 需要说明的是,上述所提及的目标游戏可以是普通游戏或者云游戏。所谓的普通游戏则是指:直接在玩家用户使用的终端设备内安装的游戏客户端中运行的游戏。所谓的云游戏又可称为游戏点播(gaming on demand),是一种以云计算为基础的游戏方式,此处的云计算是一种基于互联网的计算方式;在云游戏场景下,游戏并不在玩家用户的游戏客户端,而是在云游戏服务器中运行;由云游戏服务器将游戏场景所涉及的游戏画面编码为视频流,通过网络传输给玩家用户的游戏客户端进行播放。其中,当目标游戏为普通游戏时,上述所提及的计算机设备为玩家用户所使用的终端设备;当目标游戏为云游戏时,上述所提及的计算机设备为运行云游戏的云游戏服务器。

[0087] 参见图6a所示,云游戏服务器中可包括:设备硬件层、虚拟化层(Hypervisor)以及一个或多个容器,等等。其中,设备硬件层可包括但不限于:CPU(中央处理器)、显卡、网络通信模块以及用于存储的本地空间等;显卡中可包括显卡着色器、解码管线以及渲染管线,等等。虚拟化层主要用于以软件的方式,实现一套和物理主机环境完全一样的虚拟环境;云游戏服务器可通过该虚拟化层部署并运行一个或多个容器,所谓的容器是指操作系统级虚拟化的一种类型,其可用于承载操作系统。每个容器可与一个或多个游戏客户端相连接,且每个容器可用于运行一个或多个云游戏;每个容器在运行任一云游戏的过程中,可传输任一云游戏的游戏画面至相连接的游戏客户端中进行显示。

[0088] 基于上述描述,本申请实施例进一步提出了图6b所示的图像处理方法。在本申请实施例中,主要以目标游戏为云游戏,即计算机设备为云游戏服务器为例进行说明;即图6b所示的图像处理方法可在云游戏服务器中执行,具体可以在云游戏服务器中的显卡中执行。参见图6b所示,该图像处理方法的大致流程如下:

[0089] 首先,可初始化(创建)解码管线;具体的,可在目标游戏的初始化过程中,创建独立线程,并通过该独立线程初始化解码管线。其次,可加载着色器解码文件,以及显卡着色器创建解码管理器,该解码管理器用于等待并接收目标图像的目标压缩纹理资源。

[0090] 渲染管线在加载到目标压缩纹理资源后,可检测显卡驱动CPU是否具有解码能力。若不具有,则渲染管线调用游戏引擎CPU对目标压缩纹理资源进行解码处理,以得到目标图像的目标纹理数据。若具有,则渲染管线可进一步检测目标图像的图像标识是否命中离线解码数据集,以检测该离线解码数据集中是否包括目标游戏画面图像的目标纹理数据。若成功命中,则渲染管线从离线解码数据集中加载目标图像的目标纹理数据;若未成功命中,则渲染管线可检测显卡解码模式的模式状态,以检测显卡着色器是否支持解码处理。若检测到显卡着色器不支持解码处理,则渲染管线调用显卡驱动CPU对目标压缩纹理资源进行解码处理,以得到目标图像的目标纹理数据;若检测到显卡着色器不支持解码处理,则渲染管线向解码管线派发目标图像的目标压缩纹理资源。

[0091] 相应的,解码管线可在接收到渲染管线派发的目标压缩纹理资源后,可将该目标压缩纹理资源通过解码管理器发送给显卡着色器,并通知显卡着色器对该目标压缩纹理资源进行解码处理。具体的,显卡着色器可为目标压缩纹理资源中的各个压缩纹理块分别分配用于解码的目标工作组,以及将每个压缩纹理块派发给对应的目标工作组;然后,并行地调用显卡着色器中的各个目标工作组,对相应的压缩纹理块进行解码处理,以得到目标图像的目标纹理数据。解码管线可实时或周期性地检测显卡着色器是否解码成功;若解码成功,则通知渲染管线根据目标图像的目标纹理数据进行图像渲染,若解码失败,则通知渲染管线通过显卡驱动CPU对目标压缩纹理资源进行解码处理,并在解码得到目标纹理数据后,再根据目标图像的目标纹理数据进行图像渲染。

[0092] 针对上述图2、图4以及图6b所示的方法实施例的相关描述内容,需说明以下两点:

[0093] 1) 本申请实施例可采用独立封装成SDK (Software Development Kit,软件开发工具包)的方式,对上述图2、图4以及图6b所示的方法实施例中所提及的显卡着色器所执行的解码流程进行封装处理。也就是说,显卡着色器所执行的解码流程可被独立封装在一个软件工具包(可命名为cs decoder library)中,以使在需解码任一图像的压缩纹理资源时,通过调用该软件工具包执行解码处理。其中,该软件工具包被调用的原理示意图可参见图7所示:在目标游戏加载后,可先调用sys vulkan library(系统vulkan渲染接口),其次可通过该sys vulkan library调用GPU vulkan library(GPU vulkan渲染接口),然后可通过GPU vulkan library调用cs decoder library,以实现解码处理。或者,也可先调用sys EGL/GELS libraries(系统EGL/GELS渲染接口),其次可通过该sys EGL/GELS libraries调用GPU EGL/GELS libraries(GPU EGL/GELS渲染接口),然后可通过GPU EGL/GELS libraries调用cs decoder library,以实现解码处理。

[0094] 其中,上述所提及的vulkan是一个跨平台的2D和3D绘图应用程序接口;GELS是OpenGL ES(OpenGL for Embedded Systems)的简称,是OpenGL(一种用于渲染2D、3D矢量图形的跨语言、跨平台的应用程序编程接口)的三维图形API的子集;EGL是OpenGL ES渲染接口和本地窗口系统(native platform window system)之间的一个中间接口层。通过独立封装SDK的方式,可具有如下有益效果:①避免对显卡驱动进行过多修改,而引入潜在的渲染风险;②与渲染管线保持独立,避免破坏游戏的渲染状态机;③以显卡着色器支持显卡驱动中的openGLES渲染接口和vulkan渲染接口,避免重复造车。

[0095] 2) 综上可知,当目标图像为目标游戏中的目标游戏画面图像时,本申请实施例针对目标图像的目标压缩资源而言,可提供如下几种解码方式:①通过游戏引擎CPU解码,②通过显卡驱动CPU解码,③通过显卡着色器解码,④通过离线解码数据集解码;其中,解码方式①和解码方式②的本质均是:通过CPU解码。本申请实施例采用了ASTC格式下的不同尺寸的压缩纹理资源,对通过CPU解码、通过离线解码数据集解码以及通过显卡着色器解码这三种解码方式所需的解码时长进行测试,测试结果可参见下述表1所示:

[0096] 表1

[0097]

压缩纹理资源的尺寸 (压缩资源块为6×6)	CPU	离线解码数据集	显卡着色器
256×256	3ms	0ms	1ms
512×2512	12ms	0ms	2ms
1024×1024	50ms	2ms	5ms

2048×2048	79ms	2ms	9ms
-----------	------	-----	-----

[0098] 其中,表1中的“ms”表示毫秒;通过表1可知,通过离线解码数据集解码的解码方式所需的解码时长最少,其解码性能最优;而通过显卡着色器解码的解码方式所需的解码时长则远小于通过CPU解码的解码方式所需的解码时长,即通过显卡着色器解码的解码方式的解码性能优于通过CPU 解码的解码方式的解码性能。

[0099] 本申请实施例通过利用云游戏服务器中的显卡的强大浮点运算和并行计算能力,可实现实时地快速解码目标游戏中的压缩纹理格式下的目标压缩纹理资源,从而实现目标图像的快速渲染显示。这样可有效提升游戏流畅性,减少CPU资源占用率,以及解决因CPU解码过慢所造成的游戏卡顿和低帧率问题,以及云游戏服务器因单路游戏占用CPU资源过多而造成的游戏并发路数受限的运营成本问题。另外,通过在云游戏服务器中执行整个解码过程,使得整个解码过程可不受终端类型和操作系统的限制,可全端生效;并且,由于本申请实施例支持通过显卡着色器对目标压缩纹理资源进行解码处理,因此可无需游戏开发者特地对云游戏的实现逻辑作修改和适配,以制作出符合云游戏服务器的解码条件的离线解码数据集,具有较强的适用性和兼容性;同时,还可解决因离线解码各游戏画面图像的压缩纹理资源,而导致云游戏的游戏资源包的存储内存占用过大的问题。

[0100] 基于上述图像处理方法实施例的描述,本申请实施例还公开了一种图像处理装置,所述图像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)。该图像处理装置可以执行图2、图4或图6b所示的方法。请参见图8,所述图像处理装置可以运行如下单元:

[0101] 获取单元801,用于获取目标图像的目标压缩纹理资源,所述目标压缩纹理资源是采用压缩纹理格式对所述目标图像进行编码得到的,所述目标压缩纹理资源包括多个压缩纹理块;

[0102] 处理单元802,用于在显卡着色器中为所述多个压缩纹理块分配用于解码的多个目标工作组,一个目标工作组被分配一个或多个压缩纹理块;以及,将每个压缩纹理块派发给对应的目标工作组;

[0103] 所述处理单元802,还用于并行地调用所述显卡着色器中的各个目标工作组,对相应的压缩纹理块进行解码处理,以得到所述目标图像的目标纹理数据,所述目标纹理数据包括所述每个压缩纹理块对应的解码数据。

[0104] 在一种实施方式中,所述显卡着色器设置于计算机设备中,所述计算机设备还包括渲染管线和解码管线,所述显卡着色器按照所述解码管线规定的流程进行工作;相应的,处理单元802还可用于:

[0105] 通过渲染管线根据所述目标纹理数据进行图像渲染,得到所述目标图像。

[0106] 另一种实施方式中,所述渲染管线具有虚拟的第一显卡地址空间,所述解码管线具有虚拟的第二显卡地址空间;

[0107] 其中,所述第一显卡地址空间和所述第二显卡地址空间,与同一个纹理存储空间相映射。

[0108] 另一种实施方式中,处理单元802还可用于:所述显卡着色器中的任一目标工作组对相应的压缩纹理块进行解码处理后,根据所述第二显卡地址空间和所述纹理存储空间之间的映射关系,将所述任一目标工作组解码得到的解码数据缓存至所述纹理存储空间中;

[0109] 相应的,处理单元802在用于通过渲染管线根据所述目标纹理数据进行图像渲染,得到所述目标图像时,可具体用于:

[0110] 在各个压缩纹理块对应的解码数据均被存储至所述纹理存储空间后,通过所述渲染管线根据所述第一显卡地址空间和所述纹理存储空间之间的映射关系,在所述纹理存储空间中根据所述目标纹理数据进行图像渲染,得到所述目标图像。

[0111] 另一种实施方式中,处理单元802还可用于:

[0112] 通过所述渲染管线在所述计算机设备的本地空间中,为所述目标图像分配所述纹理存储空间;

[0113] 通过所述渲染管线将所述第一显卡地址空间映射到所述纹理存储空间,并将所述纹理存储空间的空间标识传递给所述解码管线;

[0114] 通过所述解码管线根据所述空间标识,将所述第二显卡地址空间映射到所述纹理存储空间。

[0115] 另一种实施方式中,获取单元801在用于获取目标图像的目标压缩纹理资源时,可具体用于:通过所述渲染管线从所述计算机设备的本地空间中,加载目标图像的目标压缩纹理资源;

[0116] 相应的,处理单元802还可用于:

[0117] 通过所述渲染管线将所述目标压缩纹理资源派发给所述解码管线;

[0118] 通过所述解码管线在成功接收到所述目标压缩纹理资源后,向所述显卡着色器发送解码通知,以通知所述显卡着色器执行在显卡着色器中为所述多个压缩纹理块分配用于解码的多个目标工作组的步骤。

[0119] 另一种实施方式中,所述计算机设备具有显卡解码模式,所述显卡解码模式是指通过所述显卡着色器进行解码处理的模式;相应的,处理单元802还可用于:

[0120] 通过所述渲染管线检测所述显卡解码模式所处的模式状态,所述模式状态包括模式开启状态或模式关闭状态;

[0121] 若所述模式状态为所述模式开启状态,则通过所述渲染管线执行将所述目标压缩纹理资源派发给所述解码管线的步骤;

[0122] 若所述模式状态为所述模式关闭状态,则通过所述渲染管线调用显卡驱动中央处理器对所述目标压缩纹理资源中的各个压缩纹理块进行解码处理,以得到所述目标图像的目标纹理数据。

[0123] 另一种实施方式中,所述目标图像为目标游戏中的目标游戏画面图像;相应的,处理单元802还可用于:

[0124] 通过所述渲染管线获取所述目标游戏的离线解码数据集,所述离线解码数据集中包括:所述目标游戏中的一帧或多帧游戏画面图像的图像标识,以及每个图像标识所关联的纹理数据;任一图像标识所关联的纹理数据,是对所述任一图像标识所指示的游戏画面图像的压缩纹理资源进行离线解码得到的;

[0125] 通过所述渲染管线采用所述目标图像的图像标识对所述离线解码数据集中的图像标识进行标识命中处理;

[0126] 若未成功命中,则通过所述渲染管线执行检测所述显卡解码模式所处的模式状态的步骤;

[0127] 若成功命中,则通过所述渲染管线从所述离线解码数据集中,加载被命中的图像标识所关联的纹理数据,并将加载到的纹理数据作为所述目标图像的目标纹理数据。

[0128] 另一种实施方式中,处理单元802还可用于:

[0129] 通过所述渲染管线检测所述显卡驱动中央处理器关于所述目标压缩纹理资源的解码能力;

[0130] 若检测到所述显卡驱动中央处理器具有所述解码能力,则通过所述渲染管线执行获取所述目标游戏的离线解码数据集的步骤;

[0131] 若检测到所述显卡驱动中央处理器不具备所述解码能力,则通过所述渲染管线调用所述目标游戏的游戏引擎中央处理器,对所述目标压缩纹理资源中的各个压缩纹理块进行解码处理,以得到所述目标图像的目标纹理数据。

[0132] 另一种实施方式中,处理单元802还可用于:

[0133] 若所述目标压缩纹理资源中的全部或部分压缩纹理块被所述显卡着色器解码失败,则通过调用显卡驱动中央处理器对所述目标压缩纹理资源中的各个压缩纹理块进行解码处理,以得到所述目标图像的目标纹理数据。

[0134] 另一种实施方式中,所述显卡着色器中包括M个工作组,M为大于1的整数;相应的,处理单元802在用于在显卡着色器中为所述多个压缩纹理块分配用于解码的多个目标工作组时,可具体用于:

[0135] 遍历所述多个压缩纹理块,并确定当前遍历的当前压缩纹理块;

[0136] 若所述显卡着色器中的M个工作组中存在未被分配的一个或多个剩余工作组,则在所述一个或多个剩余工作组中为所述当前压缩纹理块分配一个用于解码的目标工作组;

[0137] 若所述M个工作组中的各个工作组均已被分配一个或多个压缩纹理块,则在所述M个工作组中为所述当前压缩纹理块分配一个用于解码的目标工作组。

[0138] 另一种实施方式中,所述显卡着色器所执行的解码流程被独立封装在一个软件工具包中,以使在需解码任一图像的压缩纹理资源时,通过调用所述软件工具包执行解码处理。

[0139] 根据本申请的另一个实施例,图8所示的图像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,基于图像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。

[0140] 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2中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中所示的图像处理装置设备,以及来实现本申请实施例的图像处理方法。所述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。

[0141] 本申请实施例通过引入显卡着色器,并在显卡着色器中预先部署多个相互独立的

工作组;使得在获取到目标图像的目标压缩纹理资源后,可在显卡着色器中为目标压缩纹理资源中的每个压缩纹理块分别分配一个用于解码的目标工作组,并将每个压缩纹理块派发给对应的目标工作组。然后,可通过并行地调用显卡着色器中的各个目标工作组的方式,对相应的压缩纹理块进行解码处理;这样的处理方式,不仅可实现目标压缩纹理资源的解码处理,还可实现每个压缩纹理块的并行解码,从而有效提升解码效率。另外,通过显卡着色器实现整个解码过程,使得整个解码过程无需中央处理器的参与,这样可有效节省中央处理器的资源,从而有效减少中央处理器的资源占用率,提升计算机设备的运行效率。

[0142] 基于上述方法实施例以及装置实施例的描述,本申请实施例还提供一种计算机设备。请参见图9,该计算机设备至少包括显卡901、输入接口902、输出接口903以及计算机存储介质904;且该计算机设备内的显卡901、输入接口902、输出接口903以及计算机存储介质904可通过总线或其他方式连接。其中,显卡901中可进一步包括图形处理器9011、显卡着色器9012、解码管线9013以及渲染管线9014。计算机存储介质904可以存储在计算机设备的存储器中,所述计算机存储介质904用于存储计算机程序,所述计算机程序包括程序指令,所述处理器901用于执行所述计算机存储介质904存储的程序指令。进一步的,计算机设备还可包括中央处理器905(Central Processing Unit,CPU),该中央处理器905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,具体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。

[0143] 在一个实施例中,本申请实施例所述的显卡901可用于进行一系列的图像处理,具体包括:获取目标图像的目标压缩纹理资源,所述目标压缩纹理资源是采用压缩纹理格式对所述目标图像进行编码得到的,所述目标压缩纹理资源包括多个压缩纹理块;在显卡着色器中为所述多个压缩纹理块分配用于解码的多个目标工作组,一个目标工作组被分配一个或多个压缩纹理块;以及,将每个压缩纹理块派发给对应的目标工作组;并行地调用所述显卡着色器中的各个目标工作组,对相应的压缩纹理块进行解码处理,以得到所述目标图像的目标纹理数据,所述目标纹理数据包括所述每个压缩纹理块对应的解码数据,等等。

[0144] 本申请实施例还提供了一种计算机存储介质(Memory),所述计算机存储介质是计算机设备中的记忆设备,用于存放程序和数据。可以理解的是,此处的计算机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。计算机存储介质提供存储空间,该存储空间存储了计算机设备的操作系统。并且,在该存储空间中还存放了适于被显卡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一个以上的计算机程序(包括程序代码)。需要说明的是,此处的计算机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存储介质。

[0145] 在一个实施例中,可由显卡901加载并执行计算机存储介质中存放的一条或多条指令,以实现上述有关图2、图4或图6b所示的图像处理方法实施例中的方法的相应步骤;具体实现中,计算机存储介质中的一条或多条指令由显卡901加载并执行如下步骤:

[0146] 获取目标图像的目标压缩纹理资源,所述目标压缩纹理资源是采用压缩纹理格式对所述目标图像进行编码得到的,所述目标压缩纹理资源包括多个压缩纹理块;

[0147] 在显卡着色器中为所述多个压缩纹理块分配用于解码的多个目标工作组,一个目标工作组被分配一个或多个压缩纹理块;以及,将每个压缩纹理块派发给对应的目标工作

组；

[0148] 并行地调用所述显卡着色器中的各个目标工作组，对相应的压缩纹理块进行解码处理，以得到所述目标图像的目标纹理数据，所述目标纹理数据包括所述每个压缩纹理块对应的解码数据。

[0149] 在一种实施方式中，所述显卡着色器设置于计算机设备中，所述计算机设备还包括渲染管线和解码管线，所述显卡着色器按照所述解码管线规定的流程进行工作；相应的，所述一条或多条指令还可由显卡901加载并执行如下步骤：

[0150] 通过渲染管线根据所述目标纹理数据进行图像渲染，得到所述目标图像。

[0151] 另一种实施方式中，所述渲染管线具有虚拟的第一显卡地址空间，所述解码管线具有虚拟的第二显卡地址空间；

[0152] 其中，所述第一显卡地址空间和所述第二显卡地址空间，与同一个纹理存储空间相映射。

[0153] 另一种实施方式中，所述一条或多条指令还可由显卡901加载并执行如下步骤：

[0154] 所述显卡着色器中的任一目标工作组对相应的压缩纹理块进行解码处理后，根据所述第二显卡地址空间和所述纹理存储空间之间的映射关系，将所述任一目标工作组解码得到的解码数据缓存至所述纹理存储空间中；

[0155] 相应的，在通过渲染管线根据所述目标纹理数据进行图像渲染，得到所述目标图像时，所述一条或多条指令可由显卡901加载并具体执行如下步骤：

[0156] 在各个压缩纹理块对应的解码数据均被存储至所述纹理存储空间后，通过所述渲染管线根据所述第一显卡地址空间和所述纹理存储空间之间的映射关系，在所述纹理存储空间中根据所述目标纹理数据进行图像渲染，得到所述目标图像。

[0157] 另一种实施方式中，所述一条或多条指令还可由显卡901加载并执行如下步骤：

[0158] 通过所述渲染管线在所述计算机设备的本地空间中，为所述目标图像分配所述纹理存储空间；

[0159] 通过所述渲染管线将所述第一显卡地址空间映射到所述纹理存储空间，并将所述纹理存储空间的空间标识传递给所述解码管线；

[0160] 通过所述解码管线根据所述空间标识，将所述第二显卡地址空间映射到所述纹理存储空间。

[0161] 另一种实施方式中，在获取目标图像的目标压缩纹理资源时，所述一条或多条指令可由显卡901加载并具体执行如下步骤：通过所述渲染管线从所述计算机设备的本地空间中，加载目标图像的目标压缩纹理资源；

[0162] 相应的，所述一条或多条指令还可由显卡901加载并执行如下步骤：

[0163] 通过所述渲染管线将所述目标压缩纹理资源派发给所述解码管线；

[0164] 通过所述解码管线在成功接收到所述目标压缩纹理资源后，向所述显卡着色器发送解码通知，以通知所述显卡着色器执行在显卡着色器中为所述多个压缩纹理块分配用于解码的多个目标工作组的步骤。

[0165] 另一种实施方式中，所述计算机设备具有显卡解码模式，所述显卡解码模式是指通过所述显卡着色器进行解码处理的模式；相应的，所述一条或多条指令还可由显卡901加载并执行如下步骤：

[0166] 通过所述渲染管线检测所述显卡解码模式所处的模式状态,所述模式状态包括模式开启状态或模式关闭状态;

[0167] 若所述模式状态为所述模式开启状态,则通过所述渲染管线执行将所述目标压缩纹理资源派发给所述解码管线的步骤;

[0168] 若所述模式状态为所述模式关闭状态,则通过所述渲染管线调用显卡驱动中央处理器对所述目标压缩纹理资源中的各个压缩纹理块进行解码处理,以得到所述目标图像的目标纹理数据。

[0169] 另一种实施方式中,所述目标图像为目标游戏中的目标游戏画面图像;相应的,所述一条或多条指令还可由显卡901加载并执行如下步骤:

[0170] 通过所述渲染管线获取所述目标游戏的离线解码数据集,所述离线解码数据集中包括:所述目标游戏中的一帧或多帧游戏画面图像的图像标识,以及每个图像标识所关联的纹理数据;任一图像标识所关联的纹理数据,是对所述任一图像标识所指示的游戏画面图像的压缩纹理资源进行离线解码得到的;

[0171] 通过所述渲染管线采用所述目标图像的图像标识对所述离线解码数据集中的图像标识进行标识命中处理;

[0172] 若未成功命中,则通过所述渲染管线执行检测所述显卡解码模式所处的模式状态的步骤;

[0173] 若成功命中,则通过所述渲染管线从所述离线解码数据集中,加载被命中的图像标识所关联的纹理数据,并将加载到的纹理数据作为所述目标图像的目标纹理数据。

[0174] 另一种实施方式中,所述一条或多条指令还可由显卡901加载并执行如下步骤:

[0175] 通过所述渲染管线检测所述显卡驱动中央处理器关于所述目标压缩纹理资源的解码能力;

[0176] 若检测到所述显卡驱动中央处理器具有所述解码能力,则通过所述渲染管线执行获取所述目标游戏的离线解码数据集的步骤;

[0177] 若检测到所述显卡驱动中央处理器不具备所述解码能力,则通过所述渲染管线调用所述目标游戏的引擎中央处理器,对所述目标压缩纹理资源中的各个压缩纹理块进行解码处理,以得到所述目标图像的目标纹理数据。

[0178] 另一种实施方式中,所述一条或多条指令还可由显卡901加载并执行如下步骤:

[0179] 若所述目标压缩纹理资源中的全部或部分压缩纹理块被所述显卡着色器解码失败,则通过调用显卡驱动中央处理器对所述目标压缩纹理资源中的各个压缩纹理块进行解码处理,以得到所述目标图像的目标纹理数据。

[0180] 另一种实施方式中,所述显卡着色器中包括M个工作组,M为大于1的整数;相应的,在显卡着色器中为所述多个压缩纹理块分配用于解码的多个目标工作组时,所述一条或多条指令可由显卡901加载并具体执行如下步骤:

[0181] 遍历所述多个压缩纹理块,并确定当前遍历的当前压缩纹理块;

[0182] 若所述显卡着色器中的M个工作组中存在未被分配的一个或多个剩余工作组,则在所述一个或多个剩余工作组中为所述当前压缩纹理块分配一个用于解码的目标工作组;

[0183] 若所述M个工作组中的各个工作组均已被分配一个或多个压缩纹理块,则在所述M个工作组中为所述当前压缩纹理块分配一个用于解码的目标工作组。

[0184] 另一种实施方式中,所述显卡着色器所执行的解码流程被独立封装在一个软件工具包中,以使在需解码任一图像的压缩纹理资源时,通过调用所述软件工具包执行解码处理。

[0185] 本申请实施例通过引入显卡着色器,并在显卡着色器中预先部署多个相互独立的工作组;使得在获取到目标图像的目标压缩纹理资源后,可在显卡着色器中为目标压缩纹理资源中的每个压缩纹理块分别分配一个用于解码的目标工作组,并将每个压缩纹理块派发给对应的目标工作组。然后,可通过并行地调用显卡着色器中的各个目标工作组的方式,对相应的压缩纹理块进行解码处理;这样的处理方式,不仅可实现目标压缩纹理资源的解码处理,还可实现每个压缩纹理块的并行解码,从而有效提升解码效率。另外,通过显卡着色器实现整个解码过程,使得整个解码过程无需中央处理器的参与,这样可有效节省中央处理器的资源,从而有效减少中央处理器的资源占用率,提升计算机设备的运行效率。

[0186] 需要说明的是,根据本申请的一个方面,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图2、图4或图6b所示的图像处理方法实施例方面的各种可选方式中提供的方法。

[0187] 并且,应理解的是,以上所揭露的仅为本申请较佳实施例而已,当然不能以此来限定本申请之权利范围,因此依本申请权利要求所作的等同变化,仍属本申请所涵盖的范围。

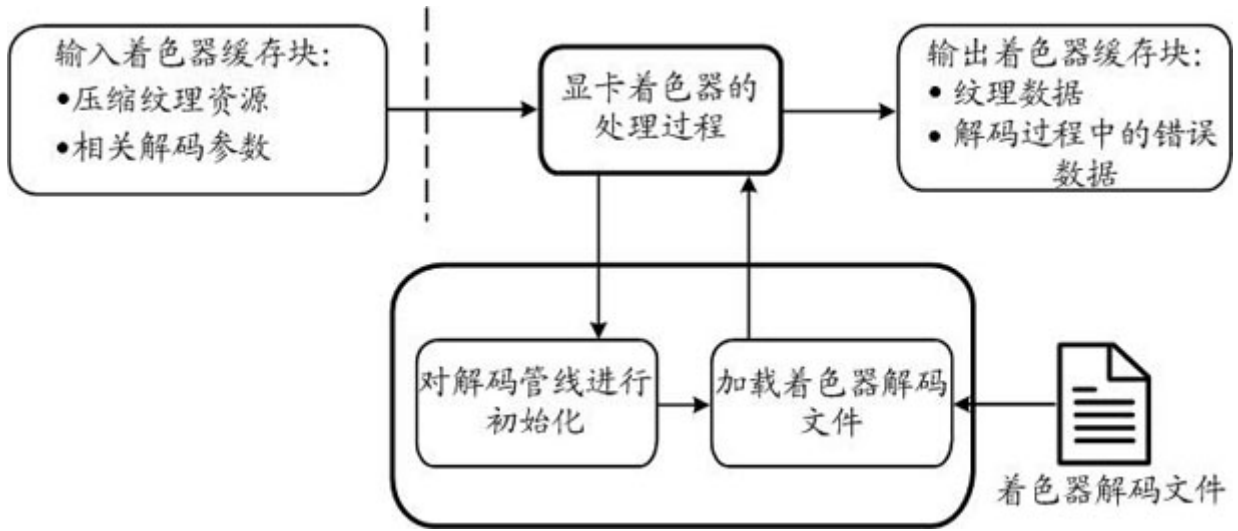


图1a

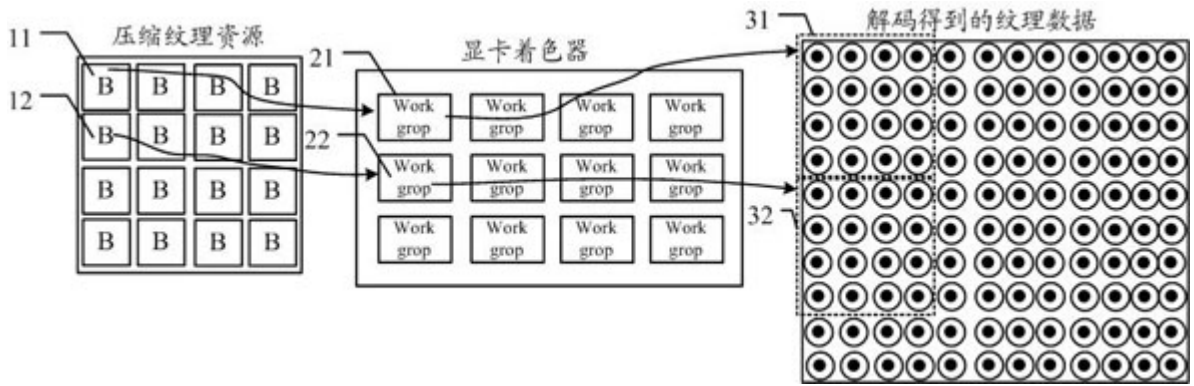


图1b

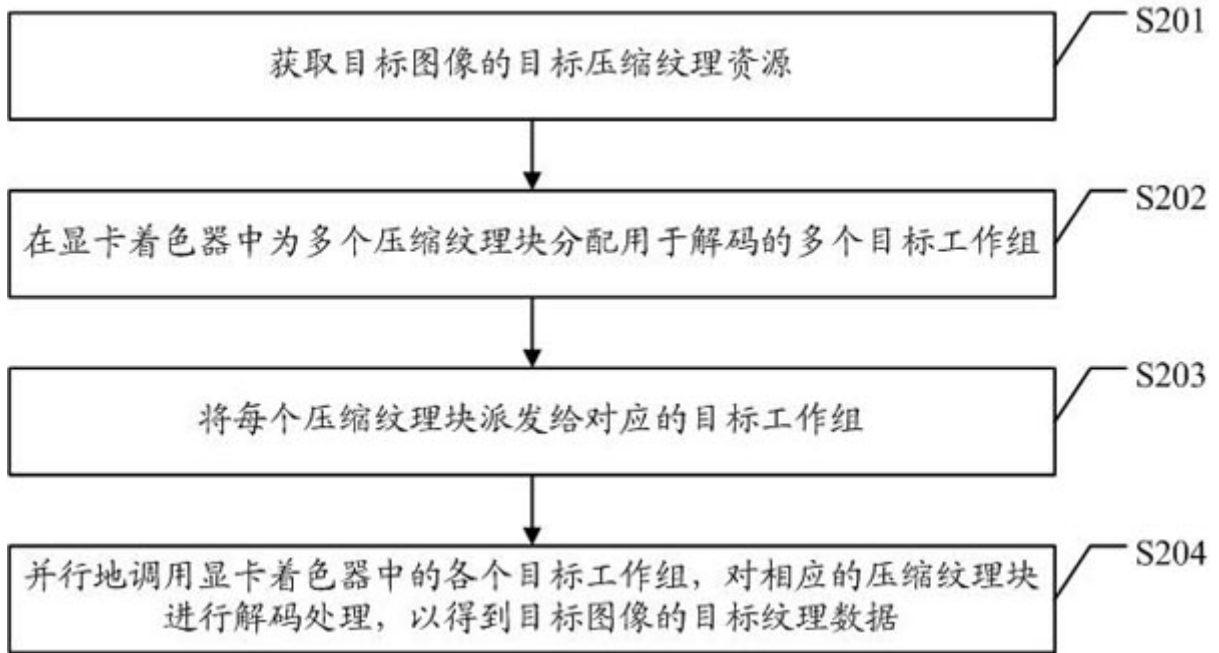


图2

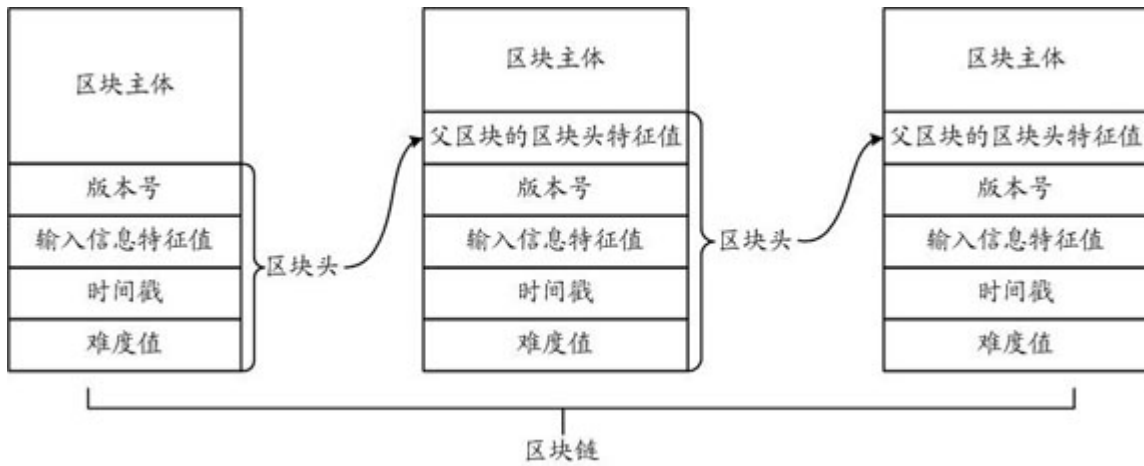


图3a

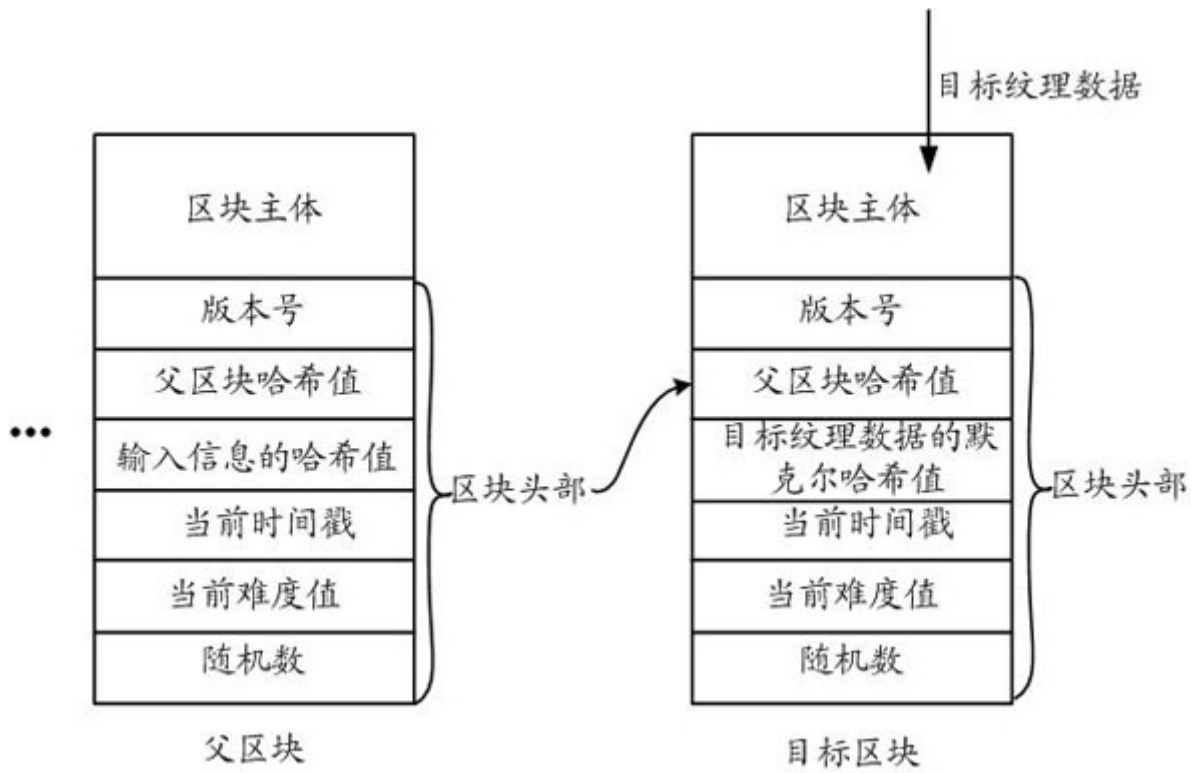


图3b

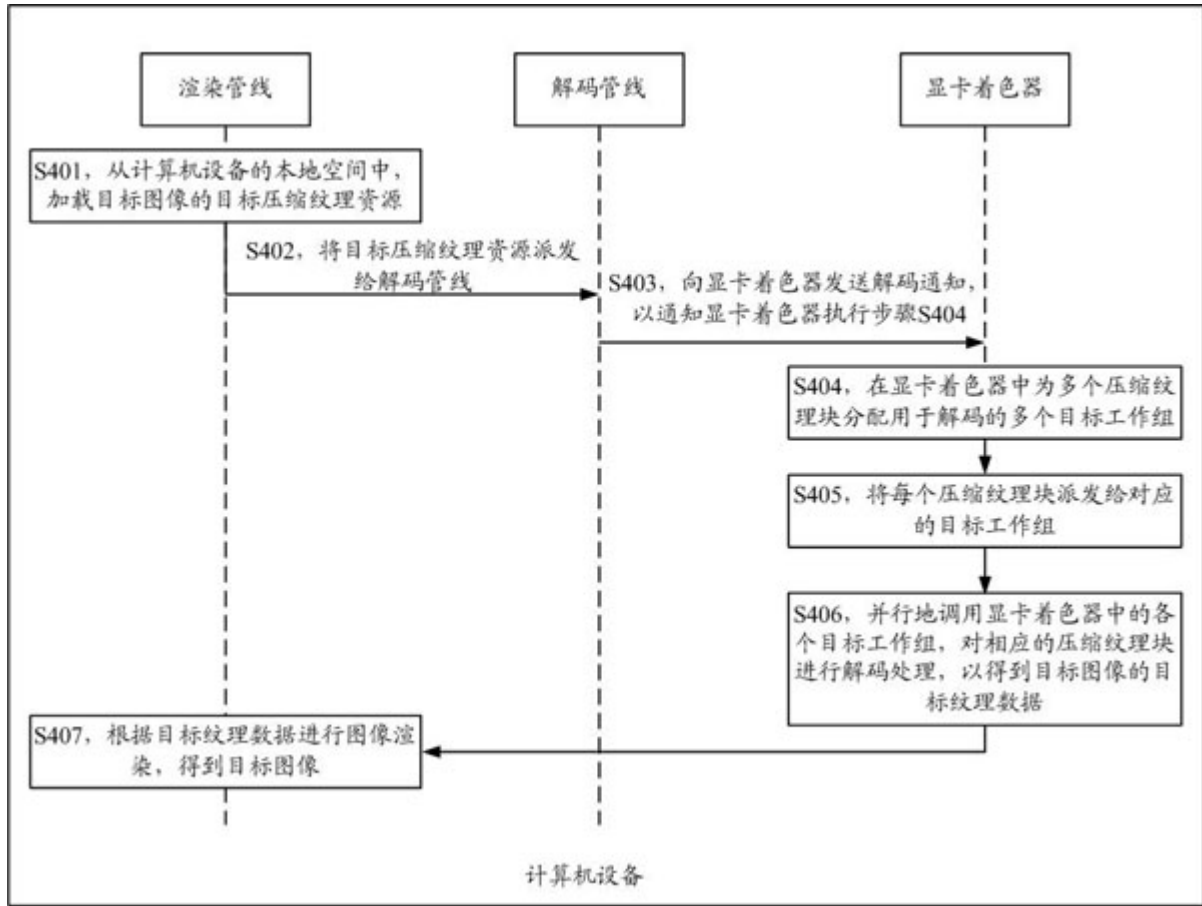


图4

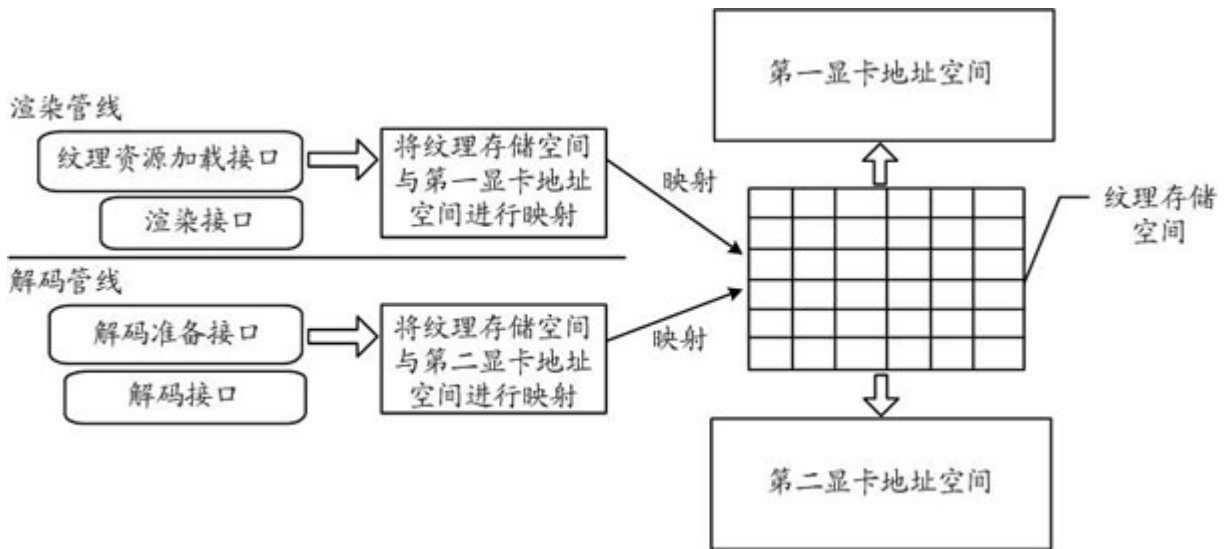


图5

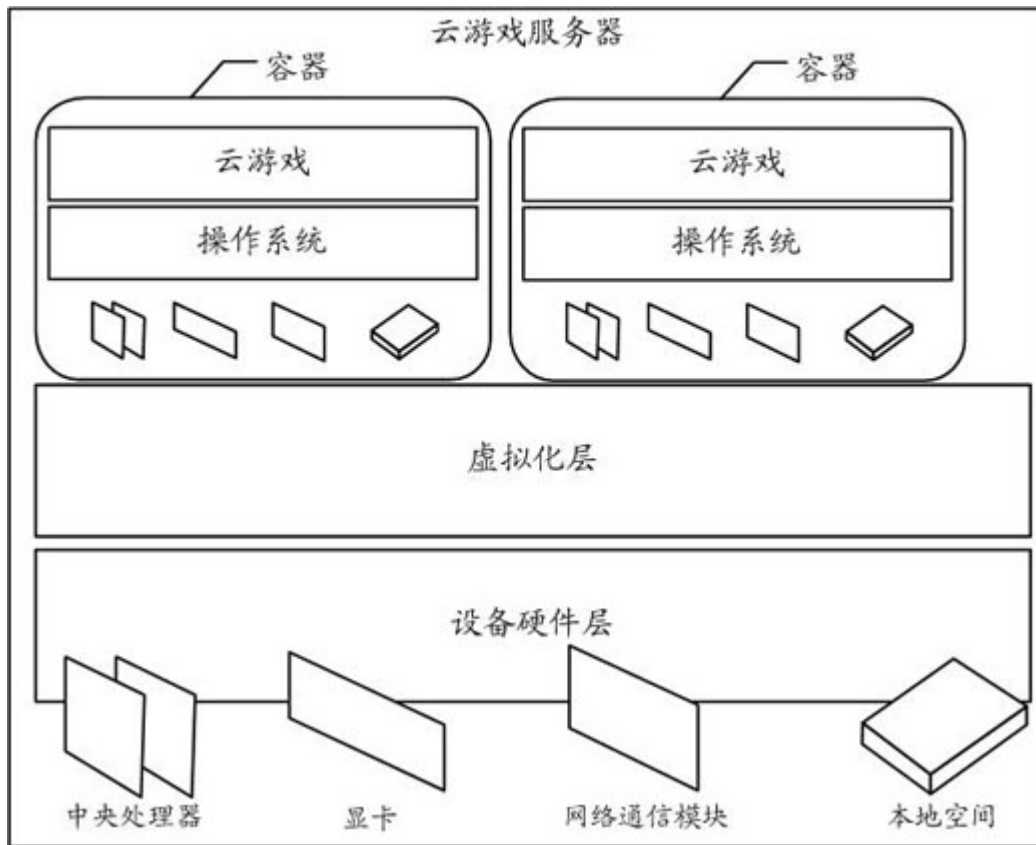


图6a

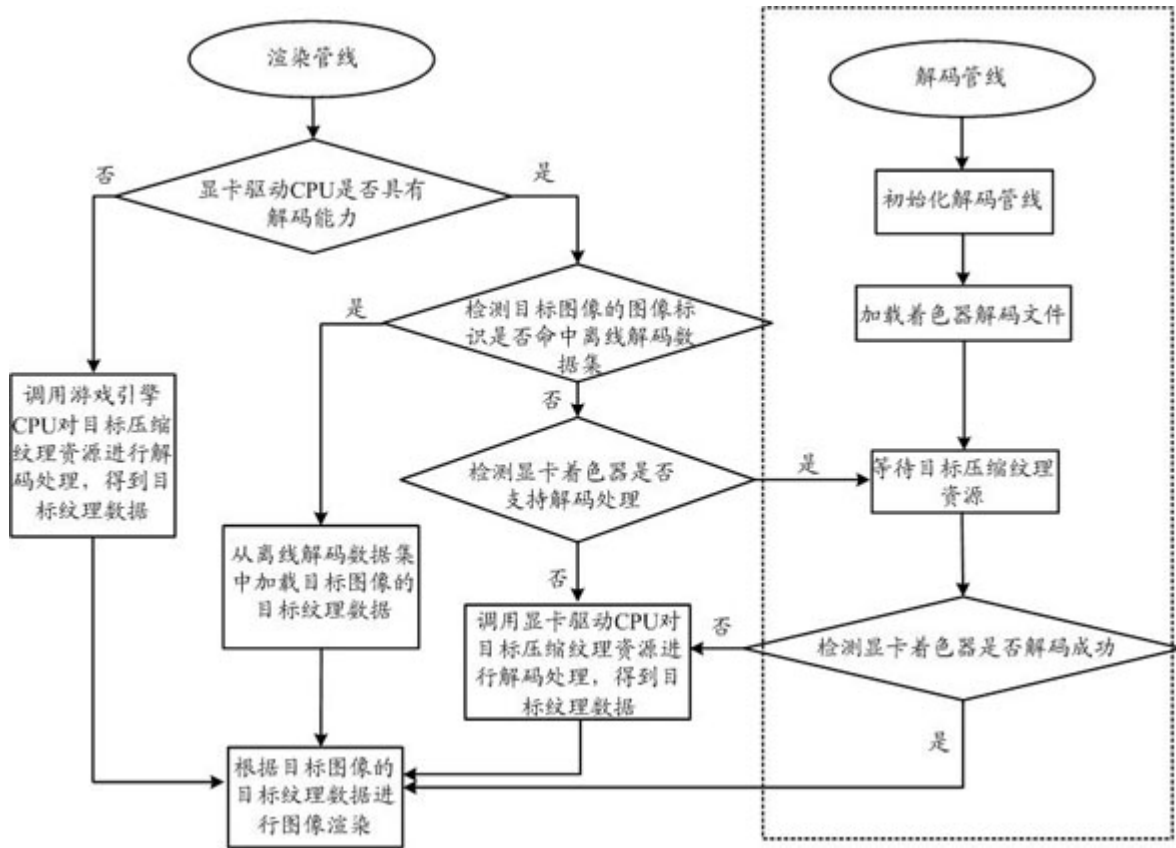


图6b

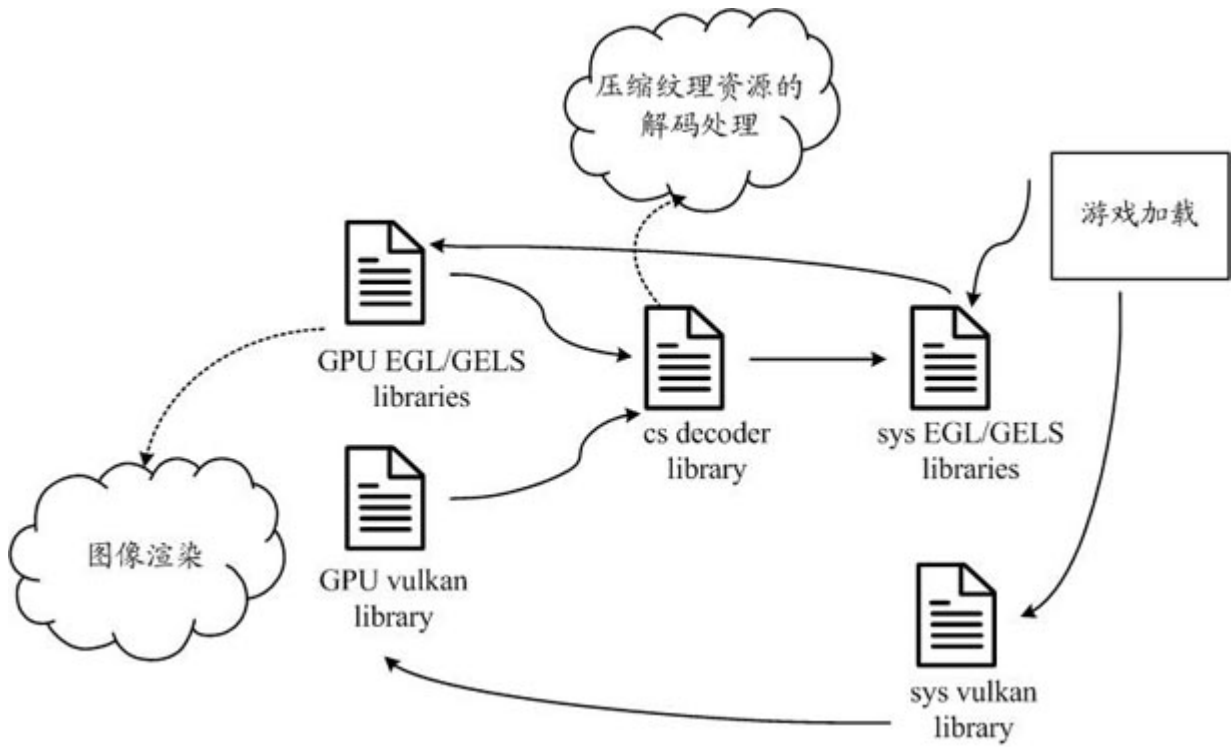


图7

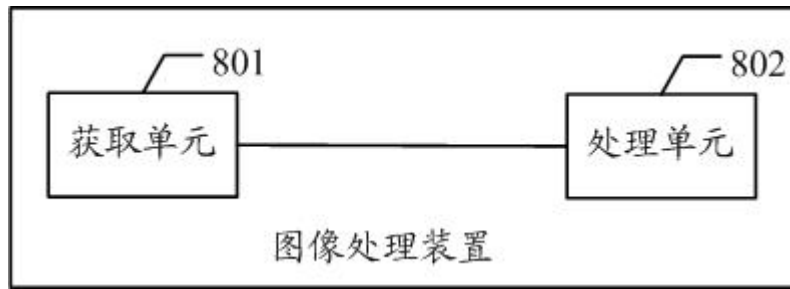


图8

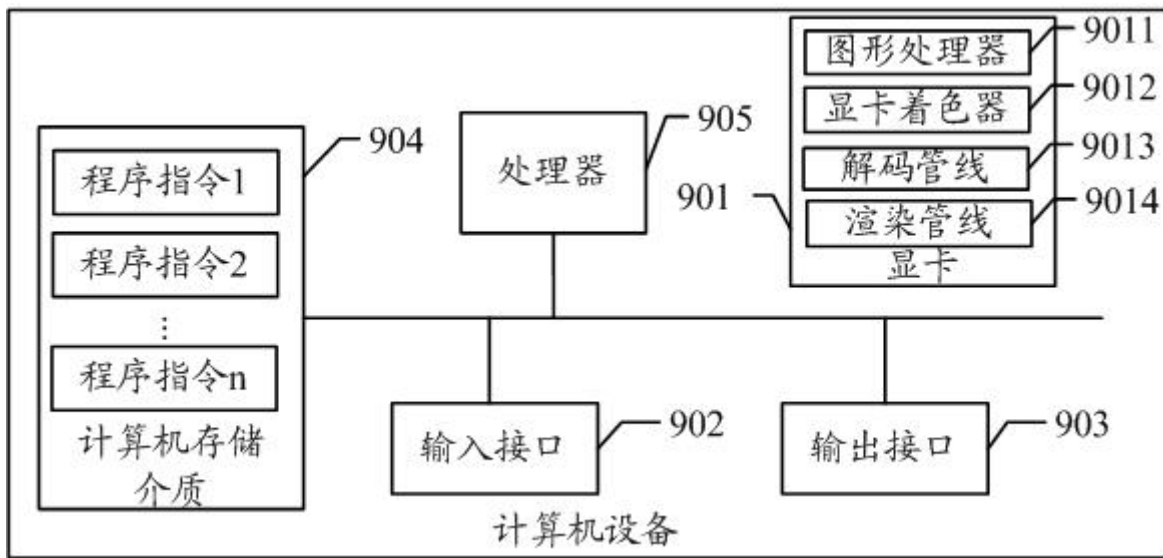


图9