



(12) 发明专利申请

(10) 申请公布号 CN 117065357 A

(43) 申请公布日 2023. 11. 17

(21) 申请号 202210504909.5

(22) 申请日 2022.05.10

(71) 申请人 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 沈咸飞

(74) 专利代理机构 华进联合专利商标代理有限
公司 44224

专利代理师 杨欢

(51) Int. Cl.

A63F 13/77 (2014.01)

G10L 19/16 (2013.01)

H04N 19/40 (2014.01)

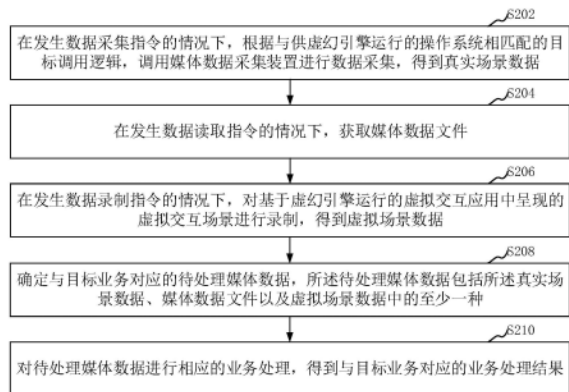
权利要求书3页 说明书16页 附图9页

(54) 发明名称

媒体数据处理方法、装置、计算机设备和存储介质

(57) 摘要

本申请涉及一种媒体数据处理方法、装置、计算机设备、存储介质和计算机程序产品。所述方法包括:根据与供虚幻引擎运行的操作系统相匹配的目标调用逻辑,调用媒体数据采集装置进行数据采集,得到真实场景数据;获取媒体数据文件;对基于虚幻引擎运行的虚拟交互应用中呈现的虚拟交互场景进行录制,得到虚拟场景数据;确定与目标业务对应的待处理媒体数据,所述待处理媒体数据包括所述真实场景数据、媒体数据文件以及虚拟场景数据中的至少一种;对所述待处理媒体数据进行相应的业务处理,得到与目标业务对应的业务处理结果。采用本方法能够适配不同的平台,并满足多样化的业务需求。



1. 一种媒体数据处理方法,其特征在于,所述方法包括:

在发生数据采集指令的情况下,根据与供虚幻引擎运行的操作系统相匹配的目标调用逻辑,调用媒体数据采集装置进行数据采集,得到真实场景数据;

在发生数据读取指令的情况下,获取媒体数据文件;

在发生数据录制指令的情况下,对基于虚幻引擎运行的虚拟交互应用中呈现的虚拟交互场景进行录制,得到虚拟场景数据;

确定与目标业务对应的待处理媒体数据,所述待处理媒体数据包括如下至少一种:所述真实场景数据、所述媒体数据文件以及所述虚拟场景数据;

对所述待处理媒体数据进行相应的业务处理,得到与所述目标业务对应的业务处理结果。

2. 根据权利要求1所述的方法,其特征在于,所述根据与供虚幻引擎运行的操作系统相匹配的目标调用逻辑,调用媒体数据采集装置进行数据采集,得到真实场景数据,包括:

确定数据采集通用接口,所述数据采集通用接口中封装有与多个不同操作系统分别对应的媒体数据采集装置相适配的调用逻辑;

确定所述数据采集通用接口中封装的、与所述操作系统的系统类型相匹配的目标调用逻辑;

基于所述目标调用逻辑,通过所述数据采集通用接口调用媒体数据采集装置进行数据采集,得到真实场景数据。

3. 根据权利要求2所述的方法,其特征在于,所述基于所述目标调用逻辑,通过所述数据采集通用接口调用媒体数据采集装置进行数据采集,得到真实场景数据,包括:

基于所述目标调用逻辑,向所述操作系统发起权限获取请求;

在根据所述权限获取请求获得调用授权的情况下,调用媒体数据采集装置进行数据采集;

接收所述媒体数据采集装置采集的真实场景数据。

4. 根据权利要求1所述的方法,其特征在于,所述对基于虚幻引擎运行的虚拟交互应用中呈现的虚拟交互场景进行录制,得到虚拟场景数据,包括:

通过图形处理器,对基于虚幻引擎运行的虚拟交互应用中的虚拟交互场景进行渲染,得到图像数据;

通过所述图形处理器将所述图像数据加载至预设缓存区中;

通过中央处理器异步读取所述预设缓存区中的图像数据,基于所述中央处理器读取的图像数据以实现采集操作,得到虚拟场景数据。

5. 根据权利要求1所述的方法,其特征在于,所述对所述待处理媒体数据进行相应的业务处理,得到与所述目标业务对应的业务处理结果之前,所述方法还包括:

确定与所述操作系统相匹配的加速接口;

通过与所述操作系统相匹配的加速接口对所述待处理媒体数据进行相应的解码处理,基于解码后的待处理媒体数据执行对所述待处理媒体数据进行相应的业务处理的步骤。

6. 根据权利要求1至5中任一项所述的方法,其特征在于,所述目标业务包括检测业务,与所述目标业务对应的待处理媒体数据包括所述真实场景数据和所述媒体数据文件中的至少一种;

所述对所述待处理媒体数据进行相应的业务处理,得到与所述目标业务对应的业务处理结果,包括:

确定适配于目标检测算法的目标格式;

将所述待处理媒体数据的数据格式转换成所述目标格式;

基于所述目标检测算法对所述待处理媒体数据中的待处理视频数据进行图像检测,得到视频检测结果,对所述待处理媒体数据中的待处理音频数据进行音频检测,得到音频检测结果;

基于所述视频检测结果和所述音频检测结果,确定与所述目标业务对应的业务处理结果。

7. 根据权利要求6所述的方法,其特征在于,所述将所述待处理媒体数据的数据格式转换成所述目标格式,包括:

启动子线程,通过所述子线程确定所述待处理媒体数据的原始格式,并通过逻辑转换函数,将所述待处理媒体数据从所述原始格式转换至目标格式;

在所述子线程执行完转换操作后,通知主线程已完成格式转换。

8. 根据权利要求6所述的方法,其特征在于,所述将所述待处理媒体数据的数据格式转换成所述目标格式,包括:

调用与所述操作系统相匹配的绘制函数,将所述待处理媒体数据输入至图形处理器中;

通过所述图形处理器进行批量的格式转换处理,以将所述待处理媒体数据的数据格式转换成所述目标格式;

将所述图形处理器中属于目标格式的待处理媒体数据回传至中央处理器。

9. 根据权利要求6所述的方法,其特征在于,所述业务处理结果包括驱动相关数据,所述方法还包括:

根据所述驱动相关数据,控制所述虚拟交互场景中的虚拟角色,按照所述待处理媒体数据中对象的动作进行运动。

10. 根据权利要求1至5中任一项所述的方法,其特征在于,所述目标业务包括播放业务,与所述目标业务对应的待处理媒体数据包括所述虚拟场景数据,还包括所述真实场景数据和所述媒体数据文件中的至少一种;

所述对所述待处理媒体数据进行相应的业务处理,得到与所述目标业务对应的业务处理结果,包括:

对所述真实场景数据和所述媒体数据文件中的至少一种进行格式转换,得到适应于所述虚幻引擎的目标视频数据;

在通过所述虚拟交互应用播放与所述虚拟场景数据对应的虚拟交互场景的过程中,叠加所述目标视频数据进行播放。

11. 根据权利要求1至5中任一项所述的方法,其特征在于,所述目标业务包括视频编码业务,与所述目标业务对应的待处理媒体数据包括所述虚拟场景数据,或者,所述待处理媒体数据还包括所述真实场景数据和所述媒体数据文件中的至少一种;

所述对所述待处理媒体数据进行相应的业务处理,得到与所述目标业务对应的业务处理结果,包括:

获取预设的编码状态；

启动子线程,通过所述子线程并按照所述编码状态,对所述待处理媒体数据进行编码处理,得到适配于所述操作系统的目标编码文件。

12. 一种媒体数据处理装置,其特征在于,所述装置包括:

采集模块,用于在发生数据采集指令的情况下,根据与供虚幻引擎运行的操作系统相匹配的目标调用逻辑,调用媒体数据采集装置进行数据采集,得到真实场景数据;

读取模块,用于在发生数据读取指令的情况下,获取媒体数据文件;

录制模块,用于在发生数据录制指令的情况下,对基于虚幻引擎运行的虚拟交互应用中呈现的虚拟交互场景进行录制,得到虚拟场景数据;

确定模块,用于确定与目标业务对应的待处理媒体数据,所述待处理媒体数据包括如下至少一种:所述真实场景数据、所述媒体数据文件以及所述虚拟场景数据;

处理模块,用于对所述待处理媒体数据进行相应的业务处理,得到与所述目标业务对应的业务处理结果。

13. 一种计算机设备,包括存储器和处理器,所述存储器存储有计算机程序,其特征在于,所述处理器执行所述计算机程序时实现权利要求1至11中任一项所述的方法的步骤。

14. 一种计算机可读存储介质,其上存储有计算机程序,其特征在于,所述计算机程序被处理器执行时实现权利要求1至11中任一项所述的方法的步骤。

15. 一种计算机程序产品,包括计算机程序,其特征在于,该计算机程序被处理器执行时实现权利要求1至11中任一项所述的方法的步骤。

媒体数据处理方法、装置、计算机设备和存储介质

技术领域

[0001] 本申请涉及计算机处理技术领域,特别是涉及一种媒体数据处理方法、装置、计算机设备、存储介质和计算机程序产品。

背景技术

[0002] 随着游戏开发领域的技术发展,虚幻引擎的应用越来越普遍。为了实现不同的功能,游戏开发人员通常基于虚幻引擎的运行平台进行有针对性地开发。

[0003] 目前虚幻引擎中已有的各种插件都是针对于一种平台,这就导致插件的使用受限与操作系统,在另外的平台上可能无法正常使用,例如无法正常采集数据等。如果想要获取不同来源的数据,游戏开发人员必须有针对性的进行相应插件的开发和安装,从而实现插件与系统的兼容,并耗费相当大的人力物力进行维护,这就导致虚拟引擎插件的应用场景受限,无法满足多样化的业务需求。

发明内容

[0004] 基于此,有必要针对上述技术问题,提供一种媒体数据处理方法、装置、计算机设备、计算机可读存储介质和计算机程序产品。

[0005] 一方面,本申请提供了一种媒体数据处理方法。所述方法包括:

[0006] 在发生数据采集指令的情况下,根据与供虚幻引擎运行的操作系统相匹配的目标调用逻辑,调用媒体数据采集装置进行数据采集,得到真实场景数据;

[0007] 在发生数据读取指令的情况下,获取媒体数据文件;

[0008] 在发生数据录制指令的情况下,对基于虚幻引擎运行的虚拟交互应用中呈现的虚拟交互场景进行录制,得到虚拟场景数据;

[0009] 确定与目标业务对应的待处理媒体数据,所述待处理媒体数据包括如下至少一种:所述真实场景数据、所述媒体数据文件以及所述虚拟场景数据;

[0010] 对所述待处理媒体数据进行相应的业务处理,得到与所述目标业务对应的业务处理结果。

[0011] 另一方面,本申请还提供了一种媒体数据处理装置。所述装置包括:

[0012] 采集模块,用于在发生数据采集指令的情况下,根据与供虚幻引擎运行的操作系统相匹配的目标调用逻辑,调用媒体数据采集装置进行数据采集,得到真实场景数据;

[0013] 读取模块,用于在发生数据读取指令的情况下,获取媒体数据文件;

[0014] 录制模块,用于在发生数据录制指令的情况下,对基于虚幻引擎运行的虚拟交互应用中呈现的虚拟交互场景进行录制,得到虚拟场景数据;

[0015] 确定模块,用于确定与目标业务对应的待处理媒体数据,所述待处理媒体数据包括如下至少一种:所述真实场景数据、所述媒体数据文件以及所述虚拟场景数据;

[0016] 处理模块,用于对所述待处理媒体数据进行相应的业务处理,得到与所述目标业务对应的业务处理结果。

[0017] 另一方面,本申请还提供了一种计算机设备。所述计算机设备包括存储器和处理器,所述存储器存储有计算机程序,所述处理器执行所述计算机程序时实现上述媒体数据处理方法的步骤。

[0018] 另一方面,本申请还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,其上存储有计算机程序,所述计算机程序被处理器执行时实现上述媒体数据处理方法的步骤。

[0019] 另一方面,本申请还提供了一种计算机程序产品。所述计算机程序产品,包括计算机程序,该计算机程序被处理器执行时实现上述媒体数据处理方法的步骤。

[0020] 上述媒体数据处理方法、装置、计算机设备、存储介质和计算机程序产品,通过根据与操作系统相匹配的目标调用逻辑,调用媒体数据采集装置进行数据采集得到真实场景数据,可适配不同的操作系统与平台,无需针对于特定的平台开发插件,适用性更高;同时,响应于数据读取指令获取媒体数据文件、以及响应于数据录制指令,对基于虚幻引擎运行的虚拟交互应用中呈现的虚拟交互场景进行录制,得到虚拟场景数据,由此,通过获取不同来源方式的媒体数据,并从各种媒体数据中选择其中至少一种作为与目标业务对应的待处理媒体数据,并根据目标业务的业务需求对待处理媒体数据相应的业务处理,从而得到与目标业务对应的业务处理结果,能够按照实际的业务需求对多种来源的媒体数据进行灵活处理,能够满足多样化的业务需求,适用的场景更加广泛。

附图说明

[0021] 图1为一个实施例中媒体数据处理方法的应用环境图;

[0022] 图2为一个实施例中媒体数据处理方法的流程示意图;

[0023] 图2A为一个实施例中获取真实场景数据的示意图;

[0024] 图3为一个实施例中整体框架的示意图;

[0025] 图4A为一个实施例中检测业务的示意图;

[0026] 图4B为一个实施例中格式转换的示意图;

[0027] 图4C为一个实施例中检测业务的具体步骤的示意图;

[0028] 图5A为一个实施例中视频编码业务的示意图;

[0029] 图5B为一个实施例中视频编码业务的具体步骤的示意图;

[0030] 图5C为一个实施例中PBO原理的模块示意图;

[0031] 图5D为另一个实施例中PBO原理的模块示意图;

[0032] 图6为一个实施例中播放业务的示意图;

[0033] 图7为一个实施例中媒体数据处理方法的整体框架示意图;

[0034] 图8为一个实施例中媒体数据处理装置的结构框图;

[0035] 图9为一个实施例中计算机设备的内部结构图。

具体实施方式

[0036] 为了使本申请的目的、技术方案及优点更加清楚明白,以下结合附图及实施例,对本申请进行进一步详细说明。应当理解,此处描述的具体实施例仅仅用以解释本申请,并不用于限定本申请。

[0037] 本申请实施例提供的媒体数据处理方法,可以应用于如图1所示的应用环境中。其中,终端102通过网络与服务器104进行通信。数据存储系统可以存储服务器104需要处理的数据。数据存储系统可以集成在服务器104上,也可以放在云上或其他服务器上。其中,终端102通过插件调用媒体数据采集装置进行数据采集得到真实场景数据、通过插件获取媒体数据文件、以及通过插件捕获虚拟交互应用中呈现的虚拟交互场景得到虚拟场景数据,根据真实场景数据、媒体数据文件以及虚拟场景数据中的至少一种,终端确定与目标业务对应的待处理媒体数据,并对待处理媒体数据进行相应的业务处理,得到与目标业务对应的业务处理结果。其中,媒体数据文件可以是终端102通过服务器104下载传输得到的。

[0038] 其中,终端102可以但不限于各种台式计算机、笔记本电脑、智能手机、平板电脑、物联网设备和便携式可穿戴设备,物联网设备可为智能音箱、智能电视、智能空调、智能车载设备等。便携式可穿戴设备可为智能手表、智能手环、头戴设备等。

[0039] 其中,服务器104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。

[0040] 在一些实施例中,终端上装载有基于虚幻引擎开发的APP(Application,应用程序),终端在运行应用程序时驱动虚幻引擎实现画面展示和音频播放等。

[0041] 其中,虚幻引擎(Unreal Engine)是一款游戏引擎,包括多个版本,目前广泛使用的是4代版本,即UE4。游戏引擎是指一系列已经编写好的可编辑游戏系统或者交互式实时图像应用程序的核心组件,以供游戏开发人员利用可编辑游戏系统所提供的工具快速编写游戏程式。可编辑游戏系统例如包括渲染器、物理引擎、碰撞检测系统、音效、脚本引擎、电脑动画、以及网络引擎等。虚幻引擎可在不同的操作系统中运行。操作系统(Operating System)指的是终端中配置的管理计算机硬件与软件资源的计算机程序,包括但不限于Windows操作系统、macOS操作系统、Linux操作系统、iOS操作系统、Android操作系统、以及iPad OS操作系统等。

[0042] 虚幻引擎中预设有模块化系统、自定义插件以及源控制集等。游戏开发人员需要在虚幻引擎提供的代码框架的基础上进行开发。为了实现不同的功能,通常以插件的形式来提供能够应用的各项服务。其中,插件是游戏开发人员可在编辑器(Editor)中逐项目启用或禁用的代码和数据集合。插件可添加运行时修改内置引擎功能或添加功能、新建文件类型、以及使用新菜单、工具栏命令和子模式扩展编辑器的功能。

[0043] 由于虚幻引擎提供的代码框架可适用于不同的平台(包括手机、电脑、以及主机等),而不同的平台有自身的操作系统和运行逻辑,因此,诸如游戏开发人员等则会根据所使用的平台针对性地进行开发,这就导致目前的插件仅适用于一种操作系统。比如,虚幻引擎的Media Framework(媒体框架)插件只能适用于Windows系统,在macOS操作系统中无法获取系统权限以获取相机的使用权限。

[0044] 针对于跨平台的游戏项目,一方面,大部分的插件不支持二次开发,比如Arkit(一款可用于创建增强现实应用程序的插件)中提供了获取iOS操作系统的底层相机流的能力,但其不允许相关插件进行二次开发,并且不适用于其他操作系统。另一方面,则需要对这些插件的代码逻辑进行大量的修改,需要耗费极大的工作量和时间成本来进行插件兼容、插

件拼接等相关开发与测试。再者,在不同的平台上维护不同插件的组的代码,极易出错,可维护性差。

[0045] 因此,本申请实施例提供一种媒体数据处理方法以解决上述问题。本申请实施例提供的媒体数据处理方法可集成在一个音视频套件中,并通过终端运行该音视频套件实现对接多种平台并执行多样化的业务功能。

[0046] 在一些实施例中,音视频套件可以作为一个插件,上述媒体数据处理方法的步骤集成于该插件的代码中。在另一些实施例中,音视频套件可以包括多个插件,各个插件之间可进行相互之间的数据传输、插件复用等,并由各个插件共同执行上述媒体数据处理方法的步骤。

[0047] 在一个实施例中,如图2所示,提供了一种媒体数据处理方法,该方法可以应用于图1中的终端或服务器,也可以由终端和服务器协同执行。下面以该方法应用于图1中的终端为例进行说明,终端通过其所运行的音视频套件执行媒体数据处理方法,以实现跨平台的业务功能。具体地,包括以下步骤:

[0048] 步骤S202,在发生数据采集指令的情况下,根据与供虚幻引擎运行的操作系统相匹配的目标调用逻辑,调用媒体数据采集装置进行数据采集,得到真实场景数据。

[0049] 其中,数据采集指令用于指示终端中配置的媒体数据采集装置进行一种或多种媒体数据的数据采集。媒体数据采集装置指的是终端硬件提供的用于采集媒体数据的装置,包括但不限于麦克风、相机等中的一种或多种。媒体数据包括但不限于图像数据和音频数据等。媒体数据还可以为视频数据,视频数据可以由图像数据或音频数据中的一种构成,或者由图像数据和音频数据共同构成。

[0050] 其中,调用逻辑指的是为实现相应功能所配套的一系列的操作流程。比如,调用逻辑包括调用系统接口、通过系统接口申请系统权限、以及在获取系统权限后调用相应的硬件装置这一系列的流程。

[0051] 对于媒体数据的采集,终端可通过音视频套件的系统接口发起系统权限申请,从而获取相应的媒体数据采集装置的使用权限。

[0052] 本申请实施例中,音视频套件的系统接口中封装了各种操作系统的媒体数据采集装置的调用逻辑,由此可以适配各种操作系统,终端通过运行音视频套件即可唤起相应的硬件来采集本地的媒体数据。

[0053] 为了便于区分,将通过终端硬件提供的媒体数据采集装置所采集得到的媒体数据,称作真实场景数据。

[0054] 具体地,在发生数据采集指令的情况下,终端响应于数据采集指令,确定当前运行的操作系统,并根据该操作系统,从音视频套件中封装的多套分别与不同的操作系统对应的调用逻辑中,选择与当前运行的操作系统相匹配的目标调用逻辑,并通过音视频套件的系统接口发起调用,由该音视频套件执行该目标调用逻辑,从而调用终端硬件提供的媒体数据采集装置进行数据采集,得到真实场景数据。

[0055] 示例性地,数据采集指令指示采集图像数据和音频数据,则终端响应于数据采集指令,确定当前运行的操作系统为Windows操作系统,并通过音视频套件的系统接口发起系统权限获取,该音视频套件执行与Windows操作系统相匹配的目标调用逻辑,从而获取Windows操作系统的系统权限,从而调用相机和麦克风进行数据采集。

[0056] 如图2A所示,终端通过音视频套件唤起请求,调用相机和麦克风获取相机画面和音频数据,得到真实场景数据,该真实场景数据用于后续加载至音视频套件中以完成目标业务的处理。

[0057] 由此,通过音视频套件提供适配于各种操作系统的调用逻辑,并且将多套调用逻辑封装在系统接口中,实现了在运行音视频套件时对底层的操作流程无感知,并且只需调用系统接口即可完成权限获取,操作便携并降低了使用门槛。

[0058] 步骤S204,在发生数据读取指令的情况下,获取媒体数据文件。

[0059] 其中,数据读取指令用于指示终端读取媒体数据文件。媒体数据文件是以特定的文件格式封装的媒体数据。封装媒体数据的过程通常为将视频压缩数据和音频压缩数据打包并合并成一个文件的过程。文件格式包括但不限于AVI(Audio Video Interleave,音频视频交错)格式、WMV(Windows Media Video,微软流媒体)格式、以及MPEG(Moving Picture Experts Group,动态图像专家组)格式等中的一种或多种。

[0060] 具体地,在发生数据读取指令的情况下,终端响应于数据读取指令,从本地存储空间中读取预先存储的媒体数据文件,或读取实时接收的由其他终端或服务器传输的媒体数据文件。比如,媒体数据文件可以由终端预先下载并存储至本地存储空间中,又如,终端响应于数据读取指令,与其他终端或服务器建立通信连接,并读取其他终端或服务器传输的媒体数据文件。

[0061] 步骤S206,在发生数据录制指令的情况下,对基于虚幻引擎运行的虚拟交互应用中呈现的虚拟交互场景进行录制,得到虚拟场景数据。

[0062] 其中,数据录制指令用于指示终端对当前运行的虚拟交互应用的画面和声音进行录制。虚拟交互应用指的是基于虚幻引擎运行的应用程序,例如游戏程序等。虚拟交互应用在运行过程中会呈现虚拟交互场景,该虚拟交互场景由画面和声音构成,用于供操作者观看剧情、控制角色进行各种动作等。

[0063] 对虚拟交互场景录制得到的数据包括图像数据和音频数据,为了与终端通过调用媒体数据采集装置所采集的真实场景数据进行区分,将虚拟交互应用提供的数据称为虚拟场景数据。

[0064] 具体地,在发生数据录制指令的情况下,终端响应于数据录制指令,对操作系统中运行的虚拟交互应用中呈现的虚拟交互场景进行录制,得到虚拟场景数据。比如,终端在运行虚拟交互应用时,通过显示屏显示画面并通过扬声器播放声音,则终端通过采集显示屏展示的图像和扬声器播放的音频以得到虚拟场景数据。

[0065] 示例性地,终端可以通过调用媒体采集装置进行录制,从而得到虚拟场景数据。或者,终端还可以通过调用音视频套件以实现录制。在一些实施例中,终端可以通过虚幻引擎所提供的Media Framework插件获取虚拟场景数据。

[0066] 步骤S208,确定与目标业务对应的待处理媒体数据,待处理媒体数据包括真实场景数据、媒体数据文件以及虚拟场景数据中的至少一种。

[0067] 其中,根据业务需求的不同,需要对不同来源的媒体数据进行处理,从而实现相应的业务功能。因此,具体地,在对媒体数据进行具体处理之前,终端确定当前的目标业务,并确定目标业务所对应的待处理媒体数据。待处理媒体数据可以为真实场景数据、媒体数据文件以及虚拟场景数据中的单种媒体数据,也可以为其中两种媒体数据的组合,或者也可

以为三种媒体数据的组合,等。

[0068] 例如,当需要执行游戏场景录制的功能时,待处理媒体数据至少包括虚拟场景数据,在此基础上根据实际需求还可以包括虚拟场景数据和真实场景数据,亦可以包括虚拟场景数据、真实场景数据和媒体数据文件。又如,当需要执行游戏内播放音视频的功能时,待处理媒体数据至少包括媒体数据文件,在此基础上根据实际需求还可以包括另外两种来源的媒体数据中的至少一种。再如,当需要执行数据定制化改造的功能时,所需的待处理媒体数据至少包括真实场景数据,在此基础上根据实际需求还可以包括真实场景数据和媒体数据文件,亦可以包括全部三种来源的媒体数据,等。

[0069] 步骤S210,对待处理媒体数据进行相应的业务处理,得到与目标业务对应的业务处理结果。

[0070] 其中,根据目标业务的不同,对待处理媒体数据进行具体业务处理的处理流程也不同。具体地,终端根据所确定的目标业务,按照与该目标业务相匹配的、预先设置的处理流程,运行音视频套件对待处理媒体数据进行相应的业务处理,从而得到与目标业务对应的业务处理结果。仍以上述举例进行说明,例如,对于游戏场景录制的目标业务,终端对待处理媒体数据进行业务处理得到的业务处理结果即为编码后的录制文件。

[0071] 需要说明的是,业务处理结果为目标业务的最终处理结果,该最终处理结果可以作为其他目标业务的输入。例如上述示例中的录制文件,后续终端对其进行存储、播放、以及传输等。

[0072] 举例而言,如图3所示,终端将所获取的待处理媒体数据加载至音视频套件中,并根据目标业务的不同,通过运行音视频套件完成对待处理媒体数据的业务处理。例如,对于用于算法检测的目标业务,终端通过运行音视频套件对待处理媒体数据进行格式转换,并对格式转换后的数据进行相应的算法检测,例如进行图像检测和语音检测等,从而得到算法检测结果作为业务处理结果,该业务处理结果可用于后续的其他目标业务并作为输入,例如作为神经网络的训练样本输入至待训练的神经网络中,等。

[0073] 又如,对于游戏内播放音视频的目标业务,终端获取包括媒体数据文件的待处理媒体数据,并通过运行音视频套件将该待处理媒体数据转换为虚幻引擎支持的数据格式,并结合虚幻引擎提供的可视化控件,使得该待处理媒体数据在终端运行的虚拟交互应用中播放,例如将待处理媒体数据的图像数据展示在虚拟交互应用提供的虚拟交互场景中,等。

[0074] 再如,对于游戏场景录制的目标业务,终端在获取包括虚拟场景数据和媒体数据文件的待处理媒体数据之后,通过运行音视频套件对该待处理媒体数据进行业务处理,获取编码状态并对该待处理媒体数据中的视频数据和音频数据进行编码,从而得到编码后的录制文件。

[0075] 上述媒体数据处理方法、装置、计算机设备、存储介质和计算机程序产品,通过根据与操作系统相匹配的目标调用逻辑,调用媒体数据采集装置进行数据采集得到真实场景数据,可适配不同的操作系统与平台,无需针对于特定的平台开发插件,适用性更高;同时,响应于数据读取指令获取媒体数据文件、以及响应于数据录制指令,对基于虚幻引擎运行的虚拟交互应用中呈现的虚拟交互场景进行录制,得到虚拟场景数据,由此,通过获取不同来源方式的媒体数据,并从各种媒体数据中选择其中至少一种作为与目标业务对应的待处理媒体数据,并根据目标业务的业务需求对待处理媒体数据相应的业务处理,从而得到与

目标业务对应的业务处理结果,能够按照实际的业务需求对多种来源的媒体数据进行灵活处理,能够满足多样化的业务需求,适用的场景更加广泛。

[0076] 音视频套件中可设置有多个接口,用于实现不同的功能,例如通过接口实现数据传输等。为了实现方便、快捷地对接各种不同的平台并适配各种不同的操作系统,音视频套件设置有数据采集通用接口,从而通过调用数据采集通用接口快速唤起终端硬件进行数据采集。为此,在一些实施例中,根据与供虚幻引擎运行的操作系统相匹配的目标调用逻辑,调用媒体数据采集装置进行数据采集,得到真实场景数据,包括:确定数据采集通用接口,数据采集通用接口中封装有与多个不同操作系统分别对应的媒体数据采集装置相适配的调用逻辑;确定数据采集通用接口中封装的、与操作系统的系统类型相匹配的目标调用逻辑;基于目标调用逻辑,通过所述数据采集通用接口调用媒体数据采集装置进行数据采集,得到真实场景数据。

[0077] 具体地,音视频套件中提供的数据采集通用接口中,封装有所支持的各种操作系统的调用逻辑,该调用逻辑用于被执行时调用终端硬件,例如唤起相机或麦克风等。终端通过调用音视频套件的数据采集通用接口,根据当前的操作系统从多套调用逻辑中,选择与该操作系统相匹配的一套调用逻辑作为目标调用逻辑。终端运行的音视频套件执行该目标调用逻辑所指示的操作流程,通过该数据采集通用接口调用媒体数据采集装置进行数据采集,从而得到真实场景数据。

[0078] 上述实施例中,通过在数据采集通用接口中封装多套调用逻辑,在运行时再根据当前配置的操作系统确定具体所要使用的目标调用逻辑,由此能够对接不同的平台并适配各种操作系统,具有广泛的适用性。

[0079] 对于数据采集的任务,主要在于获取终端硬件的调用权限。在一些实施例中,基于目标调用逻辑,调用媒体数据采集装置进行数据采集,得到真实场景数据,包括:基于目标调用逻辑,向操作系统发起权限获取请求;在根据权限获取请求获得调用授权的情况下,调用媒体数据采集装置进行数据采集;接收媒体数据采集装置采集的真实场景数据。

[0080] 具体地,终端通过音视频套件执行目标调用逻辑,由音视频套件向当前的操作系统发起权限获取请求,该权限获取请求用于获取终端配置的媒体数据采集装置的调用权限。在操作系统进行授权后,音视频套件获得调用权限,并调用媒体数据采集装置进行数据采集。

[0081] 示例性地,终端通过音视频套件向操作系统请求获取麦克风的使用权限,并在获取授权后调用麦克风采集音频数据。或者,终端通过音视频套件向操作系统请求获取相机的使用权限,并在获取授权后调用相机采集图像数据或视频数据。

[0082] 上述实施例中,通过在数据采集通用接口中封装多套调用逻辑,在运行时再根据当前配置的操作系统确定具体所要使用的目标调用逻辑,由此能够对接不同的平台并适配各种操作系统,具有广泛的适用性。同时,在调用媒体数据采集装置之前需获取系统权限,具有一定的安全性。

[0083] 为了提高整体的处理效率,优化音视频套件的性能,在一些实施例中,在音视频套件的数据加载、以及音视频套件针对不同的目标业务进行业务处理(例如数据定制化处理、解码、以及编码等)等的过程中,还进行了加速处理,以进一步提高音视频套件的业务处理效率。

[0084] 再如图3所示,举例而言,终端将待处理媒体数据加载至音视频套件中时,可以对数据加载的过程进行加速。为此,除了上述实施例中提及的用于适配不同平台的数据采集通用接口以外,音视频套件中还设置有加速接口,该加速接口用于在数据加载的过程中提高速度,以加快数据传输。

[0085] 相应地,在一些实施例中,对待处理媒体数据进行相应的业务处理,得到与目标业务对应的业务处理结果之前,上述方法还包括:确定与操作系统相匹配的加速接口;通过与操作系统相匹配的加速接口对待处理媒体数据进行相应的解码处理,基于解码后的待处理媒体数据执行对待处理媒体数据进行相应的业务处理的步骤。

[0086] 其中,加速接口预先设置有适配于不同操作系统的接口使用规范,使得音视频套件在接收到待处理媒体数据后,无需再进行进一步地识别或转换等处理,从而加快数据加载的速度。接口使用规范例如包括操作系统所使用的接口参数、数据格式、以及通信协议等等。

[0087] 具体地,终端通过调用音视频套件的加速接口,将待处理媒体数据加载至音视频套件中,并由该音视频套件对该待处理媒体数据进行解码,从而基于解码后的待处理媒体数据,执行相应的业务处理得到业务处理结果。由此,通过对将待处理媒体数据加载至音视频套件的过程进行加加速处理,从而实现数据传输的效率提升,进而提高了目标业务的处理效率。

[0088] 为了更好地理解本发明的技术构思,下面在多个不同的应用场景下阐述本申请实施例提供的媒体数据的处理方法。

[0089] 以图4A为例,在一些实施例中,目标业务包括检测业务,该检测业务用于对加载至音视频套件中的待处理媒体数据进行算法检测。该检测业务中,终端通过运行音视频套件对待处理媒体数据进行算法检测,例如进行图像检测和语音检测等,从而得到算法检测结果作为业务处理结果,该业务处理结果可用于后续的其他目标业务并作为输入,例如作为神经网络的训练样本输入至待训练的神经网络中,等。换言之,与该目标业务对应的待处理媒体数据,可以包括真实场景数据和媒体数据文件中的至少一种。

[0090] 相应地,对待处理媒体数据进行相应的业务处理,得到与目标业务对应的业务处理结果,包括:确定适配于目标检测算法的目标格式;将待处理媒体数据的数据格式转换成目标格式;基于目标检测算法对待处理媒体数据中的待处理视频数据进行图像检测,得到视频检测结果,对待处理媒体数据中的待处理音频数据进行音频检测,得到音频检测结果;基于视频检测结果和音频检测结果,确定与目标业务对应的业务处理结果。

[0091] 对于真实场景数据,不同的操作系统通过相机获取图像数据的编码方式不统一:有的操作系统使用YUV(一种颜色编码方法)格式作为编码方式,而有的平台用RGB(一种色彩模式)格式作为编码方式。而作为检测算法的输入的数据格式往往是统一的。因此,需要将输入的媒体数据的格式,转换成检测算法所支持的格式,即目标格式。

[0092] 具体地,根据当前的检测业务,终端对加载至音视频套件的待处理媒体数据进行数据预处理,先将待处理媒体数据的数据格式转换成适配于目标检测算法的目标格式。例如,如图4B所示,终端对待处理媒体数据进行格式转换,从而将待处理媒体数据从原始格式转换至适配于目标检测算法的目标格式。

[0093] 在格式转换后,终端通过音视频套件运行目标检测算法,对该待处理媒体数据中

的待处理视频数据进行图像检测,得到视频检测结果,视频检测结果包括但不限于目标对象图像、背景图像等检测结果。其中,根据算法用途的不同,图像检测的相关检测算法包括但不限于图像识别算法、目标检测算法、阈值分割算法、以及动作检测算法等中的一种或多种。同时,对待处理媒体数据中的待处理音频数据进行音频检测,得到音频检测结果,视频检测结果包括但不限于所提取的音频数据中的人声、环境声、以及音乐声等。其中,根据算法用途的不同,音频检测的相关检测算法包括但不限于语音情感识别算法、音频节奏检测算法、语义检测算法、以及噪声检测算法等中的一种或多种。根据视频检测结果和音频检测结果,终端即可确定与该检测业务对应的业务处理结果。

[0094] 示例性地,如图4C所示,以待处理媒体数据包括真实场景数据和媒体数据文件为例,终端分别对真实场景数据和媒体数据文件中的音视频进行分离,得到真实场景数据的原生音频数据和原生图像数据,以及得到媒体数据文件中的文件视频数据和文件音频数据。

[0095] 在该检测业务中,音视频套件对数据进行预处理,预处理包括对数据进行格式转换,以使得输入至检测算法中的数据的格式符合检测算法所规定的格式。具体地,音视频套件根据算法协议所规定的目标格式,对输入的数据进行格式转换,得到视频算法数据和音频算法数据。再由图像检测算法对视频算法数据进行算法检测,得到视频检测结果;由语音检测算法对音频算法数据进行音频检测,得到音频检测结果;由此得到算法检测结果。该算法检测结果可用于后续的其他目标业务。

[0096] 上述实施例中,通过将原生音视频输入转换成与音视频算法协议匹配的数据,之后作为算法输入进行算法检测,可供后续进行进一步地处理,实现对数据的定制化改造。同时,根据业务需求可以采用不同的算法进行检测,算法可以即时替换和更新,使用更加灵活,能够应对不同的业务需求。

[0097] 为了进一步提高处理效率,再如图4B所示,在进行数据格式转换的过程中也进行了加速处理。为此,在一些实施例中,将待处理媒体数据的数据格式转换成目标格式,包括:启动子线程,通过子线程确定待处理媒体数据的原始格式,并通过逻辑转换函数,将待处理媒体数据从原始格式转换至目标格式;在子线程执行完转换操作后,通知主线程已完成格式转换。

[0098] 具体地,终端创建子线程,并启动子线程以执行格式转换的过程。子线程首先访问待处理媒体数据来确定待处理媒体数据的原始格式,再通过加载逻辑转换函数,从而将该待处理媒体数据从原始格式转换至目标格式。比如,对于图像数据而言,子线程访问待处理媒体数据并逐个像素进行读取,并通过逻辑转换函数将待处理媒体数据从原始格式转换至目标格式,由此输出目标格式的像素。

[0099] 示例性地,可以通过NEON指令对转换过程进行加速,在触发格式转换时启动子线程,以解决格式转换时阻塞主线程的问题。其中,NEON指令是适用于ARM Cortex-A系列处理器的一种128位SIMD(Single Instruction, Multiple Data,单指令、多数据)扩展指令集。

[0100] 在子线程执行完转换操作后,子线程向主线程发送消息,用于提示主线程已完成格式转换,从而由主线程结束格式转换的流程。而在子线程执行格式转换的过程中,主线程可以同时执行其他逻辑,以提高效率。

[0101] 上述实施例中,通过启动子线程,由子线程独立执行格式转换,子线程与主线程互

不干扰,有效地避免了媒体数据在格式转换的过程中阻塞主线程的问题,提升了格式转换的速度,进一步提高了处理效率。

[0102] 除了通过创建子线程进行格式转换以进行加速的方式以外,在另一些实施例中,将待处理媒体数据的数据格式转换成目标格式,包括:调用与操作系统相匹配的绘制函数,将待处理媒体数据输入至图形处理器中;通过图形处理器进行批量的格式转换处理,以将待处理媒体数据的数据格式转换成目标格式;将图形处理器中属于目标格式的待处理媒体数据回传至中央处理器。

[0103] 换言之,在本实施例中,终端通过图形处理器(Graphics Processing Unit,GPU)执行格式转换。具体地,终端调用与当前的操作系统相匹配的图形接口(API)实现对绘制函数的调用,将待处理媒体数据输入至图形处理器中,由该图形处理器进行批量的格式转换处理,以将待处理媒体数据的数据格式转换成目标格式,在转换完成后,图形处理器再将转换好的且属于目标格式的待处理媒体数据回传至中央处理器(Central Processing Unit,CPU)。其中,图形处理器进行批量的格式转换处理的过程,可以通过运行Shader(着色器)程序实现。

[0104] 比如,终端调用GPU中的图形接口来调用绘制函数,并通过调用绘制函数将待处理媒体数据输入至GPU中,在GPU中通过Shader程序使得待处理媒体数据在GPU中进行批量的格式转换处理,以将待处理媒体数据的数据格式转换成目标格式,在转换完成后,CPU读取GPU纹理以实现由GPU转换完成的数据的回传。

[0105] 上述实施例中,通过图形处理进行格式转换,能够大批量并行化计算,计算速度远远高于中央处理器的处理速度,并且转换过程中不占用中央处理器的计算资源,并极大提高格式转换的效率。

[0106] 承前所述,经过目标业务处理得到的业务处理结果可用于进一步的处理。在一些应用场景中,业务处理结果可以包括驱动相关数据,驱动相关数据指的是用于指示虚拟交互场景中的虚拟角色的肢体驱动数据和表情驱动数据等。例如,驱动相关数据为人体骨骼和面部表情的三维重建结果,该三维重建结果可以结合所采集的真实场景数据中的图像数据,在三维展示的场景中进行表情驱动和肢体动作等。为此,在该应用场景中,上述方法还包括:根据驱动相关数据,控制虚拟交互场景中的虚拟角色,按照待处理媒体数据中对象的动作进行运动。

[0107] 具体地,终端根据对待处理媒体数据进行业务处理得到的驱动相关数据,将该驱动相关数据加载至虚拟交互应用中,以在虚拟交互应用中,根据该驱动相关数据控制其所展示的虚拟交互场景中的虚拟角色,并控制该虚拟角色按照待处理媒体数据中对象的动作进行运动。比如,终端根据驱动相关数据,控制虚拟交互场景中的虚拟角色被驱动的肢体部位按照所采集的真实场景数据中的对象的动作而运动,或者,控制虚拟角色的面部表情按照所采集的真实场景数据中的对象的面部表情而变化。

[0108] 由此,通过对待处理媒体数据进行算法检测,并将经算法检测得到的业务处理结果进一步用于对虚拟角色的控制,提供了虚拟交互应用与真实对象之间的交互的功能,无需使用额外的插件实现,同时并能对接不同的平台,具有良好的适配性。

[0109] 以图5A为例,在一些实施例中,目标业务包括视频编码业务,该视频编码业务用于将至少包括虚拟场景数据的媒体数据进行编码。该视频编码业务中,可以将虚拟交互应用

(例如游戏)内的画面、声音等虚拟场景数据加载至音视频套件中,由音视频套件进行编码得到目标编码文件。或者,也可以将虚拟场景数据与终端通过调用媒体数据采集装置捕获的原生的相机、麦克风等真实场景数据一起加载至音视频套件中并进行编码,得到目标编码文件。或者,也可以对虚拟场景数据、真实场景数据以及媒体数据文件一起进行编码,得到对应的目标编码文件。

[0110] 换言之,与目标业务对应的待处理媒体数据至少包括虚拟场景数据,在此基础上还可以包括真实场景数据和媒体数据文件中的一种或多种。

[0111] 相应地,对待处理媒体数据进行相应的业务处理,得到与目标业务对应的业务处理结果,包括:获取预设的编码状态;启动子线程,通过子线程并按照编码状态,对待处理媒体数据进行编码处理,得到适配于操作系统的目标编码文件。

[0112] 其中,编码状态是与编码相关的参数信息,包括但不限于编码时间、编码格式、码率、以及分辨率等参数信息。本申请实施例提供的音视频套件可提供用于配置参数的接口,终端可通过调用该接口来灵活地设置编码状态。

[0113] 具体地,终端可以向操作者提供可视化的编辑界面,该编辑界面可供操作者设置编码状态。由此,终端根据该编辑界面的输入即可获取所设置的编码状态。进而,终端按照该编码状态对待处理媒体数据进行相应的编码处理,从而得到适配于本地操作系统的目标编码文件。

[0114] 终端在对待处理媒体数据进行编码的过程中,可基于本地操作系统适配的文件格式(比如AVI或mp4等格式),将待处理媒体数据编码成本地的操作系统支持的各种格式的目标编码文件。

[0115] 在一些实施例中,终端在进行编码处理前,可启动子线程,并通过子线程执行编码步骤,这样不影响主线程的进程,可以提高处理效率。

[0116] 上述实施例中,通过预先设置的编码状态,并在子线程下对待处理媒体数据进行相应的编码处理,可以在虚幻引擎中实现多种类型数据的快速编码。

[0117] 示例性地,如图5B所示,以待处理媒体数据包括真实场景数据和虚拟场景数据为例,终端分别对真实场景数据和虚拟场景数据的音视频进行分离,得到真实场景数据的原生音频数据和原生图像数据,以及虚拟场景数据的游戏屏幕数据和游戏音频数据。

[0118] 在该视频编码业务中,音视频套件在编码数据采集的过程中进行了加速。通常而言,在采集游戏运行时的屏幕数据需要将GPU中经渲染得到的屏幕数据传回CPU,由CPU进行编码数据的采集操作。但GPU向CPU传输过程的性能严重受限于显卡带宽。若调用系统接口读取GPU中的图像数据,比如调用OpenGL(开放式图形库)上采取调用ReadPixel(读取函数)接口读取GPU中的图像数据。但是这种方式会阻塞主线程,直至将完整的渲染数据读取到CPU中之后才会解除主线程的阻塞状态,极度耗时。

[0119] 为此,为了提高处理效率,在一些实施例中,对基于虚幻引擎运行的虚拟交互应用中呈现的虚拟交互场景进行录制,得到虚拟场景数据,包括:通过图形处理器,对基于虚幻引擎运行的虚拟交互应用中的虚拟交互场景进行渲染,得到图像数据;通过图形处理器将图像数据加载至预设缓存区中;通过中央处理器异步读取预设缓存区中的图像数据,基于中央处理器读取的图像数据以实现采集操作,得到虚拟场景数据。

[0120] 具体地,在虚拟交互应用运行的过程中,终端通过图形处理器,对基于虚幻引擎运

行的虚拟交互应用中的虚拟交互场景进行渲染,得到图像数据。对于渲染得到的图像数据,终端通过图形处理器将图像数据加载至预设缓存区中。终端通过中央处理器异步读取预设缓存区中的图像数据,从而完成采集操作并得到虚拟场景数据。

[0121] 示例性地,预设缓存区例如为基于PBO (Pixel Buffer Object,像素缓冲对象)技术设置的高速存储区。如图5C所示,可以通过glReadPixels (读取函数)函数从OpenGL的帧缓冲区读取像素数据,并将数据写入至PBO中,这一过程可以视作打包的过程。然后,可以通过glDrawPixels (绘制函数)函数从PBO中读取像素数据并将其复制到OpenGL的帧缓冲区,这一过程可以视作解压缩的过程。对于纹理对象也同理。由此,通过利用PBO实现异步的DMA (直接存储器访问)传输,能够实现像素数据的快速上传和下载,效率更高且不会影响CPU周期。如图5D所示,GPU通过DMA传输直接将图像纹理等数据加载到PBO中,CPU可以异步读取PBO中数据并进行数据采集,从而得到图像数据,由于前一步是由GPU执行,因此不会占用CPU运行周期。

[0122] 上述实施例中,通过异步传输能够实现快速的传输所渲染的图像数据,节省了图像采集的时间,提高了处理效率。

[0123] 以图6为例,在一些实施例中,目标业务还包括播放业务。该播放业务用于结合虚幻引擎中的UI (User Interface,用户界面)控件,在对加载至音视频套件中的待处理媒体数据进行数据预处理之后,得到虚幻引擎所支持的数据格式的目标视频数据,并将该目标视频数据在运行的虚拟交互应用中进行播放。

[0124] 该播放业务中,可以将读取的媒体数据文件在虚拟交互应用中进行播放(例如播放音乐、动画等)。或者,也可以将采集得到的真实场景数据在虚拟交互应用中进行播放。

[0125] 换言之,与目标业务对应的待处理媒体数据至少包括虚拟场景数据,在此基础上还可以包括真实场景数据和媒体数据文件中的一种或多种。

[0126] 相应地,对待处理媒体数据进行相应的业务处理,得到与目标业务对应的业务处理结果,包括:对真实场景数据和媒体数据文件中的至少一种进行格式转换,得到适应于虚幻引擎的目标视频数据;在通过虚拟交互应用播放虚拟交互场景的过程中,叠加目标视频数据进行播放。

[0127] 具体地,终端可对真实场景数据和媒体数据文件中的至少一种数据进行格式转换,得到适应于虚幻引擎的目标视频数据。也就是说,终端可基于实际的业务需求,只对真实场景数据进行格式转换,得到适用于虚幻引擎的目标视频数据,或者,只对媒体数据文件进行格式转换,得到适用于虚幻引擎的目标视频数据,或者,对真实场景数据和媒体数据文件均进行格式转换,得到适用于虚幻引擎的目标视频数据。

[0128] 进而,在通过虚拟交互应用播放与虚拟场景数据对应的虚拟交互场景的过程中,终端可叠加目标视频数据进行播放。也就是在虚拟交互场景中叠加目标视频数据进行播放。以虚拟交互场景为游戏场景示例,本申请可以实现在游戏场景中叠加目标视频数据进行播放。

[0129] 在一些实施例中,终端可对各种来源的媒体数据进行格式转换,以转换成虚幻引擎支持的统一数据格式,进而通过交互控件结合虚幻引擎中的Media Framework进行播放。本申请实施例中提及的音视频套件打通了虚幻引擎中原生UI控件,使得目标视频数据可以直接显示在UI控件的画布上。

[0130] 上述实施例中,通过对真实场景数据和媒体数据文件中的至少一种进行格式转换,得到适应于虚幻引擎的目标视频数据,进而可在虚拟交互场景中叠加目标视频进行播放,拓宽了视频的呈现方式和内容,大大增强了互动性和交互性,还可以将现实场景叠加虚拟场景进行融合显示,大大提高了交互性。

[0131] 为了进一步提高处理速度,本申请实施例在数据预处理的过程中也进行了加速处理。该加速处理的步骤可以参照前述实施例的描述,此处不再赘述。

[0132] 本申请还提供一种应用场景,该应用场景应用上述的媒体数据处理方法。具体地,该媒体数据处理方法在该应用场景的应用例如如下:终端通过插件获取不同来源的媒体数据,例如通过插件调用电脑或手机的麦克风和相机采集真实场景数据,又如通过网络下载获取媒体数据文件,再如通过录屏软件录制游戏得到虚拟场景数据等。根据实际的业务需求,终端通过插件中预设的各种业务处理流程,对一种或多种媒体数据进行业务处理,从而得到业务处理结果,该业务处理结果可用于后续的处理并作为输入,例如将获取的媒体数据文件在游戏内进行播放、或者根据真实场景数据驱动游戏内的角色进行相应的动作,等等。

[0133] 当然并不局限于此,本申请提供的媒体数据处理方法还可以应用在其他应用场景中,例如游戏直播、VR(Virtual Reality,虚拟现实)视频或游戏、以及虚拟形象绘制等等。

[0134] 在一个具体的实施例中,终端确定当前的目标业务,并根据目标业务的业务需求获取不同来源的媒体数据。当目标业务需要使用到真实场景数据时,终端根据该目标业务生成数据采集指令,并根据该数据采集指令运行音视频套件,由该音视频套件根据与供虚幻引擎运行的操作系统相匹配的目标调用逻辑,调用媒体数据采集装置进行数据采集,得到真实场景数据。当目标业务需要使用到媒体数据文件时,终端根据该目标业务生成数据读取指令,并根据该数据读取指令读取媒体数据文件。当目标业务需要使用到虚拟场景数据时,终端根据该目标业务生成数据录制指令,并根据该数据录制指令运行音视频套件或其他插件,对基于虚幻引擎运行的虚拟交互应用中呈现的虚拟交互场景进行录制,得到虚拟场景数据。当然并不局限于此,目标业务也可以使用两种或两种以上不同来源的媒体数据。根据目标业务,终端确定与目标业务对应的待处理媒体数据,并对所述待处理媒体数据进行相应的业务处理,得到与所述目标业务对应的业务处理结果。该业务处理例如包括检测业务、视频编码业务、以及播放业务等。

[0135] 以下结合实际应用场景以一个具体的示例进行说明。以虚幻引擎4代(UE4)为例,本申请实施例提供的媒体数据处理方法的整体应用框架可如图7所示,对于不同来源的媒体数据(包括真实场景数据、媒体数据文件、以及虚拟场景数据),分别得到相应的音频数据和图像/视频/屏幕数据,并基于这些数据分别进行不同的目标业务的处理,例如上述实施例中的检测业务、视频编码业务、播放业务等等。并且在数据加载、编解码等相关操作的过程中,也进行了深度优化和加速,提升了处理效率,性能更加卓越。

[0136] 由此,本申请实施例提供了针对于虚拟引擎的跨平台的媒体数据处理方法,解决目前只能组装使用各个独立平台各自的插件所带来的高开发成本、高维护成本、高稳定性风险的问题。同时,音视频套件能够基于实际的业务需求进行相应地业务处理,具有极高的拓展性,并且便于行二次开发。此外,对于数据加载及目标业务的处理过程均进行了深度的性能加速,提升了处理效率。

[0137] 应该理解的是,虽然如上的各实施例所涉及的流程图中的各个步骤按照箭头的指示依次显示,但是这些步骤并不是必然按照箭头指示的顺序依次执行。除非本文中有明确的说明,这些步骤的执行并没有严格的顺序限制,这些步骤可以以其它的顺序执行。而且,如上的各实施例所涉及的流程图中的至少一部分步骤可以包括多个步骤或者多个阶段,这些步骤或者阶段并不必然是在同一时刻执行完成,而是可以在不同的时刻执行,这些步骤或者阶段的执行顺序也不必然是依次进行,而是可以与其它步骤或者其它步骤中的步骤或者阶段的至少一部分轮流或者交替地执行。

[0138] 基于同样的发明构思,本申请实施例还提供了一种用于实现上述所涉及的媒体数据处理方法的媒体数据处理装置。该装置所提供的解决问题的实现方案与上述方法中所记载的实现方案相似,故下面所提供的一个或多个媒体数据处理装置实施例中的具体限定可以参见上文中对于媒体数据处理方法的限定,在此不再赘述。

[0139] 在一个实施例中,如图8所示,提供了一种媒体数据处理装置800,包括:采集模块801、读取模块802、录制模块803、确定模块804和处理模块805,其中:

[0140] 采集模块801,用于在发生数据采集指令的情况下,根据与供虚幻引擎运行的操作系统相匹配的目标调用逻辑,调用媒体数据采集装置进行数据采集,得到真实场景数据。

[0141] 读取模块802,用于在发生数据读取指令的情况下,获取媒体数据文件。

[0142] 录制模块803,用于在发生数据录制指令的情况下,对基于虚幻引擎运行的虚拟交互应用中呈现的虚拟交互场景进行录制,得到虚拟场景数据。

[0143] 确定模块804,用于确定与目标业务对应的待处理媒体数据,待处理媒体数据包括如下至少一种:真实场景数据、媒体数据文件以及虚拟场景数据。

[0144] 处理模块805,用于对待处理媒体数据进行相应的业务处理,得到与目标业务对应的业务处理结果。

[0145] 在一些实施例中,读取模块还用于确定数据采集通用接口,数据采集通用接口中封装有与多个不同操作系统分别对应的媒体数据采集装置相适配的调用逻辑;确定数据采集通用接口中封装的、与操作系统的系统类型相匹配的目标调用逻辑;基于目标调用逻辑,通过数据采集通用接口调用媒体数据采集装置进行数据采集,得到真实场景数据。

[0146] 在一些实施例中,读取模块还用于基于目标调用逻辑,向操作系统发起权限获取请求;在根据权限获取请求获得调用授权的情况下,调用媒体数据采集装置进行数据采集;接收媒体数据采集装置采集的真实场景数据。

[0147] 在一些实施例中,录制模块还用于通过图形处理器,对基于虚幻引擎运行的虚拟交互应用中的虚拟交互场景进行渲染,得到图像数据;通过图形处理器将图像数据加载至预设缓存区中;通过中央处理器异步读取预设缓存区中的图像数据,基于中央处理器读取的图像数据以实现采集操作,得到虚拟场景数据。

[0148] 在一些实施例中,上述装置还包括加速模块,用于确定与操作系统相匹配的加速接口;通过与操作系统相匹配的加速接口对待处理媒体数据进行相应的解码处理,基于解码后的待处理媒体数据执行对待处理媒体数据进行相应的业务处理的步骤。

[0149] 在一些实施例中,目标业务包括检测业务,与目标业务对应的待处理媒体数据包括真实场景数据和媒体数据文件中的至少一种;处理模块还用于确定适配于目标检测算法的目标格式;将待处理媒体数据的数据格式转换成目标格式;基于目标检测算法对待处理

媒体数据中的待处理视频数据进行图像检测,得到视频检测结果,对待处理媒体数据中的待处理音频数据进行音频检测,得到音频检测结果;基于视频检测结果和音频检测结果,确定与目标业务对应的业务处理结果。

[0150] 在一些实施例中,处理模块还用于启动子线程,通过子线程确定待处理媒体数据的原始格式,并通过逻辑转换函数,将待处理媒体数据从原始格式转换至目标格式;在子线程执行完转换操作后,通知主线程已完成格式转换。

[0151] 在一些实施例中,处理模块还用于调用与操作系统相匹配的绘制函数,将待处理媒体数据输入至图形处理器中;通过图形处理器进行批量的格式转换处理,以将待处理媒体数据的数据格式转换成目标格式;将图形处理器中属于目标格式的待处理媒体数据回传至中央处理器。

[0152] 在一些实施例中,业务处理结果包括驱动相关数据,上述装置还包括交互模块,用于根据驱动相关数据,控制虚拟交互场景中的虚拟角色,按照待处理媒体数据中对象的动作进行运动。

[0153] 在一些实施例中,目标业务包括播放业务,与目标业务对应的待处理媒体数据包括虚拟场景数据,还包括真实场景数据和媒体数据文件中的至少一种;处理模块还用于对真实场景数据和媒体数据文件中的至少一种进行格式转换,得到适应于虚幻引擎的目标视频数据;在通过虚拟交互应用播放与虚拟场景数据对应的虚拟交互场景的过程中,叠加目标视频数据进行播放。

[0154] 在一些实施例中,目标业务包括视频编码业务,与目标业务对应的待处理媒体数据包括虚拟场景数据,以及真实场景数据或媒体数据文件中的至少一种;处理模块还用于获取预设置的编码状态;启动子线程,通过子线程并按照编码状态,对待处理媒体数据进行编码处理,得到适配于操作系统的目标编码文件。

[0155] 上述媒体数据处理装置中的各个模块可全部或部分通过软件、硬件及其组合来实现。上述各模块可以硬件形式内嵌于或独立于计算机设备中的处理器中,也可以以软件形式存储于计算机设备中的存储器中,以便于处理器调用执行以上各个模块对应的操作。

[0156] 在一个实施例中,提供了一种计算机设备,该计算机设备可以是终端或服务器。以该计算机设备为终端为例,其内部结构图可以如图9所示。该计算机设备包括处理器、存储器、输入/输出接口、通信接口、显示单元和输入装置。其中,处理器、存储器和输入/输出接口通过系统总线连接,通信接口、显示单元和输入装置通过输入/输出接口连接到系统总线。其中,该计算机设备的处理器用于提供计算和控制能力。该计算机设备的存储器包括非易失性存储介质、内存储器。该非易失性存储介质存储有操作系统和计算机程序。该内存储器为非易失性存储介质中的操作系统和计算机程序的运行提供环境。该计算机设备的输入/输出接口用于处理器与外部设备之间交换信息。该计算机设备的通信接口用于与外部的终端进行有线或无线方式的通信,无线方式可通过WIFI、移动蜂窝网络、NFC(近场通信)或其他技术实现。该计算机程序被处理器执行时以实现一种媒体数据处理方法。该计算机设备的显示单元用于形成视觉可见的画面,可以是显示屏、投影装置或虚拟现实成像装置,显示屏可以是液晶显示屏或电子墨水显示屏,该计算机设备的输入装置可以是显示屏上覆盖的触摸层,也可以是计算机设备外壳上设置的按键、轨迹球或触控板,还可以是外接的键盘、触控板或鼠标等。

[0157] 本领域技术人员可以理解,图9中示出的结构,仅仅是与本申请方案相关的部分结构的框图,并不构成对本申请方案所应用于其上的计算机设备的限定,具体的计算机设备可以包括比图中所示更多或更少的部件,或者组合某些部件,或者具有不同的部件布置。

[0158] 在一个实施例中,还提供了一种计算机设备,包括存储器和处理器,存储器中存储有计算机程序,该处理器执行计算机程序时实现上述各方法实施例中的步骤。

[0159] 在一个实施例中,提供了一种计算机可读存储介质,其上存储有计算机程序,该计算机程序被处理器执行时实现上述各方法实施例中的步骤。

[0160] 在一个实施例中,提供了一种计算机程序产品,包括计算机程序,该计算机程序被处理器执行时实现上述各方法实施例中的步骤。

[0161] 需要说明的是,本申请所涉及的媒体数据(包括但不限于用于分析的数据、存储的数据、展示的数据等),均为经用户授权或者经过各方充分授权的信息和数据,且相关数据的收集、使用和处理需要遵守相关国家和地区的相关法律法规和标准。

[0162] 本领域普通技术人员可以理解实现上述实施例方法中的全部或部分流程,是可以通过计算机程序来指令相关的硬件来完成,所述的计算机程序可存储于一非易失性计算机可读存储介质中,该计算机程序在执行时,可包括如上述各方法的实施例的流程。其中,本申请所提供的各实施例中所使用的对存储器、数据库或其它介质的任何引用,均可包括非易失性和易失性存储器中的至少一种。非易失性存储器可包括只读存储器(Read-Only Memory, ROM)、磁带、软盘、闪存、光存储器、高密度嵌入式非易失性存储器、阻变存储器(ReRAM)、磁变存储器(Magnetoresistive Random Access Memory, MRAM)、铁电存储器(Ferroelectric Random Access Memory, FRAM)、相变存储器(Phase Change Memory, PCM)、石墨烯存储器等。易失性存储器可包括随机存取存储器(Random Access Memory, RAM)或外部高速缓冲存储器等。作为说明而非局限,RAM可以是多种形式,比如静态随机存取存储器(Static Random Access Memory, SRAM)或动态随机存取存储器(Dynamic Random Access Memory, DRAM)等。本申请所提供的各实施例中所涉及的数据库可包括关系型数据库和非关系型数据库中至少一种。非关系型数据库可包括基于区块链的分布式数据库等,不限于此。本申请所提供的各实施例中所涉及的处理器可为通用处理器、中央处理器、图形处理器、数字信号处理器、可编程逻辑器、基于量子计算的数据处理逻辑器等,不限于此。

[0163] 以上实施例的各技术特征可以进行任意的组合,为使描述简洁,未对上述实施例中的各个技术特征所有可能的组合都进行描述,然而,只要这些技术特征的组合不存在矛盾,都应当认为是本说明书记载的范围。

[0164] 以上所述实施例仅表达了本申请的几种实施方式,其描述较为具体和详细,但并不能因此而理解为对本申请专利范围的限制。应当指出的是,对于本领域的普通技术人员来说,在不脱离本申请构思的前提下,还可以做出若干变形和改进,这些都属于本申请的保护范围。因此,本申请的保护范围应以所附权利要求为准。

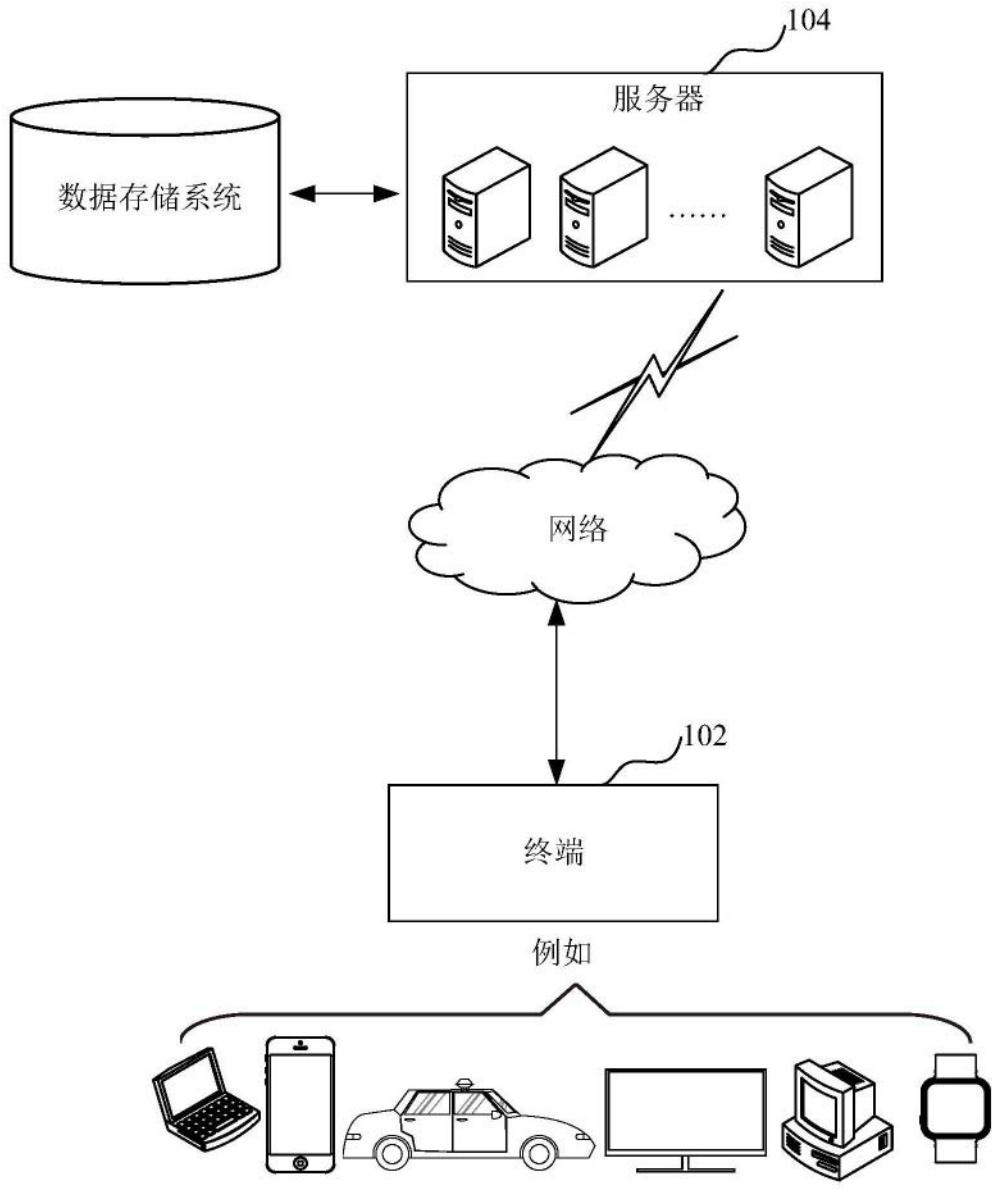


图1

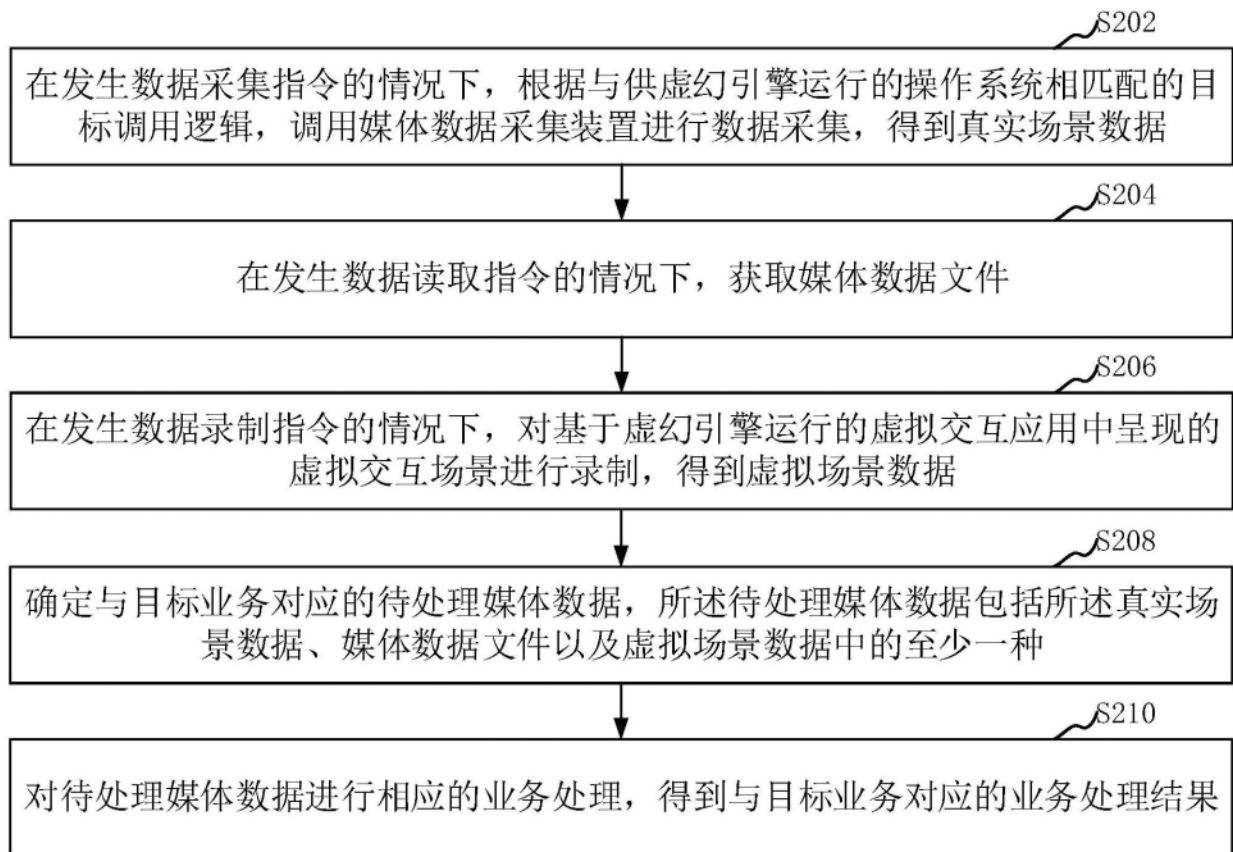


图2

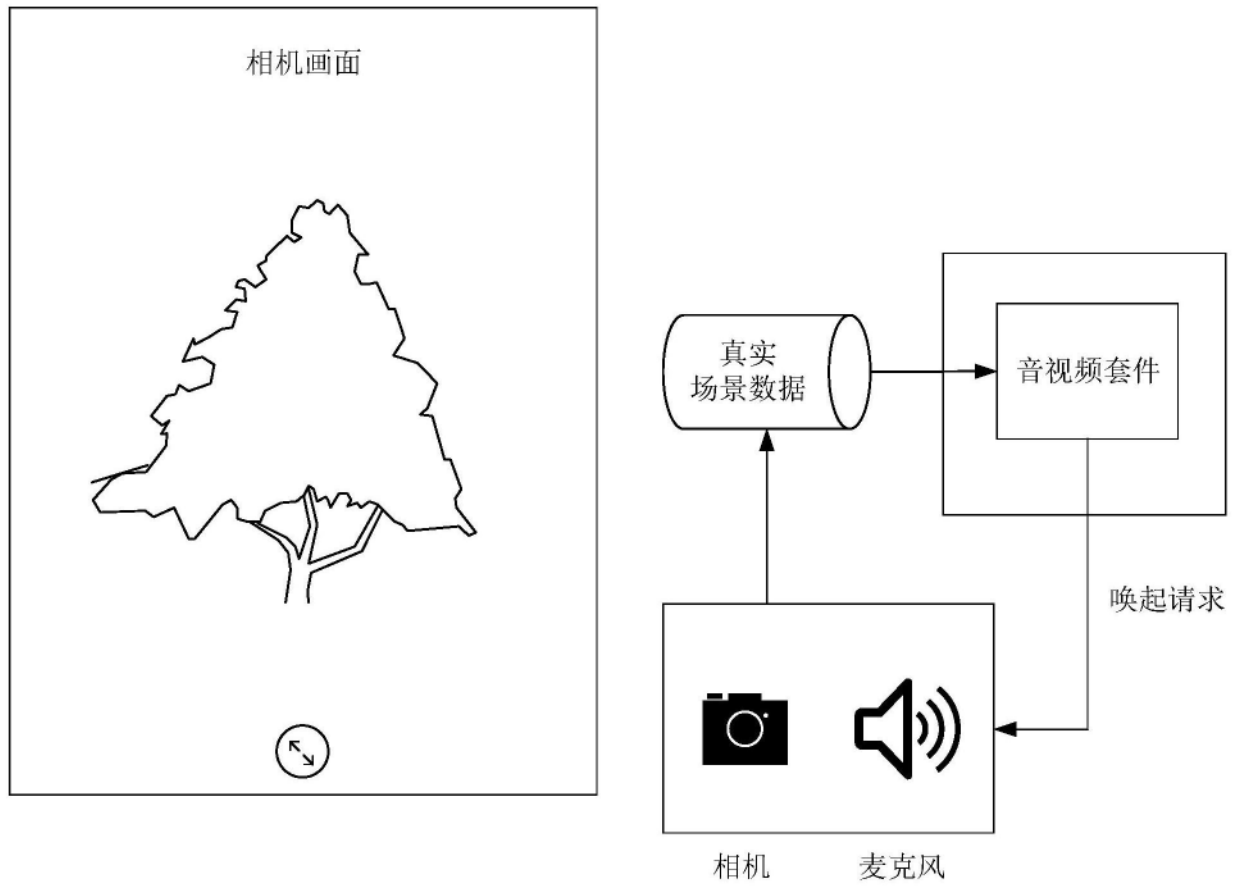


图2A

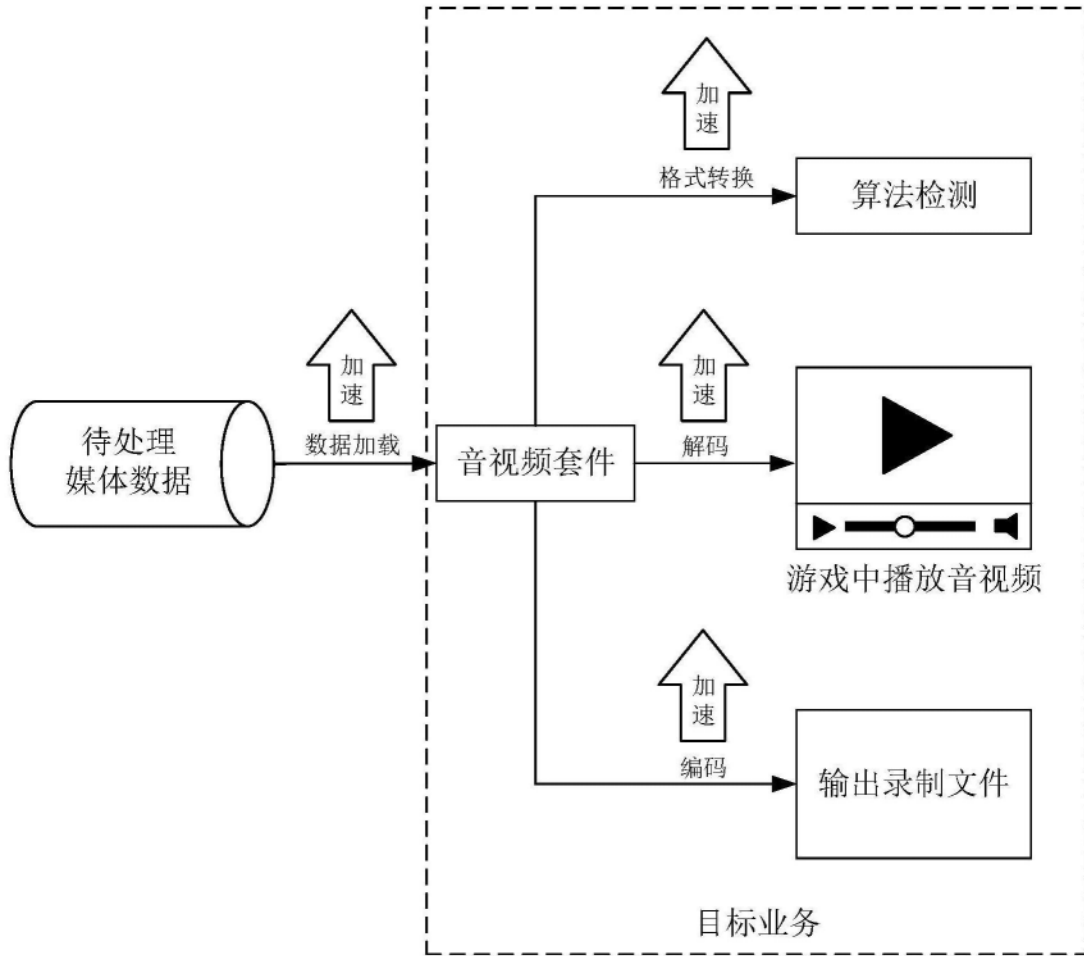


图3

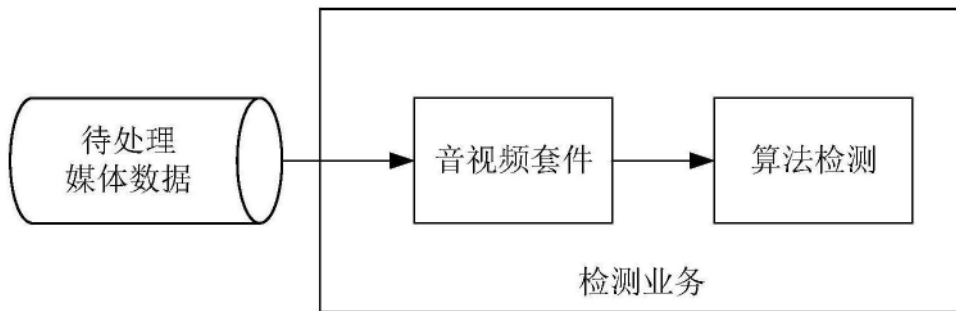


图4A

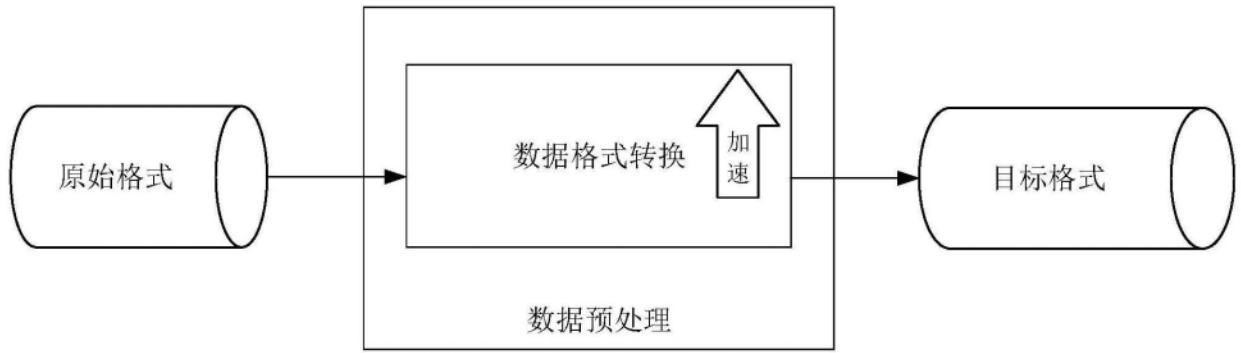


图4B

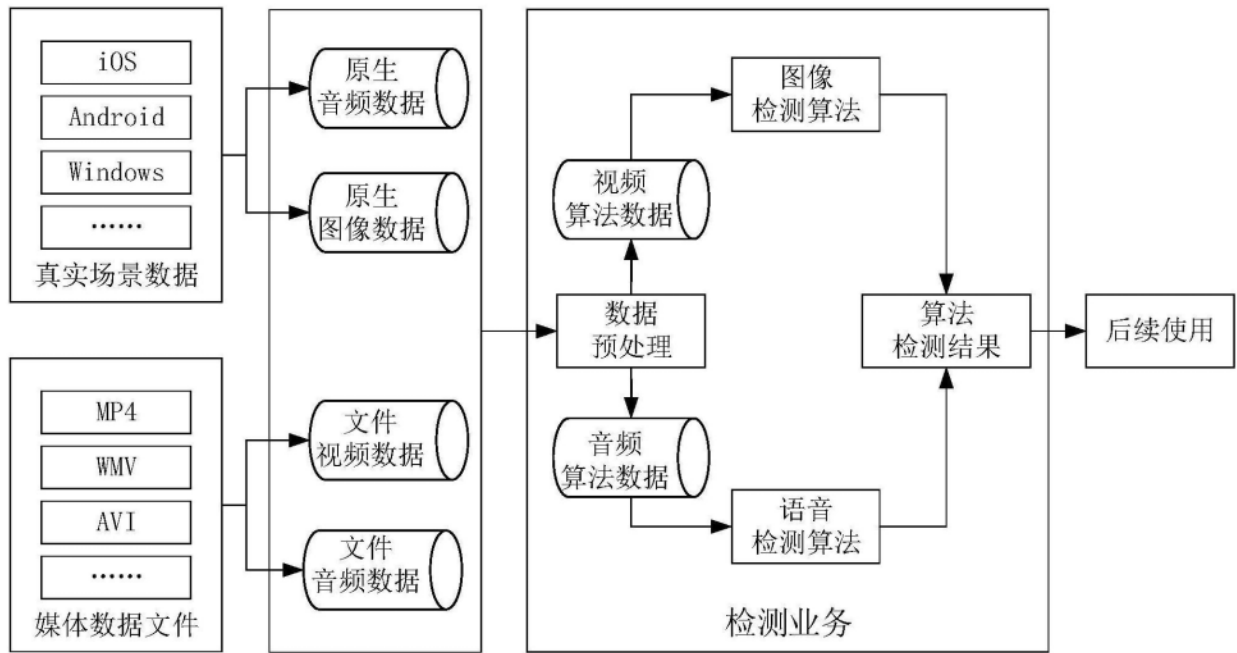


图4C

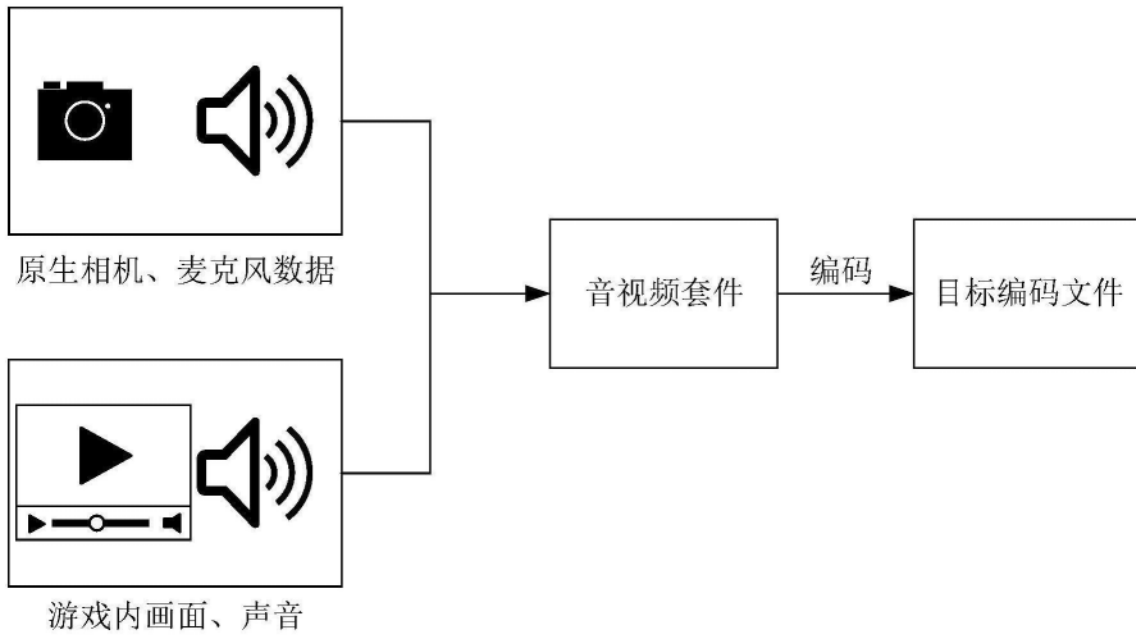


图5A

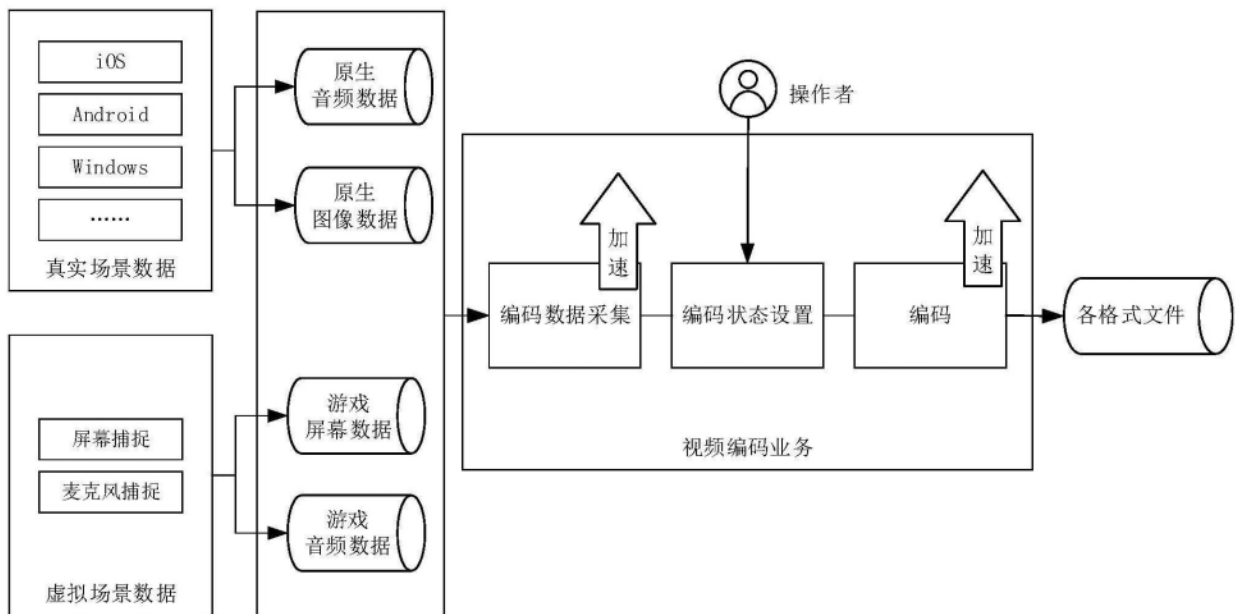


图5B

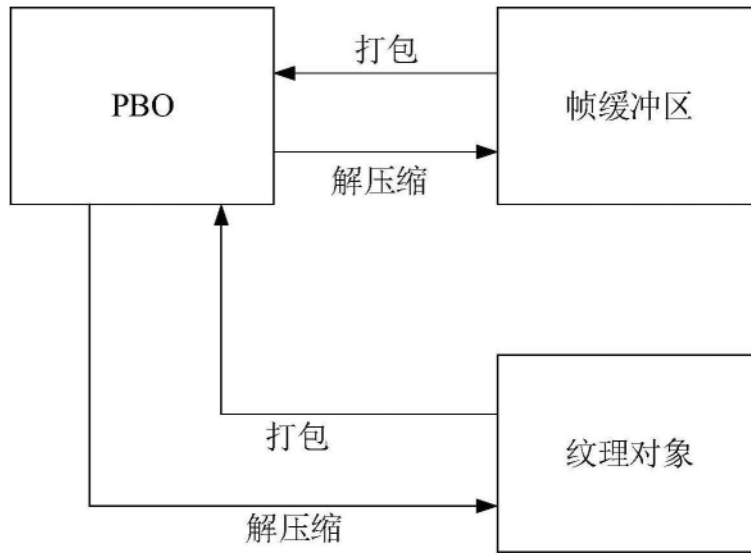


图5C

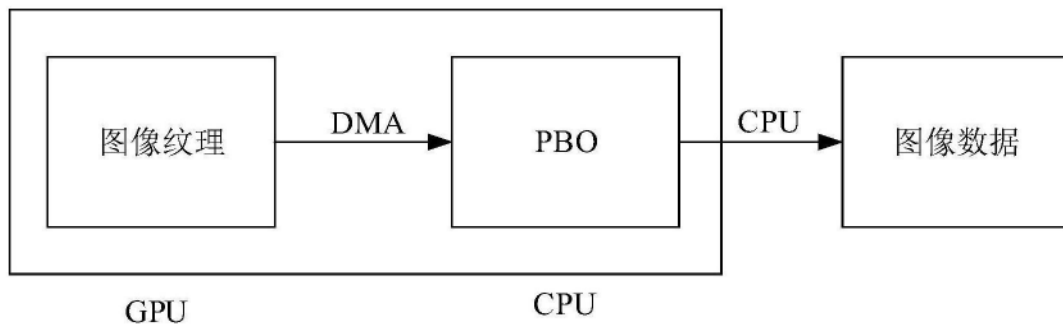


图5D

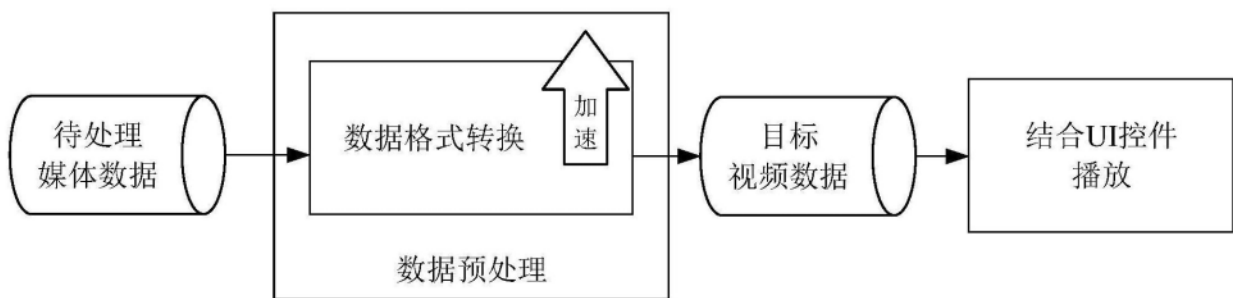


图6

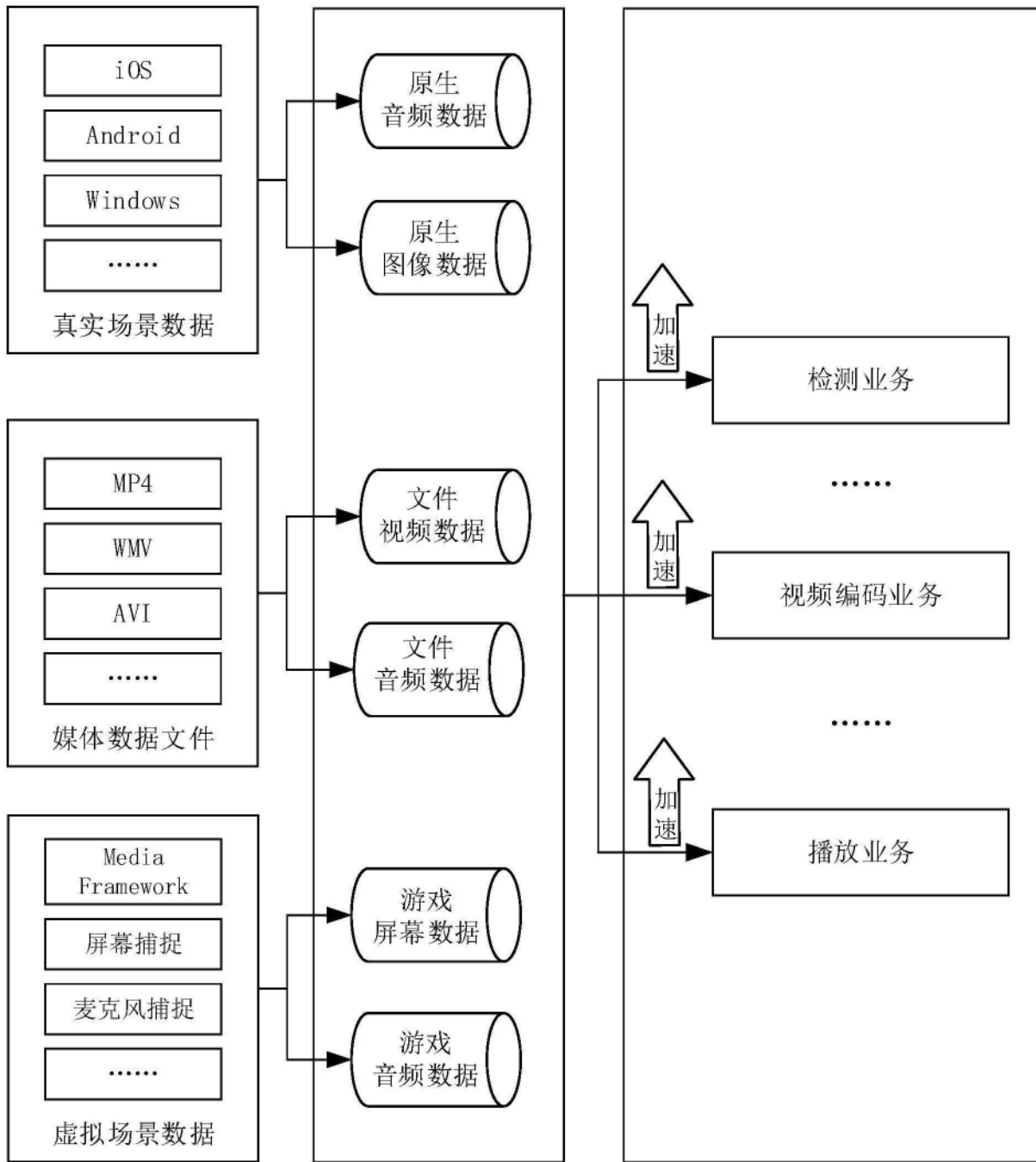


图7

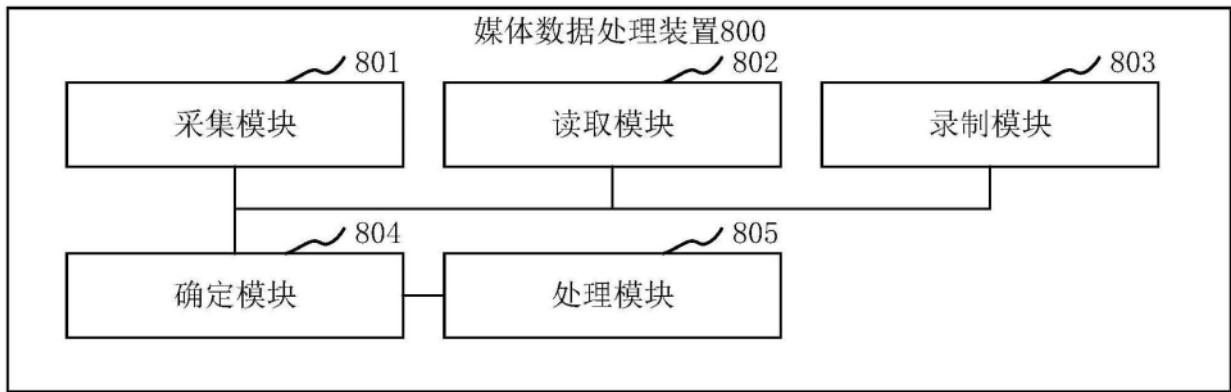


图8

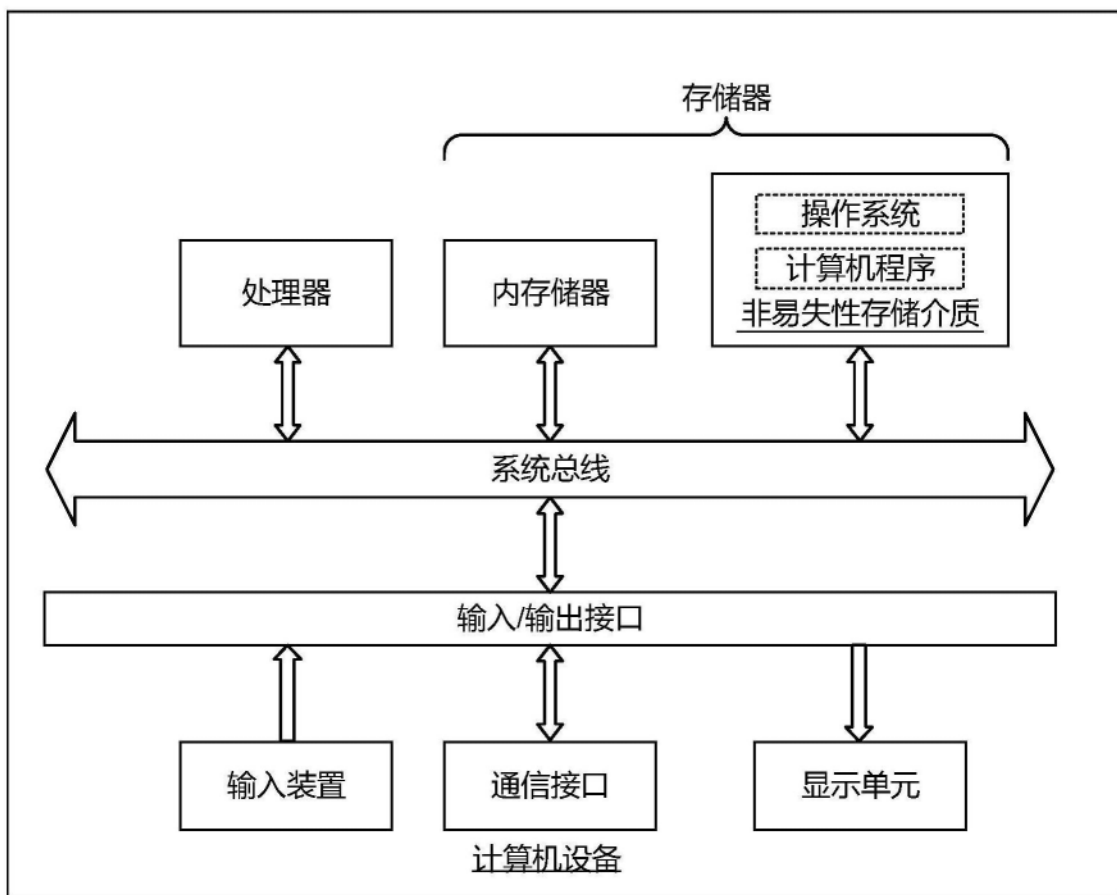


图9