



(12)发明专利申请

(10)申请公布号 CN 107589998 A

(43)申请公布日 2018.01.16

(21)申请号 201710775918.7

(22)申请日 2017.08.31

(71)申请人 广东欧珀移动通信有限公司

地址 523860 广东省东莞市长安镇乌沙海  
滨路18号

(72)发明人 程杰 陈岩

(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 郝传鑫 熊永强

(51)Int.Cl.

G06F 9/50(2006.01)

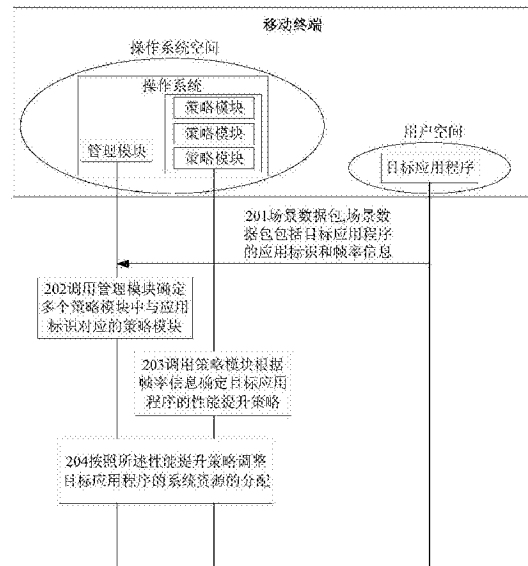
权利要求书2页 说明书13页 附图6页

(54)发明名称

资源配置方法及相关产品

(57)摘要

本申请实施例公开了一种资源配置方法及相关产品。方法包括：移动终端的操作系统中的管理模块接收正在运行的目标应用程序发送的场景数据包，所述场景数据包包括目标应用程序的应用标识和帧率信息；管理模块确定所述多个策略模块中与所述应用标识对应的策略模块；策略模块根据所述帧率信息确定所述目标应用程序的性能提升策略；操作系统按照所述性能提升策略调整所述目标应用程序的系统资源的分配。本申请实施例有利于提高移动终端控制目标应用程序进行性能优化的实时性和精确度。



1. 一种资源配置方法,其特征在于,应用于移动终端,所述移动终端上运行有操作系统以及一个或多个应用程序,所述操作系统包括管理模块和多个策略模块,每个策略模块用于确定一个应用程序的性能提升策略;所述方法包括:

所述管理模块接收正在运行的目标应用程序发送的场景数据包,所述场景数据包包括目标应用程序的应用标识和帧率信息;

所述管理模块确定所述多个策略模块中与所述应用标识对应的策略模块;

所述策略模块根据所述帧率信息确定所述目标应用程序的性能提升策略;

所述操作系统按照所述性能提升策略调整所述目标应用程序的系统资源的分配。

2. 根据权利要求1所述的方法,其特征在于,所述策略模块根据所述帧率信息确定所述目标应用程序的性能提升策略,包括:

所述策略模块确定所述帧率信息对应的待调整的系统资源;

所述策略模块根据所述待调整的系统资源和所述帧率信息确定所述目标应用程序的性能提升策略。

3. 根据权利要求2所述的方法,其特征在于,所述策略模块根据所述待调整的系统资源和所述帧率信息确定所述目标应用程序的性能提升策略,包括:

所述策略模块根据所述帧率信息确定所述目标应用程序当前界面的帧率值;

所述策略模块在检测到所述帧率值低于预设帧率值时,确定所述帧率值和所述预设帧率值之间的差值;

所述策略模块根据所述差值确定上调所述待调整的系统资源的调整量;

所述策略模块根据所述调整量生成所述目标应用程序的性能提升策略。

4. 根据权利要求2所述的方法,其特征在于,所述策略模块根据所述待调整的系统资源和所述帧率信息确定所述目标应用程序的性能提升策略,包括:

所述策略模块根据所述帧率信息确定所述目标应用程序当前界面的帧率值;

所述策略模块确定所述帧率值所属的参考帧率区间;

所述策略模块查找预设的帧率区间和待调整的系统资源的调整量之间的映射关系,确定所述参考帧率区间对应的所述待调整的系统资源的调整量。

所述策略模块根据所述调整量生成所述目标应用程序的性能提升策略。

5. 根据权利要求2所述的方法,其特征在于,所述策略模块根据所述待调整的系统资源和所述帧率信息确定所述目标应用程序的性能提升策略,包括:

所述策略模块根据所述帧率信息确定所述目标应用程序当前界面的帧率值;

所述策略模块在检测到所述帧率值高于预设帧率值时,确定所述帧率值和所述预设帧率值之间的差值;

所述策略模块根据所述差值确定下调所述待调整的系统资源的调整量;

所述策略模块根据所述调整量生成所述目标应用程序的性能提升策略。

6. 根据权利要求1-5任一项所述的方法,其特征在于,所述目标应用程序包括数据连接模块,所述操作系统包括数据连接管理模块,所述数据连接管理模块与所述数据连接模块建立预设数据通道;所述管理模块接收正在运行的目标应用程序发送的场景数据包,包括:

所述管理模块通过所述预设数据通道接收所述目标应用程序发送的场景数据包。

7. 根据权利要求1-6任一项所述的方法,其特征在于,所述管理模块接收正在运行的所

述目标应用程序发送的场景数据包之前,所述方法还包括:

所述管理模块向所述目标应用程序发送第一数据格式,所述第一数据格式用于所述目标应用程序转换所述场景数据包的第二数据格式为所述第一数据格式。

8. 一种资源配置装置,其特征在于,应用于移动终端,所述移动终端上运行有操作系统以及一个或多个应用程序,所述资源配置装置包括处理单元和通信单元,

所述处理单元,用于控制所述管理模块通过所述通信单元正在运行的目标应用程序发送的场景数据包,所述场景数据包包括目标应用程序的应用标识和帧率信息;以及用于控制所述管理模块确定所述多个策略模块中与所述应用标识对应的策略模块;以及用于控制所述策略模块根据所述帧率信息确定所述目标应用程序的性能提升策略;以及用于控制所述操作系统按照所述性能提升策略调整所述目标应用程序的系统资源的分配。

9. 一种移动终端,其特征在于,包括处理器、存储器、通信接口以及一个或多个程序,其中,所述一个或多个程序被存储在所述存储器中,并且被配置由所述处理器执行,所述程序包括用于执行权利要求1-7任一项方法中的步骤的指令。

10. 一种计算机可读存储介质,其特征在于,所述计算机可读存储介质存储用于电子数据交换的计算机程序,其中,所述计算机程序使得计算机执行如权利要求1-7任一项所述的方法,所述计算机包括移动终端。

## 资源配置方法及相关产品

### 技术领域

[0001] 本申请涉及移动终端技术领域,具体涉及资源配置方法及相关产品。

### 背景技术

[0002] 随着智能手机相关技术的快速发展,越来越多的应用被安装在用户手机中,如阅读类应用、支付类应用、游戏类应用、音乐类应用等,人们的衣食住行已经与手机密不可分。

### 发明内容

[0003] 本申请实施例提供了资源配置方法及相关产品,可以提高移动终端的目标应用程序在运行过程中的性能和稳定性。

[0004] 第一方面,本申请实施例提供一种资源配置方法,应用于移动终端,所述移动终端上运行有操作系统以及一个或多个应用程序,所述操作系统包括管理模块和多个策略模块,每个策略模块用于确定一个应用程序的性能提升策略;所述方法包括:

[0005] 所述管理模块接收正在运行的目标应用程序发送的场景数据包,所述场景数据包包括目标应用程序的应用标识和帧率信息;

[0006] 所述管理模块确定所述多个策略模块中与所述应用标识对应的策略模块;

[0007] 所述策略模块根据所述帧率信息确定所述目标应用程序的性能提升策略;

[0008] 所述操作系统按照所述性能提升策略调整所述目标应用程序的系统资源的分配。

[0009] 第二方面,本申请实施例提供一种资源配置装置,应用于移动终端,上述移动终端上运行有操作系统以及一个或多个应用程序,上述资源配置装置包括处理单元和通信单元,

[0010] 所述处理单元,用于控制所述管理模块通过所述通信单元正在运行的目标应用程序发送的场景数据包,所述场景数据包包括目标应用程序的应用标识和帧率信息;以及用于控制所述管理模块确定所述多个策略模块中与所述应用标识对应的策略模块;以及用于控制所述策略模块根据所述帧率信息确定所述目标应用程序的性能提升策略;以及用于控制所述操作系统按照所述性能提升策略调整所述目标应用程序的系统资源的分配。

[0011] 第三方面,本申请实施例提供一种移动终端,包括处理器、存储器、通信接口以及一个或多个程序,其中,上述一个或多个程序被存储在上述存储器中,并且被配置由上述处理器执行,上述程序包括用于执行本申请实施例第一方面任一方法中的步骤的指令。

[0012] 第四方面,本申请实施例提供了一种计算机可读存储介质,其中,上述计算机可读存储介质存储用于电子数据交换的计算机程序,其中,上述计算机程序使得计算机执行如本申请实施例第一方面任一方法中所描述的部分或全部步骤,上述计算机包括移动终端。

[0013] 第五方面,本申请实施例提供了一种计算机程序产品,其中,上述计算机程序产品包括存储了计算机程序的非瞬时性计算机可读存储介质,上述计算机程序可操作来使计算机执行如本申请实施例第一方面任一方法中所描述的部分或全部步骤。该计算机程序产品可以为一个软件安装包,上述计算机包括移动终端。

[0014] 可以看出,本申请实施例中,移动终端的操作系统中的管理模块首先接收正在运行的目标应用程序发送的场景数据包,所述场景数据包包括目标应用程序的应用标识和帧率信息,其次,所述管理模块确定所述多个策略模块中与所述应用标识对应的策略模块,然后,所述策略模块根据所述帧率信息确定所述目标应用程序的性能提升策略,最后,所述操作系统按照所述性能提升策略调整所述目标应用程序的系统资源的分配。有利于提高移动终端控制目标应用程序进行性能优化的实时性和精确度。

### 附图说明

[0015] 下面将对本申请实施例所涉及到的附图作简单地介绍。

[0016] 图1A是一种智能手机的程序运行空间的示意图;

[0017] 图1B是一种安卓系统的系统架构图;

[0018] 图2是本申请实施例提供的一种资源配置方法的流程示意图;

[0019] 图3是本申请实施例公开的一种资源配置方法的流程示意图;

[0020] 图4是本申请实施例公开的一种资源配置方法的流程示意图;

[0021] 图5是本申请实施例公开的一种移动终端的结构示意图;

[0022] 图6是本申请实施例公开的一种移动终端的功能单元组成框图;

[0023] 图7是本申请实施例公开的一种智能手机的结构示意图。

### 具体实施方式

[0024] 为了使本技术领域的人员更好地理解本申请方案,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。

[0025] 本申请的说明书和权利要求书及上述附图中的术语“第一”、“第二”等是用于区别不同对象,而不是用于描述特定顺序。此外,术语“包括”和“具有”以及它们任何变形,意图在于覆盖不排他的包含。例如包含了一系列步骤或单元的过程、方法、系统、产品或设备没有限定于已列出的步骤或单元,而是可选地还包括没有列出的步骤或单元,或可选地还包括对于这些过程、方法、产品或设备固有的其他步骤或单元。

[0026] 在本文中提及“实施例”意味着,结合实施例描述的特定特征、结构或特性可以包含在本申请的至少一个实施例中。在说明书中的各个位置出现该短语并不一定均是指相同的实施例,也不是与其它实施例互斥的独立的或备选的实施例。本领域技术人员显式地和隐式地理解的是,本文所描述的实施例可以与其它实施例相结合。

[0027] 本申请实施例所涉及到的移动终端可以包括各种具有无线通信功能的手持设备、车载设备、可穿戴设备、计算设备或连接到无线调制解调器的其他处理设备,以及各种形式的用户设备(User Equipment,UE),移动台(Mobile Station,MS),终端设备(terminal device)等等。为方便描述,上面提到的设备统称为移动终端。本发明实施例所涉及到的操作系统是对硬件资源进行统一管理,并向用户提供业务接口的软件系统。

[0028] 如图1A所示,目前智能手机等移动终端一般设置有程序运行空间,该程序运行空间包括用户空间和操作系统空间,其中,用户空间运行有一个或多个应用程序,该一个或多

个应用程序为移动终端安装的第三方应用程序,操作系统空间运行有移动终端的操作系统。该移动终端具体可以运行安卓Android系统、苹果公司开发的移动操作系统iOS等,此处不做唯一限定。如图1B所示,以上述移动终端运行有Android系统为例,对应的用户空间包括该Android系统中的应用层(Applications),操作系统空间可以包括该Android系统中的应用程序框架层(Application Framework)、系统运行库层(包括系统运行库层Libraries和Android运行时Android Runtime)、Linux内核层(Linux Kernel)。其中,应用层上包括各类与用户直接交互的应用程序,或由Java语言编写的运行于后台的服务程序。例如,智能手机上实现的常见基本功能的程序,诸如短消息业务(Short Messaging Service, SMS)短信,电话拨号,图片浏览器,日历,游戏,地图,万维网(World Wide Web, Web)浏览器等程序,以及开发人员开发的其他应用程序。应用程序框架层提供开发Android应用程序所需的一系列类库,能够用于重用组件,也可以通过继承实现个性化的扩展。系统运行库层是应用程序框架的支撑,为Android系统中的各个组件提供服务。系统运行库层由系统类库和Android运行时构成。Android运行时包含核心库和Dalvik虚拟机两部分。Linux内核层用于实现硬件设备驱动,进程和内存管理,网络协议栈,电源管理,无线通信等核心功能。

[0029] 上述第三方应用程序所创建的进程一开始是运行在用户空间的,当它要执行网络发送数据、读取磁盘资源等动作时,必须通过调用write、send等由操作系统提供的标准接口函数来完成,即由CPU调用操作系统空间的代码来完成用户的请求操作。可见,第三方应用程序想要调用操作系统提供的功能,只能调用其提供的标准接口函数。同时,操作系统不知道第三方应用程序具体在执行哪些功能,目标应用程序也不能命令操作系统执行特殊的操作,这两者是相互独立的,这使得操作系统无法区分目标应用程序的内部运行场景,只能使用一套标准的参数去适配,无法针对目标应用程序的内部运行场景进行针对性的性能优化。

[0030] 针对上述情况,本申请实施例提出一种针对移动终端的目标应用程序的资源配置方法,该方法中,管理模块接收正在运行的目标应用程序发送的场景数据包,场景数据包包括目标应用程序的应用标识和帧率信息;管理模块确定多个策略模块中与应用标识对应的策略模块;策略模块根据帧率信息确定目标应用程序的性能提升策略;操作系统按照性能提升策略调整目标应用程序的系统资源的分配。实现由操作系统针对目标应用程序在运行过程中的内部运行场景的性能进行优化,有利于提高移动终端控制目标应用程序进行性能优化的实时性和精确度。

[0031] 下面结合附图对本申请实施例进行介绍。

[0032] 请参阅图2,图2是本申请实施例提供了一种资源配置方法的流程示意图,应用于移动终端,所述移动终端上运行有操作系统以及一个或多个应用程序,所述操作系统包括管理模块和多个策略模块,每个策略模块用于确定一个应用程序的性能提升策略,如图所示,本资源配置方法包括:

[0033] S201,上述管理模块接收正在运行的目标应用程序发送的场景数据包,所述场景数据包包括目标应用程序的应用标识和帧率信息。

[0034] 其中,目标应用程序是指安装在移动终端的用户空间的第三方应用程序,该第三方应用程序可以是游戏类应用程序,该第三方游戏类应用程序可以由用户安装,也可以由开发人员在移动终端出厂前预装,此处不做唯一限定。

[0035] 其中,场景数据包包括目标应用程序正在运行时的内部运行场景的帧率信息,在目标应用程序为游戏应用程序时,帧率信息包括游戏应用程序的显示界面的帧率。游戏应用程序的显示界面的最高帧率一般限制在60帧,但是操作系统无法获取到显示界面的帧率信息,游戏应用程序可在获取当前的帧率信息后,将帧率信息打包在数据包中发送给操作系统的管理模块。

[0036] 其中,场景数据包可以包括正在运行的目标应用程序的游戏应用标识和帧率信息,还可以包括以下至少一种:关键性能参数、数据格式标识、启动信息、资源更新信息、账号登陆信息、界面类型信息、目标应用程序的行为信息(如数据发送行为信息、数据接收行为信息等)等,此处不做唯一限定。此外,场景数据包的格式例如可以是Java脚本对象标记(JavaScript Object Notation,JSON)、协议缓冲区(ProtocolBuffer,Protobuf)、自定义格式等数据传输格式,此处不做唯一限定。

[0037] 其中,内部运行场景可以按照不同预设条件进行划分,如按照是否有用户交互可以划分内部运行场景为交互式场景和非交互式场景,或者,按照任务的复杂度划分为单用户场景和多用户场景,或者按照不同的行为划分为启动场景、资源更新场景、账号登录场景、商城场景、主界面场景等。

[0038] 在一个可能的示例中,所述目标应用程序包括数据连接模块,所述操作系统包括数据连接管理模块,所述数据连接管理模块与所述数据连接模块建立预设数据通道;所述管理模块接收正在运行的目标应用程序发送的场景数据包,包括:所述管理模块通过所述预设数据通道接收所述目标应用程序发送的场景数据包。

[0039] 其中,上述预设数据通道为目标应用程序正在运行时,目标应用程序与操作系统之间建立的有效的数据传输链路,目标应用程序不在运行后,该数据传输链路会被清除。

[0040] 在本可能的示例中,目标应用程序还可以包括数据收集模块、数据整理模块、优先级管理模块、数据接收模块,其中,数据收集模块用于收集目标应用程序在运行时的关键信息;数据整理模块用于整理收集到的关键信息,如合并重复的信息,删除冗余的信息等;优先级管理模块用于根据关键信息的重要程度对收集到的关键信息进行分类,优先处理优先级高的关键信息,过滤优先级较低的关键信息,或者在数据包生成过程中减少优先级较低的关键信息的使用;数据接收模块用于接收并解析操作系统发送的数据,并将其传递给目标应用程序内部。

[0041] S202,上述管理模块确定所述多个策略模块中与所述应用标识对应的策略模块。

[0042] 其中,上述多个策略模块中的每个策略模块可适用于一个应用,多个策略模块对应多个不同的应用。如第一策略模块适用于第一游戏应用程序,第二策略模块适用于第二游戏应用程序等。

[0043] 其中,上述管理模块确定上述多个策略模块中与上述应用标识对应的策略模块,包括:上述管理模块以应用标识为查询标识,查询预设的策略模块和应用标识之间的映射关系,确定场景数据包中的应用标识对应的策略模块。

[0044] S203,上述策略模块根据所述帧率信息确定所述目标应用程序的性能提升策略。

[0045] 其中,性能提升策略是指用于提高上述正在运行的目标应用程序的运行性能的资源配置策略,如CPU资源调整策略、GPU资源调整策略、内存带宽资源调整策略、磁盘资源调整策略、网络资源调整策略等,此处不做唯一限定。

[0046] S204,上述操作系统按照所述性能提升策略调整所述目标应用程序的系统资源的分配。

[0047] 在一个可能的示例中,上述资源包括上述移动终端的以下资源中的至少一种:CPU资源、GPU资源、内存带宽资源、磁盘资源、网络资源。其中,网络资源包括移动终端的数据网络的资源、无线保真Wi-Fi网络的资源、蓝牙模块的控制参数资源等。

[0048] 其中,在上述系统资源为CPU资源、GPU资源内存带宽资源、磁盘资源中的至少一种时,操作系统可以通过直连通信方式与操作系统空间的内核层通信,调整该系统资源的配置,直连通信方式是指通过抽象的应用程序编程接口(Application Programming Interface,API)直接通信。

[0049] 其中,在上述系统资源为网络资源时,操作系统可以通过间接通信方式与操作系统空间的内核层通信,调整该系统资源的配置,间接通信方式是指通过调用代理服务来间接通信,如网络资源中的Wi-Fi子系统或者数据网络子系统与操作系统不运行在同一个系统内,需要通过一些代理的方式,间接访问这些系统资源,操作系统中提供了Wi-Fi的代理服务,通过调用该代理服务的接口来间接和Wi-Fi子系统通信。

[0050] 可以看出,本申请实施例中,移动终端的操作系统中的管理模块首先接收正在运行的目标应用程序发送的场景数据包,所述场景数据包包括目标应用程序的应用标识和帧率信息,其次,管理模块确定所述多个策略模块中与所述应用标识对应的策略模块,然后,策略模块根据所述帧率信息确定所述目标应用程序的性能提升策略,最后,操作系统按照所述性能提升策略调整所述目标应用程序的系统资源的分配。由于场景数据包包括目标应用程序的应用标识和帧率信息,且场景数据包由正在运行的目标应用程序发送,操作系统能够根据应用标识准确确定适用的策略模块,进而调用该策略模块根据场景标识精细化确定目标应用程序的当前的内部运行场景的性能提升策略,并由操作系统按照该性能提升策略实时调整目标应用程序的系统资源的配置,实现了由操作系统对运行过程中的目标应用程序进行实时的性能优化,有利于提高移动终端控制目标应用程序进行性能优化的实时性和精确度。

[0051] 在一个可能的示例中,所述策略模块根据所述帧率信息确定所述目标应用程序的性能提升策略,包括:所述策略模块确定所述帧率信息对应的待调整的系统资源;所述策略模块根据所述待调整的系统资源和所述帧率信息确定所述目标应用程序的性能提升策略。

[0052] 其中,策略模块可以查询预设的帧率信息与资源之间的对应关系,以获取待调整的系统资源,该对应关系例如可以包括帧率与GPU资源之间的对应关系,或者帧率与资源组合(例如:CPU资源和GPU资源)之间的对应关系,此处不做唯一限定。

[0053] 可见,本示例中,帧率信息关联的资源为待调整的系统资源,根据待调整的系统资源可进一步确定目标应用程序的性能提升策略,有利于提高移动终端控制目标应用程序进行优化的准确度。

[0054] 在一个可能的示例中,所述策略模块根据所述待调整的系统资源和所述帧率信息确定所述目标应用程序的性能提升策略,包括:所述策略模块根据所述帧率信息确定所述目标应用程序当前界面的帧率值;所述策略模块在检测到所述帧率值低于预设帧率值时,确定所述帧率值和所述预设帧率值之间的差值;所述策略模块根据所述差值确定上调所述待调整的系统资源的调整量;所述策略模块根据所述调整量生成所述目标应用程序的性能



提升策略。

[0055] 其中,帧率信息包括目标应用程序当前的帧率值,即每秒显示的帧数,游戏应用程序的帧率一般限定在60帧,预设帧率值可为小于60的数值,例如为50帧,在检测游戏应用程序当前显示界面的帧率值小于50帧时,计算帧率值和预设帧率值之间的差值,策略模块根据该差值确定待调整资源的调整量,该调整量可以由预设公式计算得到,具体可以是差值越大,调整量越大,或者,以所述差值为查询标识,查询预设的差值和调整量之间的映射关系,确定所述差值对应的调整量为所述待调整的系统资源的调整量。

[0056] 其中,策略模块在检测到帧率值低于预设帧率值时,确定帧率信息对应的待调整资源,待调整资源具体可为CPU资源,例如CPU数量和频率,或者内存资源,或者磁盘资源,以此提升游戏帧率,提高用户体验。

[0057] 可见,本示例中,通过在检测到帧率信息中的帧率值小于预设帧率值时,通过确定帧率值和预设帧率值之间的差值来确定上调待调整的系统资源的调整量,从而可进一步确定目标应用程序的性能提升策略,有利于提高移动终端控制目标应用程序进行优化的准确度。

[0058] 在一个可能的示例中,所述策略模块根据所述待调整的系统资源和所述帧率信息确定所述目标应用程序的性能提升策略,包括:所述策略模块根据所述帧率信息确定所述目标应用程序当前界面的帧率值;所述策略模块确定所述帧率值所属的参考帧率区间;所述策略模块查找预设的帧率区间和待调整的系统资源的调整量之间的映射关系,确定所述参考帧率区间对应的所述待调整的系统资源的调整量。所述策略模块根据所述调整量生成所述目标应用程序的性能提升策略。

[0059] 其中,设置有多个参考帧率区间,每个参考帧率区间对应有待调整的系统资源的调整量,形成帧率区间很热待调整的系统资源的调整量之间的映射关系。

[0060] 其中,策略模块在根据帧率信息确定目标应用程序当前界面的帧率值后,即可确定该帧率值所属的参考帧率区间,通过查找预设的帧率区间和待调整的系统资源的调整量之间的映射关系,确定参考区间对应的待调整的系统资源的调整量。

[0061] 可见,本示例中,帧率信息包括目标应用程序的当前帧率值,通过确定该帧率值对应的参考帧率区间,再进一步确定帧率值对应的待调整的系统资源的调整量,有利于提高移动终端控制目标应用程序进行运行优化的准确度。

[0062] 在一种可能的示例中,所述策略模块根据所述待调整的系统资源和所述帧率信息确定所述目标应用程序的性能提升策略,包括:所述策略模块根据所述帧率信息确定所述目标应用程序当前界面的帧率值;所述策略模块在检测到所述帧率值高于预设帧率值时,确定所述帧率值和所述预设帧率值之间的差值;所述策略模块根据所述差值确定下调所述待调整的系统资源的调整量;所述策略模块根据所述调整量生成所述目标应用程序的性能提升策略。

[0063] 其中,在检测到帧率值大于预设帧率值时,即帧率过剩的情况,这时操作系统可以下调目标应用程序的系统资源,如减少对应的系统资源的分配以降低帧率,如此可以将目标应用程序的帧率稳定在预设帧率范围内,优化目标应用程序的功耗,有利于提升用户体验。

[0064] 可见,本示例中,通过在检测到帧率信息中的帧率值大于预设帧率值时,通过确定

帧率值和预设帧率值之间的差值来确定下调待调整的系统资源的调整量,从而可进一步确定目标应用程序的性能提升策略,有利于提高移动终端控制目标应用程序进行优化的准确度。

[0065] 在一个可能的示例中,所述管理模块接收正在运行的所述目标应用程序发送的场景数据包之前,所述方法还包括:所述管理模块向所述目标应用程序发送第一数据格式,所述第一数据格式用于所述目标应用程序转换所述场景数据包的所述第二数据格式为所述第一数据格式。

[0066] 其中,数据格式标识例如可以是上述JSON、Protobuf等,此处不做限定。

[0067] 其中,上述管理模块按照上述第一数据格式标识所指示的数据格式解析上述场景数据包,确定上述目标应用程序的上述游戏应用类型标识和上述游戏加载场景标识,并向上述策略模块发送上述游戏应用类型标识和上述游戏加载场景标识;可见,由管理模块对场景数据包进行统一处理,策略模块仅需要直接接收管理模块的处理结果,如此设置有利于轻量化和精细化策略模块的功能,便于对策略模块进行维护、添加等操作,提高系统稳定性。此外,可避免因格式差异无法正确确定出应用标识和场景标识,有利于提高移动终端为运行中的目标应用程序进行准确资源配置的成功率。

[0068] 可见,本示例中,目标应用程序和操作系统约定好特定的数据格式传输场景数据包,使得操作系统在接收到场景数据包后,能够实时解析该场景数据包中的应用标识和场景标识,降低处理时延,有利于提高数据处理效率,提高针对运行中的目标应用程序进行资源调整的实时性。

[0069] 与上述图2所示的实施例一致的,请参阅图3,图3是本申请实施例提供的一种资源配置方法的流程示意图,应用于移动终端,上述移动终端上运行有操作系统以及一个或多个应用程序,上述操作系统包括管理模块和多个策略模块,每个策略模块用于确定一个应用程序的性能提升策略。如图所示,本资源配置方法包括:

[0070] S301,所述管理模块接收正在运行的目标应用程序发送的场景数据包,所述场景数据包包括目标应用程序的应用标识和帧率信息。

[0071] S302,所述管理模块确定所述多个策略模块中与所述应用标识对应的策略模块。

[0072] S303,所述策略模块根据所述帧率信息确定所述目标应用程序当前界面的帧率值。

[0073] S304,所述策略模块在检测到所述帧率值低于预设帧率值时,确定所述帧率值和所述预设帧率值之间的差值。

[0074] S305,所述策略模块根据所述差值确定上调所述待调整的系统资源的调整量。

[0075] S306,所述策略模块根据所述调整量生成所述目标应用程序的性能提升策略。

[0076] S307,所述操作系统按照所述性能提升策略调整所述目标应用程序的系统资源的分配。

[0077] 可以看出,本申请实施例中,移动终端的操作系统中的管理模块首先接收正在运行的目标应用程序发送的场景数据包,所述场景数据包包括目标应用程序的应用标识和帧率信息,其次,管理模块确定所述多个策略模块中与所述应用标识对应的策略模块,然后,策略模块根据所述帧率信息确定所述目标应用程序的性能提升策略,最后,操作系统按照所述性能提升策略调整所述目标应用程序的系统资源的分配。由于场景数据包包括目标应

用程序的应用标识和帧率信息,且场景数据包由正在运行的目标应用程序发送,操作系统能够根据应用标识准确确定适用的策略模块,进而调用该策略模块根据场景标识精细化确定目标应用程序的当前的内部运行场景的性能提升策略,并由操作系统按照该性能提升策略实时调整目标应用程序的系统资源的配置,实现了由操作系统对运行过程中的目标应用程序进行实时的性能优化,有利于提高移动终端控制目标应用程序进行性能优化的实时性和精确度。

[0078] 此外,通过划分多个策略模块能够实现对不同类型应用的精细化控制,且移动终端无需同时加载全部策略模块,仅需要加载移动终端前台运行的目标应用程序对应的策略模块,有利于降低移动终端的资源占用,提高资源利用率和控制效率。

[0079] 与上述图2所示的实施例一致的,请参阅图4,图4是本申请实施例提供的一种资源配置方法的流程示意图,应用于移动终端,上述移动终端上运行有操作系统和一个或多个目标应用程序,上述操作系统包括管理模块、多个策略模块和数据连接管理模块,每个策略模块用于确定一个应用程序的性能提升策略,上述目标应用程序包括数据连接模块,上述场景数据包包括目标应用程序的游戏应用类型标识和游戏加载场景标识,上述场景数据包包括数据格式标识。如图所示,本资源配置方法包括:

[0080] S401,上述数据连接管理模块与上述数据连接模块建立数据通道。

[0081] S402,上述数据连接管理模块通过上述数据通道接收上述目标应用程序发送的场景数据包,所述场景数据包包括目标应用程序的应用标识和帧率信息。

[0082] S403,所述管理模块确定所述多个策略模块中与所述应用标识对应的策略模块。

[0083] S404,所述策略模块根据所述帧率信息确定所述目标应用程序当前界面的帧率值。

[0084] S405,所述策略模块在检测到所述帧率值低于预设帧率值时,确定所述帧率值和所述预设帧率值之间的差值。

[0085] S406,所述策略模块根据所述差值确定上调所述待调整的系统资源的调整量。

[0086] S407,所述策略模块根据所述调整量生成所述目标应用程序的性能提升策略。

[0087] S408,所述操作系统按照所述性能提升策略调整所述目标应用程序的系统资源的分配。

[0088] 可以看出,本申请实施例中,移动终端的操作系统中的管理模块首先接收正在运行的目标应用程序发送的场景数据包,所述场景数据包包括目标应用程序的应用标识和帧率信息,其次,管理模块确定所述多个策略模块中与所述应用标识对应的策略模块,然后,策略模块根据所述帧率信息确定所述目标应用程序的性能提升策略,最后,操作系统按照所述性能提升策略调整所述目标应用程序的系统资源的分配。由于场景数据包包括目标应用程序的应用标识和帧率信息,且场景数据包由正在运行的目标应用程序发送,操作系统能够根据应用标识准确确定适用的策略模块,进而调用该策略模块根据场景标识精细化确定目标应用程序的当前的内部运行场景的性能提升策略,并由操作系统按照该性能提升策略实时调整目标应用程序的系统资源的配置,实现了由操作系统对运行过程中的目标应用程序进行实时的性能优化,有利于提高移动终端控制目标应用程序进行性能优化的实时性和精确度。

[0089] 此外,通过划分多个策略模块能够实现对不同类型应用的精细化控制,且移动终

端无需同时加载全部策略模块,仅需要加载移动终端前台运行的目标应用程序对应的策略模块,有利于降低移动终端的资源占用,提高资源利用率和控制效率。

[0090] 与上述图2、图3、图4所示的实施例一致的,请参阅图5,图5是本申请实施例提供的一种移动终端的结构示意图,该移动终端运行有一个或多个应用程序和操作系统,如图所示,该移动终端包括处理器、存储器、通信接口以及一个或多个程序,其中,上述一个或多个程序不同于上述一个或多个应用程序,且上述一个或多个程序被存储在上述存储器中,并且被配置由上述处理器执行,上述程序包括用于执行以下步骤的指令:

[0091] 控制上述管理模块接收正在运行的目标应用程序发送的场景数据包,所述场景数据包包括目标应用程序的应用标识和帧率信息;

[0092] 控制上述管理模块确定所述多个策略模块中与所述应用标识对应的策略模块;

[0093] 控制上述策略模块根据所述帧率信息确定所述目标应用程序的性能提升策略;

[0094] 按照所述性能提升策略调整所述目标应用程序的系统资源的分配。

[0095] 可以看出,本申请实施例中,移动终端的操作系统中的管理模块首先接收正在运行的目标应用程序发送的场景数据包,所述场景数据包包括目标应用程序的应用标识和帧率信息,其次,管理模块确定所述多个策略模块中与所述应用标识对应的策略模块,然后,策略模块根据所述帧率信息确定所述目标应用程序的性能提升策略,最后,操作系统按照所述性能提升策略调整所述目标应用程序的系统资源的分配。由于场景数据包包括目标应用程序的应用标识和帧率信息,且场景数据包由正在运行的目标应用程序发送,操作系统能够根据应用标识准确确定适用的策略模块,进而调用该策略模块根据场景标识精细化确定目标应用程序的当前的内部运行场景的性能提升策略,并由操作系统按照该性能提升策略实时调整目标应用程序的系统资源的配置,实现了由操作系统对运行过程中的目标应用程序进行实时的性能优化,有利于提高移动终端控制目标应用程序进行性能优化的实时性和精确度。

[0096] 在一个可能的示例中,在所述策略模块根据所述帧率信息确定所述目标应用程序的性能提升策略方面,上述程序中的指令具体用于执行以下操作:控制所述策略模块确定所述帧率信息对应的待调整的系统资源;以及控制所述策略模块根据所述待调整的系统资源和所述帧率信息确定所述目标应用程序的性能提升策略。

[0097] 在一个可能的示例中,所述策略模块根据所述待调整的系统资源和所述帧率信息确定所述目标应用程序的性能提升策略,包括:所述策略模块根据所述帧率信息确定所述目标应用程序当前界面的帧率值;所述策略模块在检测到所述帧率值低于预设帧率值时,确定所述帧率值和所述预设帧率值之间的差值;所述策略模块根据所述差值确定上调所述待调整的系统资源的调整量;所述策略模块根据所述调整量生成所述目标应用程序的性能提升策略。

[0098] 在一个可能的示例中,所述策略模块根据所述待调整的系统资源和所述帧率信息确定所述目标应用程序的性能提升策略,包括:所述策略模块根据所述帧率信息确定所述目标应用程序当前界面的帧率值;所述策略模块确定所述帧率值所属的参考帧率区间;所述策略模块查找预设的帧率区间和待调整的系统资源的调整量之间的映射关系,确定所述参考帧率区间对应的所述待调整的系统资源的调整量。所述策略模块根据所述调整量生成所述目标应用程序的性能提升策略。

[0099] 在一个可能的示例中,所述策略模块根据所述待调整的系统资源和所述帧率信息确定所述目标应用程序的性能提升策略,包括:所述策略模块根据所述帧率信息确定所述目标应用程序当前界面的帧率值;所述策略模块在检测到所述帧率值高于预设帧率值时,确定所述帧率值和所述预设帧率值之间的差值;所述策略模块根据所述差值确定下调所述待调整的系统资源的调整量;所述策略模块根据所述调整量生成所述目标应用程序的性能提升策略。

[0100] 在一个可能的示例中,所述目标应用程序包括数据连接模块,所述操作系统包括数据连接管理模块,所述数据连接管理模块与所述数据连接模块建立预设数据通道;所述管理模块接收正在运行的目标应用程序发送的场景数据包,包括:所述管理模块通过所述预设数据通道接收所述目标应用程序发送的场景数据包。

[0101] 在一个可能的示例中,所述管理模块接收正在运行的所述目标应用程序发送的场景数据包之前,所述方法还包括:所述管理模块向所述目标应用程序发送第一数据格式,所述第一数据格式用于所述目标应用程序转换所述场景数据包的所述第二数据格式为所述第一数据格式。

[0102] 在一个可能的示例中,上述资源包括上述移动终端的以下资源中的至少一种:CPU资源、GPU资源、内存带宽资源、磁盘资源、网络资源。

[0103] 上述主要从方法侧执行过程的角度对本申请实施例的方案进行了介绍。可以理解的是,移动终端为了实现上述功能,其包含了执行各个功能相应的硬件结构和/或软件模块。本领域技术人员应该很容易意识到,结合本文中所公开的实施例描述的各示例的单元及算法步骤,本申请能够以硬件或硬件和计算机软件的结合形式来实现。某个功能究竟以硬件还是计算机软件驱动硬件的方式来执行,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。

[0104] 本申请实施例可以根据上述方法示例对移动终端进行功能单元的划分,例如,可以对应各个功能划分各个功能单元,也可以将两个或两个以上的功能集成在一个处理单元中。上述集成的单元既可以采用硬件的形式实现,也可以采用软件功能单元的形式实现。需要说明的是,本申请实施例中对单元的划分是示意性的,仅仅为一种逻辑功能划分,实际实现时可以有另外的划分方式。

[0105] 在采用集成的单元的情况下,图6示出了上述实施例中所涉及的资源配置装置的一种可能的功能单元组成框图。资源配置装置600应用于移动终端,上述移动终端上运行有操作系统以及一个或多个应用程序,上述资源配置装置600包括:处理单元602和通信单元603。处理单元602用于对资源配置装置的动作进行控制管理,例如,处理单元602用于支持资源配置装置执行图2中的步骤S201-S203、图3中的步骤S301-S307、图4中的步骤S401-S408和/或用于本文所描述的技术的其它过程。资源配置装置还可以包括存储单元601,用于存储程序代码和数据。

[0106] 上述处理单元602,用于控制上述管理模块通过上述通信单元603接收正在运行的目标应用程序发送的场景数据包,所述场景数据包包括目标应用程序的应用标识和帧率信息;以及用于控制所述管理模块确定所述多个策略模块中与所述应用标识对应的策略模块;以及用于控制所述策略模块根据所述帧率信息确定所述目标应用程序的性能提升策

略;以及用于控制所述操作系统按照所述性能提升策略调整所述目标应用程序的系统资源的分配。

[0107] 在一个可能的示例中,在所述策略模块根据所述帧率信息确定所述目标应用程序的性能提升策略方面,上述处理单元602具体用于:控制所述策略模块确定所述帧率信息对应的待调整的系统资源;以及控制所述策略模块根据所述待调整的系统资源和所述帧率信息确定所述目标应用程序的性能提升策略。

[0108] 在一个可能的示例中,在所述策略模块根据所述待调整的系统资源和所述帧率信息确定所述目标应用程序的性能提升策略方面,上述处理单元602具体用于:控制所述策略模块根据所述帧率信息确定所述目标应用程序当前界面的帧率值;以及控制所述策略模块在检测到所述帧率值低于预设帧率值时,确定所述帧率值和所述预设帧率值之间的差值;以及控制所述策略模块根据所述差值确定上调所述待调整的系统资源的调整量;以及控制所述策略模块根据所述调整量生成所述目标应用程序的性能提升策略。

[0109] 在一个可能的示例中,在所述策略模块根据所述待调整的系统资源和所述帧率信息确定所述目标应用程序的性能提升策略方面,上述处理单元602具体用于:控制所述策略模块根据所述帧率信息确定所述目标应用程序当前界面的帧率值;以及控制所述策略模块确定所述帧率值所属的参考帧率区间;以及控制所述策略模块查找预设的帧率区间和待调整的系统资源的调整量之间的映射关系,确定所述参考帧率区间对应的所述待调整的系统资源的调整量。以及控制所述策略模块根据所述调整量生成所述目标应用程序的性能提升策略。

[0110] 在一个可能的示例中,在所述策略模块根据所述待调整的系统资源和所述帧率信息确定所述目标应用程序的性能提升策略方面,上述处理单元602具体用于:控制所述策略模块根据所述帧率信息确定所述目标应用程序当前界面的帧率值;控制所述策略模块在检测到所述帧率值高于预设帧率值时,确定所述帧率值和所述预设帧率值之间的差值;以及控制所述策略模块根据所述差值确定下调所述待调整的系统资源的调整量;以及控制所述策略模块根据所述调整量生成所述目标应用程序的性能提升策略。

[0111] 在一个可能的示例中,所述目标应用程序包括数据连接模块,所述操作系统包括数据连接管理模块,所述数据连接管理模块与所述数据连接模块建立预设数据通道;在所述管理模块接收正在运行的目标应用程序发送的场景数据包方面,上述处理单元602具体用于:控制所述管理模块通过所述预设数据通道接收所述目标应用程序发送的场景数据包。

[0112] 在一个可能的示例中,所述管理模块通过上述通信单元603接收正在运行的所述目标应用程序发送的场景数据包之前,上述处理单元602还用于控制所述管理模块向所述目标应用程序发送第一数据格式,所述第一数据格式用于所述目标应用程序转换所述场景数据包的所述第二数据格式为所述第一数据格式。

[0113] 其中,上述目标应用程序包括数据连接模块,上述操作系统包括数据连接管理模块;上述资源配置装置还包括通道建立单元,上述通道建立单元,用于控制数据连接管理模块与上述数据连接模块建立上述预设数据通道。

[0114] 在一个可能的示例中,上述资源包括上述移动终端的以下资源中的至少一种:CPU资源、GPU资源、内存带宽资源、磁盘资源、网络资源。

[0115] 其中,处理单元602可以是处理器或控制器,通信单元603可以是处理器和程序运行空间之间的内部通信接口,如处理器与操作系统空间的通信接口,或者处理器与用户空间的通信接口等,存储单元601可以是存储器。

[0116] 请参阅图7,图7是本申请实施例提供了一种智能手机700的结构示意图,上述智能手机700包括:壳体710、触控显示屏720、主板730、电池740和副板750,主板730上设置有前置摄像头731、处理器732、存储器733、电源管理芯片734等,副板上设置有振子751、一体音腔752、VOOC闪充接口753和指纹识别模组754。

[0117] 其中,该智能手机包括目标应用程序和操作系统,目标应用程序运行于用户空间,操作系统运行于操作系统空间,操作系统包括管理模块和多个策略模块,每个策略模块用于确定一个应用程序的性能提升策略。管理模块接收正在运行的目标应用程序发送的场景数据包,场景数据包包括目标应用程序的应用标识和帧率信息;管理模块确定多个策略模块中与应用标识对应的策略模块;策略模块根据帧率信息确定目标应用程序的性能提升策略;操作系统按照性能提升策略调整目标应用程序的系统资源的分配。

[0118] 上述处理器732是智能手机的控制中心,利用各种接口和线路连接整个智能手机的各个部分,通过运行或执行存储在存储器733内的软件程序和/或模块,以及调用存储在存储器733内的数据,执行智能手机的各种功能和处理数据,从而对智能手机进行整体监控。可选的,处理器732可包括一个或多个处理单元;优选的,处理器732可集成应用处理器和调制解调处理器,其中,应用处理器主要处理操作系统、用户界面和应用程序等,调制解调处理器主要处理无线通信。可以理解的是,上述调制解调处理器也可以不集成到处理器732中。该处理器732例如可以是中央处理器(Central Processing Unit,CPU),通用处理器,数字信号处理器(Digital Signal Processor,DSP),专用集成电路(Application-Specific Integrated Circuit,ASIC),现场可编程门阵列(Field Programmable Gate Array,FPGA)或者其他可编程逻辑器件、晶体管逻辑器件、硬件部件或者其任意组合。其可以实现或执行结合本申请公开内容所描述的各种示例性的逻辑方框,模块和电路。上述处理器也可以是实现计算功能的组合,例如包含一个或多个微处理器组合,DSP和微处理器的组合等等。

[0119] 上述存储器733可用于存储软件程序以及模块,处理器732通过运行存储在存储器733的软件程序以及模块,从而执行智能手机的各种功能应用以及数据处理。存储器733可主要包括存储程序区和存储数据区,其中,存储程序区可存储操作系统、至少一个功能所需的应用程序等;存储数据区可存储根据智能手机的使用所创建的数据等。此外,存储器733可以包括高速随机存取存储器,还可以包括非易失性存储器,例如至少一个磁盘存储器件、闪存器件、或其他易失性固态存储器件。该存储器733例如可以是随机存取存储器(Random Access Memory, RAM)、闪存、只读存储器(Read Only Memory, ROM)、可擦除可编程只读存储器(Erasable Programmable ROM, EPROM)、电可擦可编程只读存储器(Electrically EPROM, EEPROM)、寄存器、硬盘、移动硬盘、只读光盘(CD-ROM)或者本领域熟知的任何其它形式的存储介质。

[0120] 本申请实施例还提供一种计算机存储介质,其中,该计算机存储介质存储用于电子数据交换的计算机程序,该计算机程序使得计算机执行如上述方法实施例中记载的任一方法的部分或全部步骤,上述计算机包括移动终端。

[0121] 本申请实施例还提供一种计算机程序产品,上述计算机程序产品包括存储了计算机程序的非瞬时性计算机可读存储介质,上述计算机程序可操作来使计算机执行如上述方法实施例中记载的任一方法的部分或全部步骤。该计算机程序产品可以作为一个软件安装包,上述计算机包括移动终端。

[0122] 需要说明的是,对于前述的各方法实施例,为了简单描述,故将其都表述为一系列的动作组合,但是本领域技术人员应该知悉,本申请并不受所描述的动作顺序的限制,因为依据本申请,某些步骤可以采用其他顺序或者同时进行。其次,本领域技术人员也应该知悉,说明书中所描述的实施例均属于优选实施例,所涉及的动作和模块并不一定是本申请所必须的。

[0123] 在上述实施例中,对各个实施例的描述都各有侧重,某个实施例中未详述的部分,可以参见其他实施例的相关描述。

[0124] 在本申请所提供的几个实施例中,应该理解到,所揭露的装置,可通过其它的方式实现。例如,以上所描述的装置实施例仅仅是示意性的,例如上述单元的划分,仅仅为一种逻辑功能划分,实际实现时可以有另外的划分方式,例如多个单元或组件可以结合或者可以集成到另一个系统,或一些特征可以忽略,或不执行。另一点,所显示或讨论的相互之间的耦合或直接耦合或通信连接可以是通过一些接口,装置或单元的间接耦合或通信连接,可以是电性或其它的形式。

[0125] 上述作为分离部件说明的单元可以是或者也可以不是物理上分开的,作为单元显示的部件可以是或者也可以不是物理单元,即可以位于一个地方,或者也可以分布到多个网络单元上。可以根据实际的需要选择其中的部分或者全部单元来实现本实施例方案的目的。

[0126] 另外,在本申请各个实施例中的各功能单元可以集成在一个处理单元中,也可以是各个单元单独物理存在,也可以两个或两个以上单元集成在一个单元中。上述集成的单元既可以采用硬件的形式实现,也可以采用软件功能单元的形式实现。

[0127] 上述集成的单元如果以软件功能单元的形式实现并作为独立的产品销售或使用,可以存储在一个计算机可读存储器中。基于这样的理解,本申请的技术方案本质上或者说对现有技术做出贡献的部分或者该技术方案的全部或部分可以以软件产品的形式体现出来,该计算机软件产品存储在一个存储器中,包括若干指令用以使得一台计算机设备(可为个人计算机、服务器或者网络设备)执行本申请各个实施例上述方法的全部或部分步骤。而前述的存储器包括:U盘、只读存储器(ROM,Read-Only Memory)、随机存取存储器(RAM,Random Access Memory)、移动硬盘、磁碟或者光盘等各种可以存储程序代码的介质。

[0128] 本领域普通技术人员可以理解上述实施例的各种方法中的全部或部分步骤可以通过程序来指令相关的硬件来完成,该程序可以存储于一计算机可读存储器中,存储器可以包括:闪存盘、只读存储器(英文:Read-Only Memory,简称:ROM)、随机存取器(英文:Random Access Memory,简称:RAM)、磁盘或光盘等。

[0129] 以上对本申请实施例进行了详细介绍,本文中应用了具体个例对本申请的原理及实施方式进行了阐述,以上实施例的说明只是用于帮助理解本申请的方法及其核心思想;同时,对于本领域的一般技术人员,依据本申请的思想,在具体实施方式及应用范围上均会有改变之处,综上所述,本说明书内容不应理解为对本申请的限制。



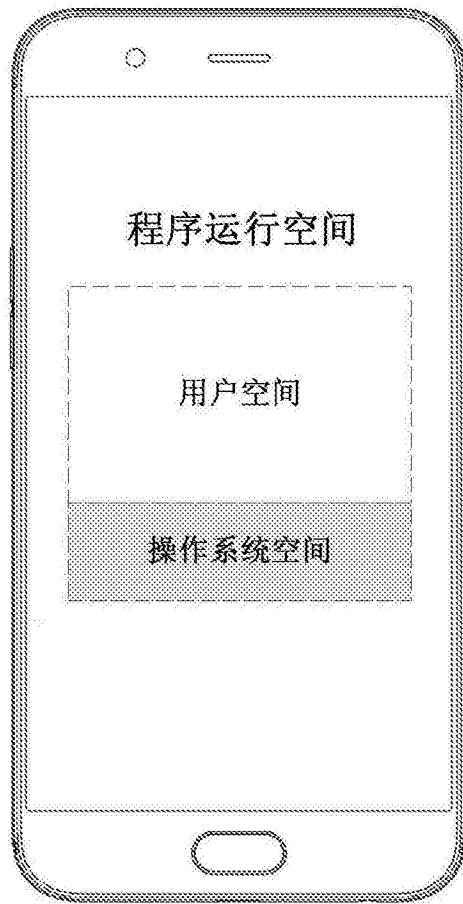
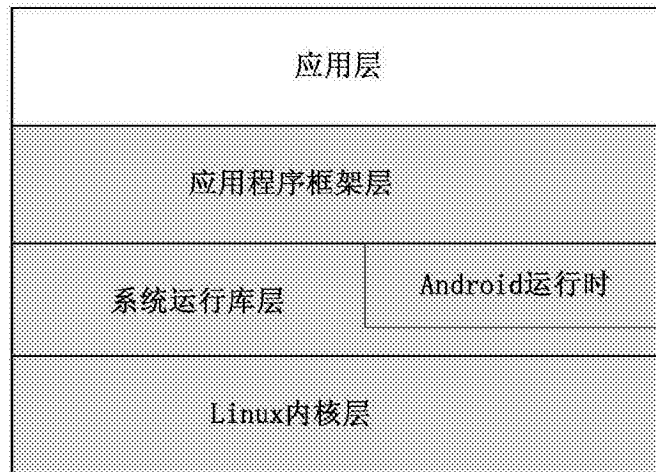


图1A



安卓系统

图1B

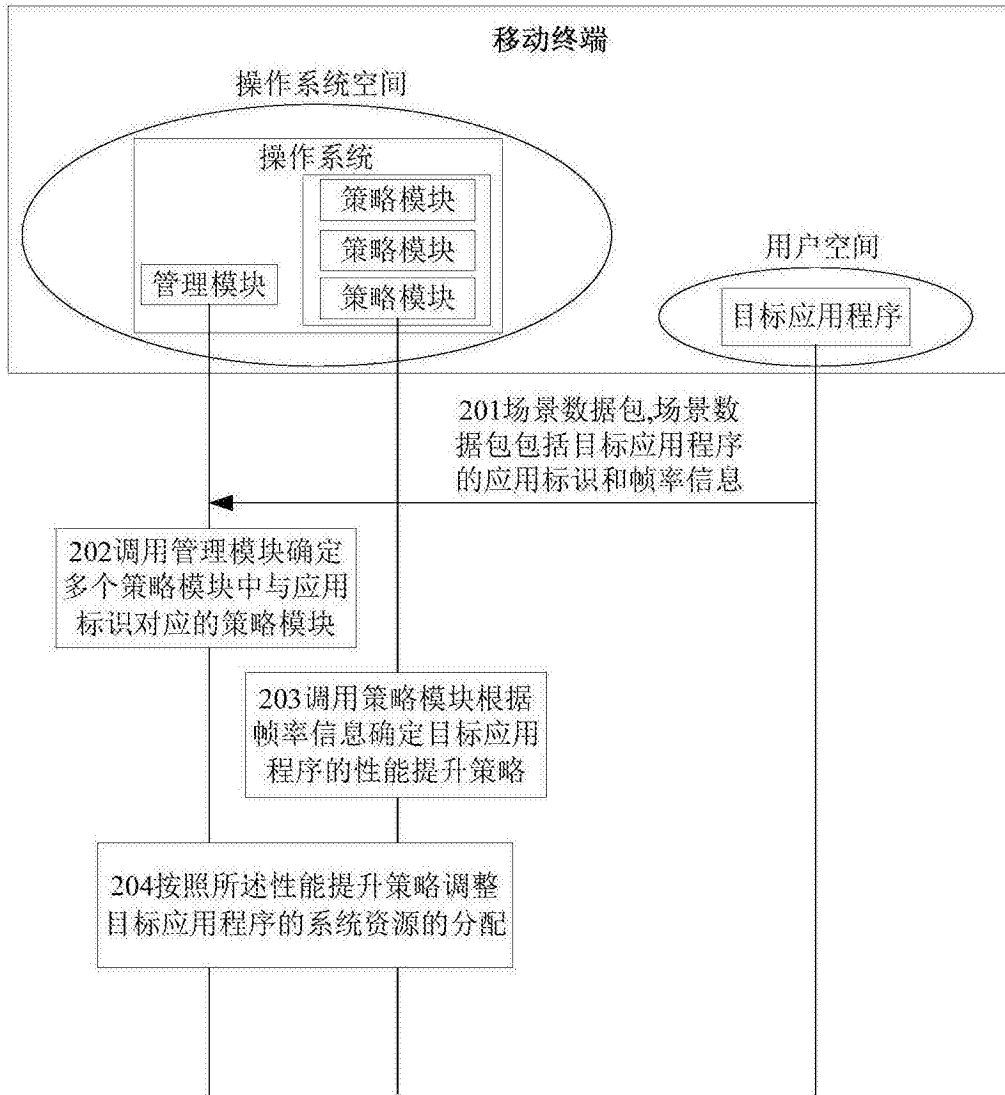


图2

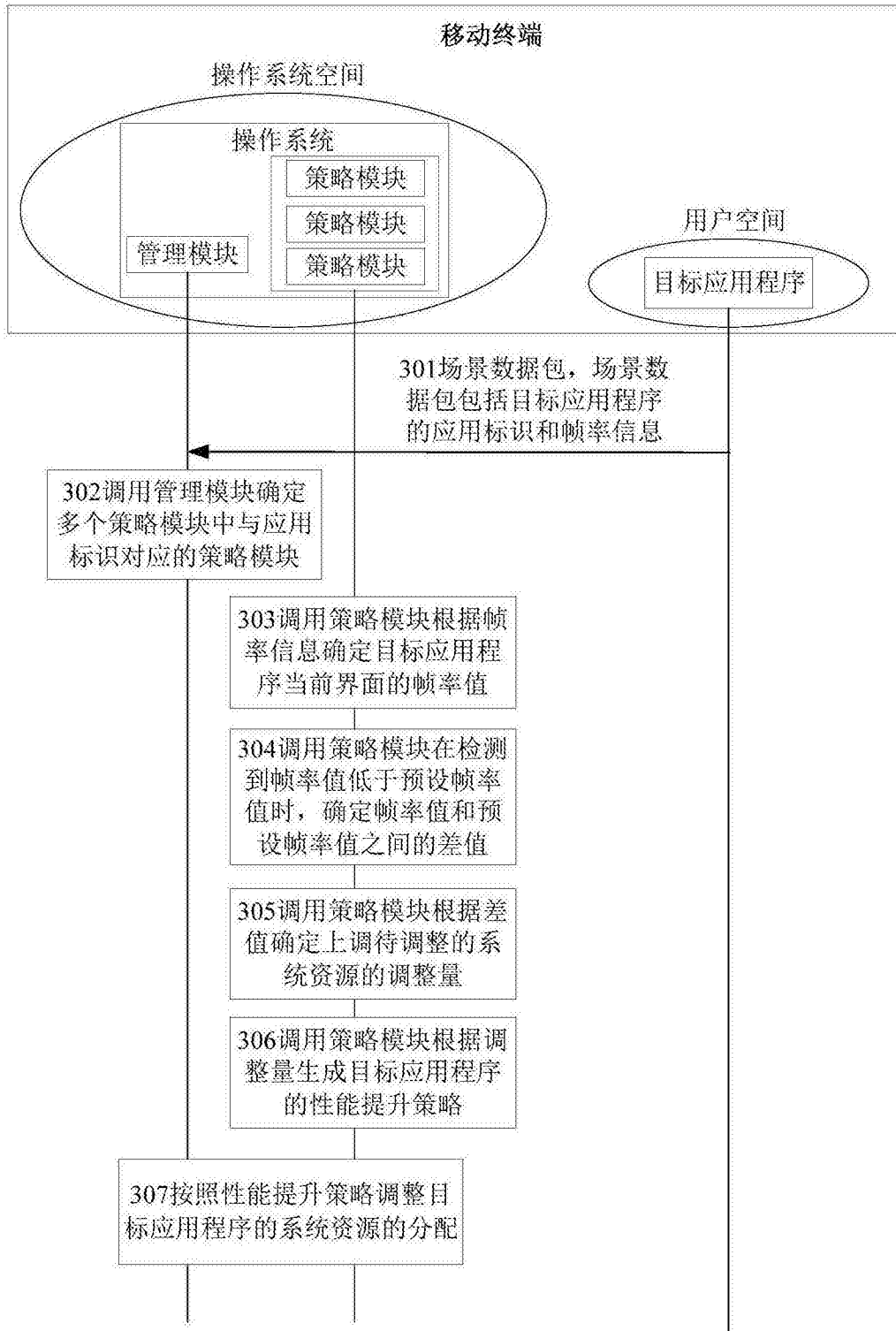


图3

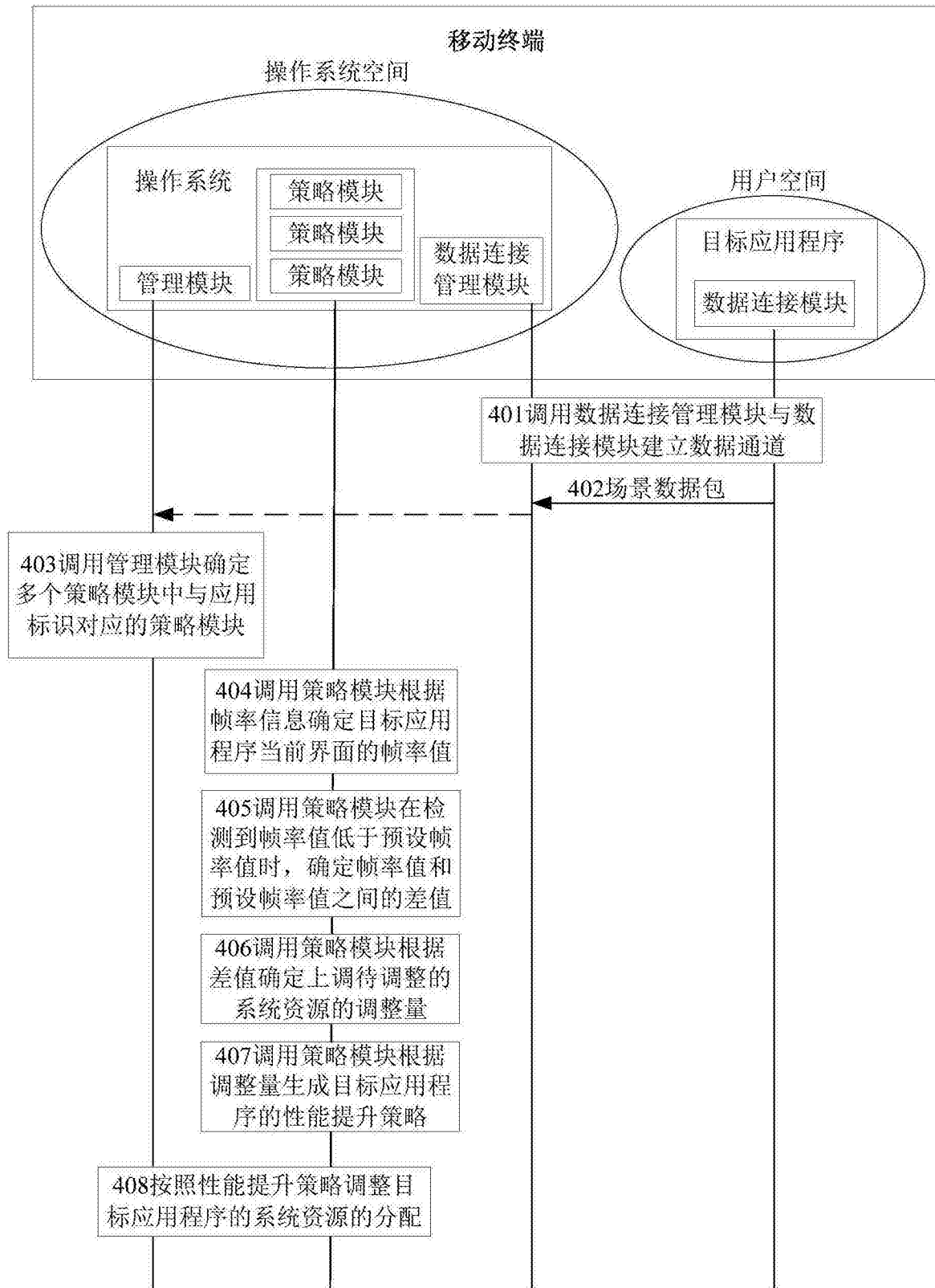


图4

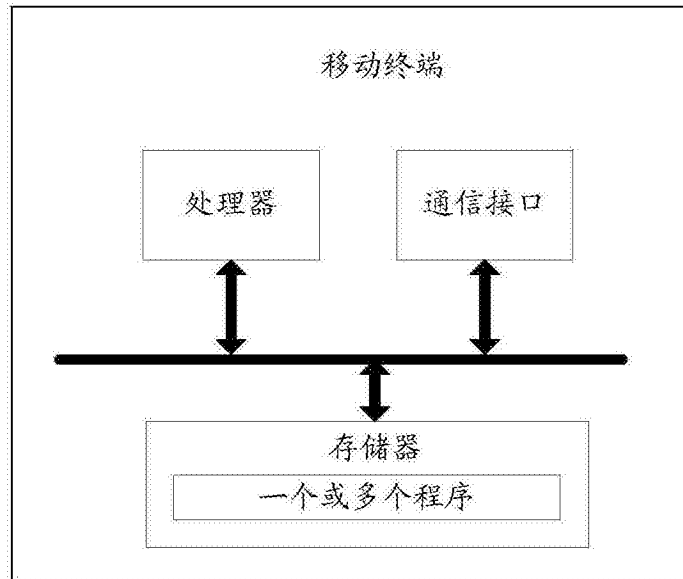


图5

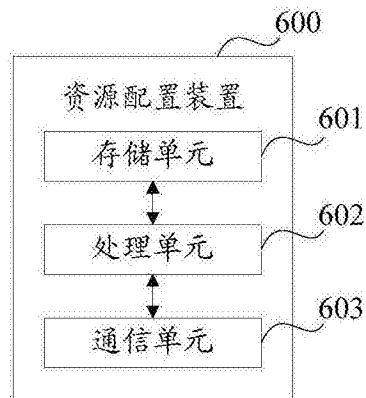


图6

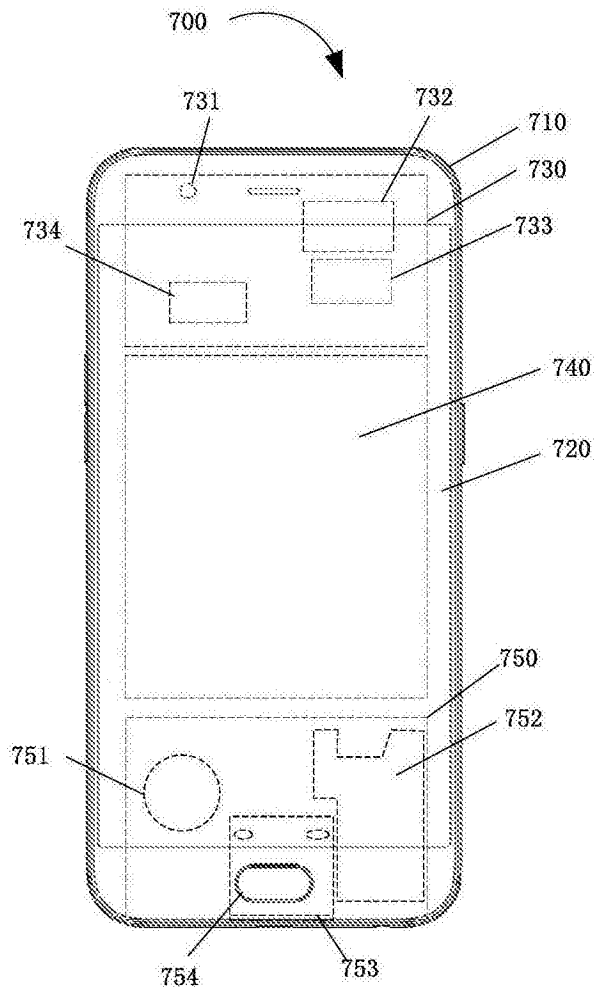


图7