



(12)发明专利申请

(10)申请公布号 CN 106484348 A

(43)申请公布日 2017. 03. 08

(21)申请号 201610833711.6

(22)申请日 2016.09.20

(71)申请人 TCL集团股份有限公司

地址 516006 广东省惠州市仲恺高新技术
开发区十九号小区

(72)发明人 卢伟超

(74)专利代理机构 深圳市君胜知识产权代理事
务所(普通合伙) 44268

代理人 王永文 刘文求

(51) Int. Cl.

G06F 3/14(2006.01)

G06F 9/54(2006.01)

G06T 13/00(2011.01)

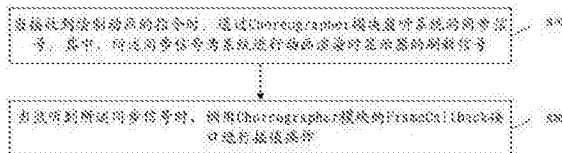
权利要求书2页 说明书5页 附图1页

(54)发明名称

一种基于同步信号的动画绘制方法及系统

(57)摘要

本发明公开了一种基于同步信号的动画绘制方法及系统,所述方法包括:当接收到绘制动画的指令时,通过Choreographer模块监听系统的同步信号,其中,所述同步信号为系统进行动画渲染时显示器的刷新信号;当监听到所述同步信号时,调用Choreographer模块的FrameCallback接口进行插值操作。本发明通过基于显示器刷新的同步信号进行插值操作,当系统卡顿时,系统仍然发送同步信息,系统根据所述同步信息进行插值操作,避免了由于系统出现卡顿而产生的插值不均匀,使得显示给用户的画面出现卡顿的问题。



1. 一种基于同步信号的动画绘制方法,其特征在于,其包括:

当接收到绘制动画的指令时,通过Choreographer模块监听系统的同步信号,其中,所述同步信号为系统进行动画渲染时显示器的刷新信号;

当监听到所述同步信号时,调用Choreographer模块的FrameCallback接口进行插值操作。

2. 根据权利要求1所述基于同步信号的动画绘制方法,其特征在于,所述当接收到绘制动画的指令时,通过Choreographer模块监听系统的同步信号之前包括:

当检测到终端设备启动应用程序时,获取Android系统的Choreographer模块,并将所述Choreographer模块配置于所述应用的主线程消息列队。

3. 根据权利要求1所述基于同步信号的动画绘制方法,其特征在于,所述当监听到所述同步信号时,调用Choreographer模块的FrameCallback接口进行插值操作之后还包括:

获取所述当前插值操作与上一次插值操作的时间间隔,并将所述时间间隔与预设时间段进行比较;

当所述时间间隔大于所述预设时间段时,调用Choreographer模块的FrameCallback接口进行预设次数插值操作。

4. 根据权利要求3所述基于同步信号的动画绘制方法,其特征在于,所述当所述时间间隔大于所述预设时间段时,调用Choreographer模块的FrameCallback接口进行预设次数插值操作具体为:

当所述时间间隔大于所述预设时间段时,根据预设时间段与插值次数的对应关系确定进行插值的次数;

调用Choreographer模块的FrameCallback接口进行预设次数的插值。

5. 根据权利要求1所述基于同步信号的动画绘制方法,其特征在于,所述FrameCallback接口预设配置循环插值操作指令。

6. 一种基于同步信号的动画绘制系统,其特征在于,其包括:

监听模块,用于当接收到绘制动画的指令时,通过Choreographer模块监听系统的同步信号,其中,所述同步信号为显示器刷新所产生的信号;

第一插值模块,用于当监听到所述同步信号时,调用Choreographer模块的FrameCallback接口进行插值操作。

7. 根据权利要求6所述基于同步信号的动画绘制系统,其特征在于,其还包括:

获取模块,用于当检测到终端设备启动应用程序时,获取Android系统的Choreographer模块,并将所述Choreographer模块配置于所述应用的主线程消息列队。

8. 根据权利要求6所述基于同步信号的动画绘制系统,其特征在于,其还包括:

比较模块,用于获取所述当前插值操作与上一次插值操作的时间间隔,并将所述时间间隔与预设时间段进行比较;

第二插值模块,用于当所述时间间隔大于所述预设时间段时,调用Choreographer模块的FrameCallback接口进行预设次数插值操作。

9. 根据权利要求8所述基于同步信号的动画绘制系统,其特征在于,所述第二插值模块具体包括:

确定单元,用于当所述时间间隔大于所述预设时间段时,根据预设时间段与插值

次数的对应关系确定进行插值的次数；

插值单元,用于调用Choreographer模块的FrameCallback接口进行预设次数的插值。

10.根据权利要求6所述基于同步信号的动画绘制系统,其特征在于,所述FrameCallback接口预设配置循环插值操作指令。

一种基于同步信号的动画绘制方法及系统

技术领域

[0001] 本发明涉及安卓系统技术领域,特别涉及一种基于同步信号的动画绘制方法及系统。

背景技术

[0002] 随着互联网技术的方案,形形色色的电子设备越来越受到消费者青睐。而在这些电子系统中,安卓(Android)系统是使用最为广泛的一种系统。安卓系统在绘制动画时普遍采用基于时间的插值方法进行动画插值。所述基于时间的插值方法指的是在系统内按照相同的时间间隔进行均匀插值。但是,所述基于时间的插值方法,当系统出现卡顿时,系统内运行的插值操作也会相应的出现卡顿不能均匀插值,而使得显示给用户的画面出现卡顿的问题。

[0003] 因而现有技术还有待改进和提高。

发明内容

[0004] 本发明要解决的技术问题在于,针对现有技术的不足,提供一种基于同步信号的动画绘制方法,以解决现有动画绘制中采用基于时间插值方法存在的由于系统出现卡顿而产生的插值不均匀,使得显示给用户的画面出现卡顿的问题。

[0005] 为了解决上述技术问题,本发明所采用的技术方案如下:

一种基于同步信号的动画绘制方法,其包括:

当接收到绘制动画的指令时,通过Choreographer模块监听系统的同步信号,其中,所述同步信号为系统进行动画渲染时显示器的刷新信号;

当监听到所述同步信号时,调用Choreographer模块的FrameCallback接口进行插值操作。

[0006] 所述基于同步信号的动画绘制方法,其中,所述当接收到绘制动画的指令时,通过Choreographer模块监听系统的同步信号之前包括:

当检测到终端设备启动应用程序时,获取Android系统的Choreographer模块,并将所述Choreographer模块配置于所述应用的主线程消息列队。

[0007] 所述基于同步信号的动画绘制方法,其中,所述当监听到所述同步信号时,调用Choreographer模块的FrameCallback接口进行插值操作之后还包括:

获取所述当前插值操作与上一次插值操作的时间间隔,并将所述时间间隔与预设时间段进行比较;

当所述时间间隔大于所述预设时间段时,调用Choreographer模块的FrameCallback接口进行预设次数插值操作。

[0008] 所述基于同步信号的动画绘制方法,其中,所述当所述时间间隔大于所述预设时间段时,调用Choreographer模块的FrameCallback接口进行预设次数插值操作具体为:

当所述时间间隔大于所述预设时间段时,根据预设时间段与插值次数的对应关系

确定进行插值的次数；

调用Choreographer模块的FrameCallback接口进行预设次数的插值。

[0009] 所述基于同步信号的动画绘制方法,其中,所述FrameCallback接口预配置循环插值操作指令。

[0010] 一种基于同步信号的动画绘制系统,其包括:

监听模块,用于当接收到绘制动画的指令时,通过Choreographer模块监听系统的同步信号,其中,所述同步信号为显示器刷新所产生的信号;

插值模块,用于当监听到所述同步信号时,调用Choreographer模块的FrameCallback接口进行插值操作。

[0011] 所述基于同步信号的动画绘制系统,其还包括:

获取模块,用于当检测到终端设备启动应用程序时,获取Android系统的Choreographer模块,并将所述Choreographer模块配置于所述应用的主线程消息队列。

[0012] 所述基于同步信号的动画绘制系统,其还包括:

比较模块,用于获取所述当前插值操作与上一次插值操作的时间间隔,并将所述时间间隔与预设时间段进行比较;

第二插值模块,用于当所述时间间隔大于所述预设时间段时,调用Choreographer模块的FrameCallback接口进行预设次数插值操作。

[0013] 所述基于同步信号的动画绘制系统,其中,所述第二插值模块具体包括:

确定单元,用于当所述时间间隔大于所述预设时间段时,根据预设时间段与插值次数的对应关系确定进行插值的次数;

插值单元,用于调用Choreographer模块的FrameCallback接口进行预设次数的插值。

[0014] 所述基于同步信号的动画绘制系统,其中,所述FrameCallback接口预配置循环插值操作指令。

[0015] 有益效果:与现有技术相比,本发明提供了一种基于同步信号的动画绘制方法及系统,所述方法包括:当接收到绘制动画的指令时,通过Choreographer模块监听系统的同步信号,其中,所述同步信号为系统进行动画渲染时显示器的刷新信号;当监听到所述同步信号时,调用Choreographer模块的FrameCallback接口进行插值操作。本发明通过基于显示器刷新的同步信号进行插值操作,当系统卡顿时,系统仍然发送同步信息,系统根据所述同步信息进行插值操作,避免了由于系统出现卡顿而产生的插值不均匀,使得显示给用户的画面出现卡顿的问题。

附图说明

[0016] 图1为本发明提供的基于同步信号的动画绘制较佳实施的流程图。

[0017] 图2为本发明提供的基于同步信号的动画绘制系统的结构原理图。

具体实施方式

[0018] 本发明提供一种基于同步信号的动画绘制方法及系统,为使本发明的目的、技术方案及效果更加清楚、明确,以下参照附图并举实施例对本发明进一步详细说明。应当理解,此处所描述的具体实施例仅用以解释本发明,并不用于限定本发明。

[0019] 本发明中,使用用于表示元件的诸如“模块”、“部件”或“单元”的后缀仅为了有利于本发明的说明,其本身并没有特定的意义。因此,“模块”、“部件”或“单元”可以混合地使用。

[0020] 本发明所述方法可以用于具有Android系统的终端设备,所述终端设备可以以各种形式来实施。例如,本发明中描述的终端可以包括诸如移动电话、智能电话、笔记本电脑、数字广播接收器、PDA(个人数字助理)、PAD(平板电脑)、PMP(便携式多媒体播放器)、导航装置等等的移动终端以及诸如数字TV、台式计算机等等的固定终端。然而,本领域技术人员将理解的是,除了特别用于移动目的的元件之外,根据本发明的实施方式的构造也能够应用于固定类型的终端。

[0021] 下面结合附图,通过对实施例的描述,对发明内容作进一步说明。

[0022] 请参照图1,图1为本发明提供的基于同步信号的动画绘制方法的较佳实施例的流程图。所述方法包括:

S100、当接收到绘制动画的指令时,通过Choreographer模块监听系统的同步信号,其中,所述同步信号为系统进行动画渲染时显示器的刷新信号。

[0023] 具体地,所述Choreographer模块为一个消息处理器,用于根据同步信号(vsync, vertical synchronization)执行回调算法。所述回调算法包括事件回调方法、动画回调方法和绘制回调方法。在本实施例中,当加入动画时将加入动画的动作发送给Choreographer模块,Choreographer模块根据加入动画的动作开启接收同步信息。其中,所述同步信号vsync与显示器的帧数同步,也就是说所述同步信号为显示器刷新所产生的信号。

[0024] 所述当接收到绘制动画的指令时值得是终端设备启动动画绘制功能,例如,启动需要绘制动画的应用APP、终端本身需要绘制动画等。

[0025] 在本实施例中,由于终端设备装载的APP不能直接调用系统的Choreographer模块,从而在终端设备启动APP时包括一个调用步骤,具体可以为:

当终端设备启动时,监听终端设备启动应用程序操作;

当监听到终端设备启动应用程序的操作时,获取系统的Choreographer模块,并将所述Choreographer模块配置于所述应用的主线程消息列队。

[0026] 具体地,当终端设备启动时,在终端启动的所有应用的主线程消息列队中添加所述Choreographer模块,用于当应用程序进行动画绘制时通过Choreographer模块接收同步信号,根据所述同步信息进行动画插值。这样可以保证终端设备安装的所用应用程序动画插值与显示器的帧数同步,避免了由于系统出现卡顿而产生的插值不均匀,使得显示给用户的画面出现卡顿的问题。

[0027] S200、当监听到所述同步信号时,调用Choreographer模块的FrameCallback接口进行插值操作。

[0028] 具体地,所述FrameCallback接口为Choreographer模块提供的提供的回调方法类;用于根据同步信号采用回调方法进行动画插值。也就是说,当接收到同步信号时,通过Choreographer模块的FrameCallback接口,根据同步信号采用回调方法进行动画插值。所述回调方法是采用安卓系统的消息机制,即提前将所述回调方法设置给应用的主线程的消息列队,当接收同步信号时,回调所述FrameCallback接口。

[0029] 所述FrameCallback接口预设配置循环插值操作指令,来实现多少插值,解决所述

FrameCallback配置所述回调方法时仅能进行一次插值操作,不能重复调用多次才可插值完成的问题。在本实施例中,所述FrameCallback接口用递归的方式,即在FrameCallback中再配置一个FrameCallback,实现FrameCallback接口的循环配置,这样就保证了当插值没有完成时循环不退出,实现了重复调用FrameCallback接口,实现动画绘制过程中的多次插值操作。

[0030] 由于在调用FrameCallback接口时可能存在延迟或者漏调用的问题,从而在调用Choreographer模块的FrameCallback接口进行插值操作之后还可以包括:记录所述当前插值绘制的时间,判断预设时间内是否进行下一次插值操作,若预设时间内未进行下一次插值操作,则进行钳位处理。

[0031] 具体地,所述预设时间可以显示器刷新时间差,即显示器的上一帧与当前帧的时间差,也可以是大于显示器刷新时间差,如50毫秒,60毫秒等。在本实施例中,所述预设时间段可以为60毫秒。而在安卓系统中显示器刷新的帧频一般为60Hz,那么上一帧与当前帧的时间差为16.6毫秒。从而,在进行一次动画绘制后,获取所述当前插值操作与上一次插值操作的时间间隔,并将所述时间间隔与预设时间段进行比较;当所述时间间隔大于所述预设时间段时,调用Choreographer模块的FrameCallback接口进行预设次数插值操作。避免了由于FrameCallback接口时可能存在延迟或者漏调用,而产生的动画卡顿,给用户使用带来不变。

[0032] 示例性的,所述获取所述当前插值操作与上一次插值操作的时间间隔具体可以为当进行当前插值操作后,启动配置预设时间段的计时器,当计时器超时时,接收到同步信号时,控制FrameCallback接口进行预设次数的插值操作。

[0033] 所述预设次数为根据所述预设时间段预先设置的,如当预设时间段为60毫秒时,所述预设次数可以为3。也就是说,当所述时间间隔大于所述预设时间段时,调用Choreographer模块的FrameCallback接口进行3数插值操作。从而,所述当所述当所述时间间隔大于所述预设时间段时,调用Choreographer模块的FrameCallback接口进行预设次数插值操作具体为:当所述时间间隔大于所述预设时间段时,根据预设预设时间段与插值次数的对应关系确定进行插值的次数;调用Choreographer模块的FrameCallback接口进行预设次数的插值。这样可以避免由于FrameCallback接口的漏调等原因产生的插值不均匀或者卡顿问题。

[0034] 本发明还提供了一种基于同步信号的动画绘制系统,如图2所示,其包括:

监听模块100,用于当接收到绘制动画的指令时,通过Choreographer模块监听系统的同步信号,其中,所述同步信号为显示器刷新所产生的信号;

插值模块200,用于当监听到所述同步信号时,调用Choreographer模块的FrameCallback接口进行插值操作。

[0035] 所述基于同步信号的动画绘制系统,其还包括:

获取模块,用于当检测到终端设备启动应用程序时,获取Android系统的Choreographer模块,并将所述Choreographer模块配置于所述应用的主线程消息列队。

[0036] 所述基于同步信号的动画绘制系统,其还包括:

比较模块,用于获取所述当前插值操作与上一次插值操作的时间间隔,并将所述时间间隔与预设时间段进行比较;

第二插值模块,用于当所述时间间隔大于所述预设时间段时,调用Choreographer模块的FrameCallback接口进行预设次数插值操作。

[0037] 所述基于同步信号的动画绘制系统,其中,所述第二插值模块具体包括:

确定单元,用于当所述时间间隔大于所述预设时间段时,根据预设预设时间段与插值次数的对应关系确定进行插值的次数;

插值单元,用于调用Choreographer模块的FrameCallback接口进行预设次数的插值。

[0038] 所述基于同步信号的动画绘制系统,其中,所述FrameCallback接口预设配置循环插值操作指令。

[0039] 在本发明所提供的实施例中,应该理解到,所揭露的系统和方法,可以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,所述模块的划分,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,机械或其它的形式。

[0040] 所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。

[0041] 另外,在本发明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,也可以采用硬件加软件功能单元的形式实现。

[0042] 上述以软件功能单元的形式实现的集成的单元,可以存储在一个计算机可读取存储介质中。上述软件功能单元存储在一个存储介质中,包括若干指令用以使得一台计算机设备(可以是个人计算机,服务器,或者网络设备等)或处理器(processor)执行本发明各个实施例所述方法的部分步骤。而前述的存储介质包括:U盘、移动硬盘、只读存储器(Read-Only Memory,ROM)、随机存取存储器(Random Access Memory,RAM)、磁碟或者光盘等各种可以存储程序代码的介质。

[0043] 最后应说明的是:以上实施例仅用以说明本发明的技术方案,而非对其限制;尽管参照前述实施例对本发明进行了详细的说明,本领域的普通技术人员应当理解:其依然可以对前述各实施例所记载的技术方案进行修改,或者对其中部分技术特征进行等同替换;而这些修改或者替换,并不使相应技术方案的本质脱离本发明各实施例技术方案的精神和范围。

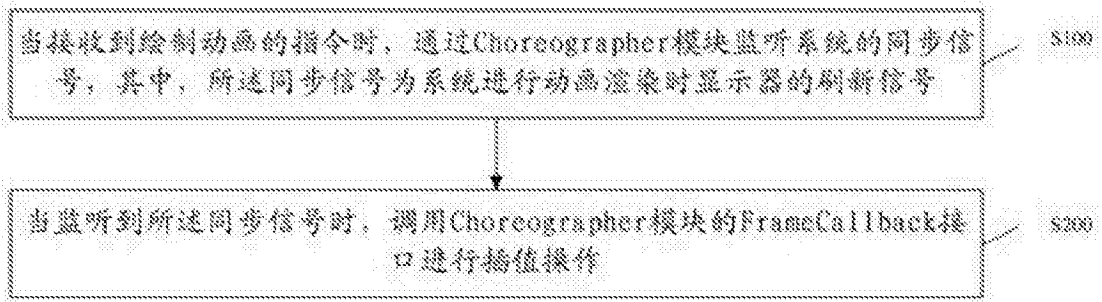


图1

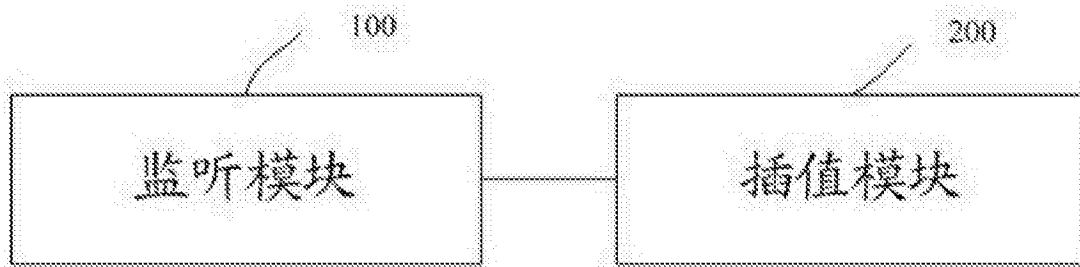


图2