



(12) 发明专利

(10) 授权公告号 CN 113254120 B

(45) 授权公告日 2022. 11. 01

(21) 申请号 202110363737.X

G06F 3/04847 (2022.01)

(22) 申请日 2021.04.02

G06F 3/0488 (2022.01)

(65) 同一申请的已公布的文献号  
申请公布号 CN 113254120 A

(56) 对比文件

CN 102681810 A, 2012.09.19

CN 104620595 A, 2015.05.13

(43) 申请公布日 2021.08.13

CN 110771160 A, 2020.02.07

(73) 专利权人 荣耀终端有限公司  
地址 518040 广东省深圳市福田区香蜜湖  
街道东海社区红荔西路8089号深业中  
城6号楼A单元3401

审查员 郭春羽

(72) 发明人 孙文涌 忻振文

(74) 专利代理机构 北京同立钧成知识产权代理  
有限公司 11205

专利代理师 余娜 臧建明

(51) Int. Cl.

G06F 9/451 (2018.01)

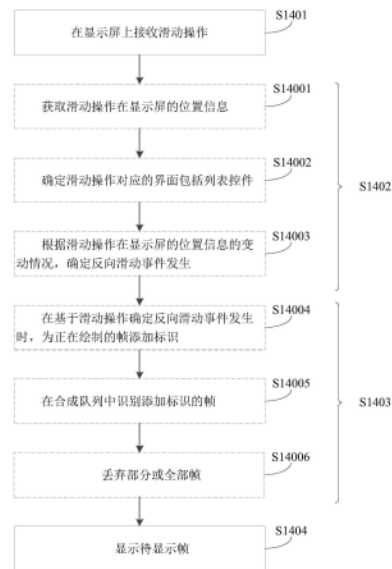
权利要求书6页 说明书27页 附图12页

(54) 发明名称

数据处理方法和相关装置

(57) 摘要

本申请实施例提供一种数据处理方法和相关装置,应用于终端技术领域。该方法包括:接收用户输入的滑动操作;确定滑动操作对应的控件为列表控件;确定反向滑动事件发生,反向滑动事件为滑动操作的滑动方向发生反向的事件;在反向滑动事件发生时,为第一帧添加标识,第一帧为反向滑动事件发生时正在绘制的帧;在缓存队列中存在携带标识的第一帧的情况下,丢弃缓存队列中的部分或全部帧,缓存队列用于存储绘制完成的帧,绘制完成的帧与滑动操作相对应;合成待显示帧,待显示帧为执行丢弃动作后得到的待显示的帧。这样,当终端设备接收到用户反向滑动操作时,可以通过丢弃缓存队列中的全部或部分帧,以降低反向滑动后界面显示的反应时延。



1. 一种数据处理方法,其特征在于,应用于终端设备,所述方法包括:

接收用户输入的滑动操作;

确定所述滑动操作对应的控件为列表控件;

确定反向滑动事件发生,所述反向滑动事件为所述滑动操作的滑动方向发生反向的事件;

在所述反向滑动事件发生时,为第一帧添加标识,所述第一帧为所述反向滑动事件发生时正在绘制的帧;所述帧是界面显示中最小单位的单幅画面;一帧为一副静止的画面;其中,帧绘制是指显示界面的图片绘制;所述显示界面由一个或多个视图组成,各个所述视图由视图系统的可视控件绘制,各个所述视图由子视图组成;帧渲染是将绘制后的视图进行着色操作或增加3D效果;帧合成是将多个经一个或多个渲染后的视图合成为所述显示界面;在缓存队列中存在携带所述标识的第一帧的情况下,丢弃所述缓存队列中的部分或全部帧,所述缓存队列用于存储绘制完成的帧,所述绘制完成的帧与所述滑动操作相对应;

合成待显示帧,所述待显示帧为执行所述丢弃动作后得到的待显示的帧;

显示所述待显示帧;

当所述终端设备丢弃的帧为所述缓存队列中的部分帧时,所述待显示帧包括丢弃所述缓存队列中的部分帧后,所述缓存队列中剩余的帧;

或者,当所述终端设备丢弃的帧为所述缓存队列中的全部帧时,所述待显示帧包括丢弃所述缓存队列中的全部帧后,所述缓存队列中新增加的渲染后的帧。

2. 根据权利要求1所述的方法,其特征在于,所述在缓存队列中存在携带所述标识的第一帧的情况下,丢弃所述缓存队列中的部分或全部帧,包括:

所述终端设备中的显示合成进程确认所述缓存队列接收到所述携带所述标识的第一帧时,丢弃所述缓存队列中部分或全部帧;

或者,所述显示合成进程从所述缓存队列获取待合成的帧时,丢弃所述缓存队列中部分或全部帧。

3. 根据权利要求2所述的方法,其特征在于,所述显示合成进程从所述缓存队列获取待合成的帧时,丢弃所述缓存队列中部分或全部帧,具体为:

每当周期信号到来时,所述显示合成进程确定所述缓存队列中是否包含有携带所述标识的第一帧;

在所述缓存队列中包含有携带所述标识的第一帧的情况下,所述显示合成进程丢弃所述缓存队列中部分或全部帧。

4. 根据权利要求2所述的方法,其特征在于,所述丢弃所述缓存队列中的部分或全部帧,包括:

丢弃所述缓存队列中的M帧,其中,所述M小于或等于N,所述N为预设的常量;

或者,丢弃所述缓存队列中的位于所述携带所述标识的第一帧之前的帧;

或者,丢弃所述携带所述标识的第一帧以及所述缓存队列中的位于所述携带所述标识的第一帧之前的帧;

或者,丢弃所述缓存队列中的全部帧。

5. 根据权利要求1-4任一项所述的方法,其特征在于,所述终端设备包括图像绘制模块、图像渲染模块和图像合成模块,所述方法还包括:

响应于所述滑动操作,所述图像绘制模块对所述滑动操作对应的帧进行绘制;  
所述图像渲染模块对绘制后的帧进行渲染;  
所述图像渲染模块将渲染后的帧存储在所述缓存队列;  
每当周期信号到来时,显示合成进程从所述缓存队列中选择第二帧;  
所述显示合成进程通过所述图像合成模块对所述第二帧进行合成。

6. 根据权利要求1-4任一项所述的方法,其特征在于,确定所述滑动操作对应的控件为列表控件之后,所述方法还包括:

确定所述列表控件的方向属性,所述列表控件的方向属性包括:横向和纵向。

7. 根据权利要求6所述的方法,其特征在于,所述终端设备包括视图系统,所述确定所述滑动操作对应的控件为列表控件,包括:

所述视图系统确定所述滑动操作对应的对象的实例化为列表控件;

所述确定所述列表控件的方向属性,包括:

所述视图系统根据所述对象使用的变量值或接口函数的返回值,确定所述列表控件的方向属性为横向或纵向;其中,所述接口函数包括getOrientation(),所述变量值包括mOrientation。

8. 根据权利要求6所述的方法,其特征在于,所述确定反向滑动事件发生,包括:

在所述列表控件的方向属性为横向时,根据所述滑动操作产生的横向位移确定所述滑动操作是否发生反向;

或者,在所述列表控件的方向属性为纵向时,根据所述滑动操作产生的纵向位移确定所述滑动操作是否发生反向。

9. 根据权利要求6所述的方法,其特征在于,所述确定反向滑动事件发生,包括:

在所述列表控件的方向属性为横向时,计算所述滑动操作产生的横向位移是否发生由负值变为正值,或由正值变为负值的变化;当所述滑动操作产生的横向位移发生由负值变为正值,或由正值变为负值的变化时,确定所述反向滑动事件发生;

或者,在所述列表控件的方向属性为纵向时,计算所述滑动操作产生的纵向位移是否发生由负值变为正值,或由正值变为负值的变化;当所述滑动操作产生的纵向位移发生由负值变为正值,或由正值变为负值的变化时,确定所述反向滑动事件发生。

10. 根据权利要求7所述的方法,其特征在于,所述视图系统还用于计算所述滑动操作的横向位移或纵向位移。

11. 根据权利要求1-4任一项所述的方法,其特征在于,所述列表控件为有序内容的集合,用于横向或纵向滑动,所述列表控件包括下述中的任一项:应用界面中的控件、设置界面中的控件、电子书阅读界面中的控件或文档界面中的控件。

12. 一种数据处理方法,其特征在于,应用于终端设备,所述方法包括:

接收用户输入的滑动操作;

确定反向滑动事件发生,所述反向滑动事件为所述滑动操作的滑动方向发生反向的事件;

在缓存队列中存在第一帧的情况下,丢弃所述缓存队列中的部分或全部帧,以降低反向滑动后界面显示的反应时延;其中,所述第一帧为所述反向滑动事件发生时正在绘制的帧,所述帧是界面显示中最小单位的单幅画面;一帧为一副静止的画面;所述缓存队列用于

存储绘制完成的帧,所述绘制完成的帧与所述滑动操作相对应,帧绘制是指显示界面的图片绘制;合成待显示帧,显示所述待显示帧,所述待显示帧为执行所述丢弃动作后得到的待显示的帧;当所述终端设备丢弃的帧为所述缓存队列中的部分帧时,所述待显示帧包括丢弃所述缓存队列中的部分帧后,所述缓存队列中剩余的帧;或者,当所述终端设备丢弃的帧为所述缓存队列中的全部帧时,所述待显示帧包括丢弃所述缓存队列中的全部帧后,所述缓存队列中新增加的渲染后的帧;帧渲染是将绘制后的视图进行着色操作或增加3D效果;帧合成是将多个经一个或多个渲染后的视图合成为所述显示界面。

13. 根据权利要求12所述的方法,其特征在于,所述在缓存队列中存在第一帧的情况下,丢弃所述缓存队列中的部分或全部帧,包括:

所述终端设备中的显示合成进程确认所述缓存队列接收到所述第一帧时,丢弃所述缓存队列中部分或全部帧;

或者,所述显示合成进程从所述缓存队列获取待合成的帧时,丢弃所述缓存队列中部分或全部帧。

14. 根据权利要求13所述的方法,其特征在于,所述显示合成进程从所述缓存队列获取待合成的帧时,丢弃所述缓存队列中部分或全部帧,具体为:

每当周期信号到来时,所述显示合成进程确定所述缓存队列中是否包含有所述第一帧;

当所述缓存队列中包含有第一帧的情况下,所述显示合成进程丢弃所述缓存队列中部分或全部帧。

15. 根据权利要求13或14所述的方法,其特征在于,所述丢弃所述缓存队列中的部分或全部帧,包括:

丢弃所述缓存队列中的M帧,其中,所述M小于或等于N,所述N为预设的常量;

或者,丢弃所述缓存队列中的位于所述第一帧之前的帧;

或者,丢弃所述第一帧以及所述缓存队列中的位于所述第一帧之前的帧;

或者,丢弃所述缓存队列中的全部帧。

16. 一种终端设备,其特征在于,所述终端设备包括:

触摸屏,其中,所述触摸屏包括触控面板和显示屏;

一个或多个处理器;

存储器;

以及一个或多个计算机程序,其中所述一个或多个计算机程序被存储在所述存储器中,所述一个或多个计算机程序包括指令,当所述指令被所述终端设备执行时,使得所述终端设备执行以下步骤:

接收用户输入的滑动操作;

确定所述滑动操作对应的控件为列表控件;

确定反向滑动事件发生,所述反向滑动事件为所述滑动操作的滑动方向发生反向的事件;

在所述反向滑动事件发生时,为第一帧添加标识,所述第一帧为所述反向滑动事件发生时正在绘制的帧;所述帧是界面显示中最小单位的单幅画面;一帧为一副静止的画面;其中,帧绘制是显示界面的图片绘制;所述显示界面由一个或多个视图组成,各个所述视图由

视图系统的可视控件绘制,各个所述视图由子视图组成;帧渲染是将绘制后的视图进行着色操作或增加3D效果;帧合成是将多个经一个或多个渲染后的视图合成为所述显示界面;

在缓存队列中存在携带所述标识的第一帧的情况下,丢弃所述缓存队列中的部分或全部帧,所述缓存队列用于存储绘制完成的帧,所述绘制完成的帧与所述滑动操作相对应;

合成待显示帧,所述待显示帧为执行所述丢弃动作后得到的待显示的帧;

显示所述待显示帧;

当所述终端设备丢弃的帧为所述缓存队列中的部分帧时,所述待显示帧包括丢弃所述缓存队列中的部分帧后,所述缓存队列中剩余的帧;

或者,当所述终端设备丢弃的帧为所述缓存队列中的全部帧时,所述待显示帧包括丢弃所述缓存队列中的全部帧后,所述缓存队列中新增加的渲染后的帧。

17.根据权利要求16所述的终端设备,其特征在于,所述终端设备具体用于执行以下步骤:

确认所述缓存队列接收到所述携带所述标识的第一帧时,丢弃所述缓存队列中部分或全部帧;

或者,从所述缓存队列获取待合成的帧时,丢弃所述缓存队列中部分或全部帧。

18.根据权利要求17所述的终端设备,其特征在于,所述终端设备具体用于执行以下步骤:

每当在周期信号到来时,确定所述缓存队列中是否包含有携带所述标识的第一帧;

在所述缓存队列中包含有携带所述标识的第一帧的情况下,丢弃所述缓存队列中部分或全部帧。

19.根据权利要求17所述的终端设备,其特征在于,所述终端设备具体用于执行以下步骤:

丢弃所述缓存队列中的M帧,其中,所述M小于或等于N,所述N为预设的常量;

或者,丢弃所述缓存队列中的位于所述携带所述标识的第一帧之前的帧;

或者,丢弃所述携带所述标识的第一帧以及所述缓存队列中的位于所述携带所述标识的第一帧之前的帧;

或者,丢弃所述缓存队列中的全部帧。

20.根据权利要求16-19任一项所述的终端设备,其特征在于,所述终端设备包括图像绘制模块、图像渲染模块和图像合成模块,

所述终端设备还用于响应于所述滑动操作,所述图像绘制模块用于对所述滑动操作对应的帧进行绘制;

所述图像渲染模块用于对绘制后的帧进行渲染;

所述图像渲染模块用于将渲染后的帧存储在所述缓存队列;

每当周期信号到来时,所述终端设备用于从所述缓存队列中选择第二帧;

所述终端设备用于通过所述图像合成模块对所述第二帧进行合成。

21.根据权利要求16-19任一项所述的终端设备,其特征在于,所述终端设备还用于执行以下步骤:

确定所述滑动操作对应的控件为列表控件之后,确定所述列表控件的方向属性,所述列表控件的方向属性包括:横向和纵向。

22. 根据权利要求21所述的终端设备,其特征在于,所述终端设备包括视图系统,所述视图系统用于确定所述滑动操作对应的对象的实例化为列表控件;

所述视图系统用于根据所述对象使用的变量值或接口函数的返回值,确定所述列表控件的方向属性为横向或纵向;其中,所述接口函数包括getOrientation(),所述变量值包括mOrientation。

23. 根据权利要求21所述的终端设备,其特征在于,所述终端设备具体用于执行以下步骤:

在所述列表控件的方向属性为横向时,根据所述滑动操作产生的横向位移确定所述滑动操作是否发生反向;

或者,在所述列表控件的方向属性为纵向时,根据所述滑动操作产生的纵向位移确定所述滑动操作是否发生反向。

24. 根据权利要求21所述的终端设备,其特征在于,所述终端设备,具体用于执行以下步骤:

在所述列表控件的方向属性为横向时,计算所述滑动操作产生的横向位移是否发生由负值变为正值,或由正值变为负值的变化;当所述滑动操作产生的横向位移发生由负值变为正值,或由正值变为负值的变化时,确定所述反向滑动事件发生;

或者,在所述列表控件的方向属性为纵向时,计算所述滑动操作产生的纵向位移是否发生由负值变为正值,或由正值变为负值的变化;当所述滑动操作产生的纵向位移发生由负值变为正值,或由正值变为负值的变化时,确定所述反向滑动事件发生。

25. 根据权利要求22所述的终端设备,其特征在于,所述视图系统还用于计算所述滑动操作的横向位移或纵向位移。

26. 根据权利要求16-19任一项所述的终端设备,其特征在于,所述列表控件为有序内容的集合,用于横向或纵向滑动,所述列表控件包括下述中的任一项:应用界面中的控件、设置界面中的控件、电子书阅读界面中的控件或文档界面中的控件。

27. 一种终端设备,其特征在于,所述终端设备包括:

触摸屏,其中,所述触摸屏包括触控面板和显示屏;

一个或多个处理器;

存储器;

以及一个或多个计算机程序,其中所述一个或多个计算机程序被存储在所述存储器中,所述一个或多个计算机程序包括指令,当所述指令被所述终端设备执行时,使得所述终端设备执行以下步骤:

接收用户输入的滑动操作;

在缓存队列中存在第一帧的情况下,确定反向滑动事件发生,所述反向滑动事件为所述滑动操作的滑动方向发生反向的事件;

丢弃所述缓存队列中的部分或全部帧,以降低反向滑动后界面显示的反应时延;其中,所述第一帧为所述反向滑动事件发生时正在绘制的帧,所述帧是界面显示中最小单位的单幅画面;一帧为一副静止的画面;所述缓存队列用于存储绘制完成的帧,所述绘制完成的帧与所述滑动操作相对应,帧绘制是指显示界面的图片绘制;

合成待显示帧,显示所述待显示帧,所述待显示帧为执行所述丢弃动作后得到的待显

示的帧；当所述终端设备丢弃的帧为所述缓存队列中的部分帧时，所述待显示帧包括丢弃所述缓存队列中的部分帧后，所述缓存队列中剩余的帧；或者，当所述终端设备丢弃的帧为所述缓存队列中的全部帧时，所述待显示帧包括丢弃所述缓存队列中的全部帧后，所述缓存队列中新增加的渲染后的帧；帧渲染是将绘制后的视图进行着色操作或增加3D效果；帧合成是将多个经一个或多个渲染后的视图合成为所述显示界面。

28. 根据权利要求27所述的终端设备，其特征在于，所述终端设备具体用于执行以下步骤：

确认所述缓存队列接收到所述第一帧时，丢弃所述缓存队列中部分或全部帧；  
或者，从所述缓存队列获取待合成的帧时，丢弃所述缓存队列中部分或全部帧。

29. 根据权利要求28所述的终端设备，其特征在于，所述终端设备具体用于执行以下步骤：

每当周期信号到来时，确定所述缓存队列中是否包含有所述第一帧；  
在所述缓存队列中包含有所述第一帧的情况下，丢弃所述缓存队列中部分或全部帧。

30. 根据权利要求28或29所述的终端设备，其特征在于，所述终端设备具体用于执行以下步骤：

丢弃所述缓存队列中的M帧，其中，所述M小于或等于N，所述N为预设的常量；  
或者，丢弃所述缓存队列中的位于所述第一帧之前的帧；  
或者，丢弃所述第一帧以及所述缓存队列中的位于所述第一帧之前的帧；  
或者，丢弃所述缓存队列中的全部帧。

31. 一种计算机可读存储介质，所述计算机可读存储介质存储有计算机程序，其特征在于，所述计算机程序被处理器执行时实现如权利要求1-11任一项所述的方法，或者，执行时实现如权利要求12-15任一项所述的方法。

## 数据处理方法和相关装置

### 技术领域

[0001] 本申请涉及终端技术领域,尤其涉及数据处理方法和相关装置。

### 背景技术

[0002] 目前,用户可以通过终端设备的显示屏查阅各类内容。当内容较多时,显示屏不能一次显示全部内容,用户可在显示幕里滑动翻阅相关内容。

[0003] 终端设备的显示屏的界面显示通常需要经过绘制、渲染、合成等过程。示例性的,终端设备的界面绘制过程可以包括背景绘制、子视图的绘制、滚动条的绘制等过程。终端设备的界面合成过程可以包括顶点处理和像素处理等处理过程。

[0004] 但是,现有的终端设备的显示屏界面进行内容显示时,经常存在显示滞后,使得用户滑动的位置与显示屏界面显示的内容有错位。

### 发明内容

[0005] 本申请实施例提供数据处理方法和相关装置,应用于终端设备。终端设备接收并响应用户在显示屏上的滑动操作,当终端设备接收到用户反向滑动操作时,通过丢弃缓存队列中的全部或部分帧以降低反向滑动后界面显示的反应时延。

[0006] 第一方面,本申请实施例提出一种数据处理方法,应用于终端设备,该方法包括:接收用户输入的滑动操作;确定滑动操作对应的控件为列表控件;确定反向滑动事件发生,反向滑动事件为滑动操作的滑动方向发生反向的事件;在反向滑动事件发生时,为第一帧添加标识,第一帧为反向滑动事件发生时正在绘制的帧;在缓存队列中存在携带标识的第一帧的情况下,丢弃缓存队列中的部分或全部帧,缓存队列用于存储绘制完成的帧,绘制完成的帧与滑动操作相对应;合成待显示帧,待显示帧为执行丢弃动作后得到的待显示的帧;显示待显示帧。

[0007] 这样,可以通过丢弃缓存队列中的全部或部分帧,降低反向滑动后界面显示的反应时延。

[0008] 可选的,在缓存队列中存在携带标识的第一帧的情况下,丢弃缓存队列中的部分或全部帧,包括:终端设备中的显示合成进程确认缓存队列接收到携带标识的第一帧时,丢弃缓存队列中部分或全部帧;或者,显示合成进程从缓存队列获取待合成的帧时,丢弃缓存队列中部分或全部帧。

[0009] 可选的,显示合成进程从缓存队列获取待合成的帧时,丢弃缓存队列中部分或全部帧,具体为:每当周期信号到来时,显示合成进程确定缓存队列中是否包含有携带标识的第一帧;在缓存队列中包含有携带标识的第一帧的情况下,显示合成进程丢弃缓存队列中部分或全部帧。

[0010] 可选的,丢弃缓存队列中的部分或全部帧,包括:丢弃缓存队列中的M帧,其中,M小于或等于N,N为预设的常量;或者,丢弃缓存队列中的位于携带标识的第一帧之前的帧;或者,丢弃携带标识的第一帧以及缓存队列中的位于携带标识的第一帧之前的帧;或者,丢弃



缓存队列中的全部帧。

[0011] 可选的,当终端设备丢弃的帧为缓存队列中的部分帧时,待显示帧包括丢弃缓存队列中的部分帧后,缓存队列中剩余的帧;或者,当终端设备丢弃的帧为缓存队列中的全部帧时,待显示帧包括丢弃缓存队列中的全部帧后,缓存队列中新增加的渲染后的帧。

[0012] 可选的,终端设备包括图像绘制模块、图像渲染模块和图像合成模块,方法还包括:响应于滑动操作,图像绘制模块对滑动操作对应的帧进行绘制;图像渲染模块对绘制后的帧进行渲染;图像渲染模块将渲染后的帧存储在缓存队列;每当周期信号到来时,显示合成进程从缓存队列中选择第二帧;显示合成进程通过图像合成模块对第二帧进行合成。

[0013] 可选的,确定滑动操作对应的控件为列表控件之后,方法还包括:确定列表控件的方向属性,列表控件的方向属性包括:横向和纵向。

[0014] 可选的,终端设备包括视图系统,确定滑动操作对应的控件为列表控件,包括:视图系统确定滑动操作对应的对象的实例化为列表控件;确定列表控件的方向属性,包括:视图系统根据对象使用的变量值或接口函数的返回值,确定列表控件的方向属性为横向或纵向;其中,接口函数包括getOrientation(),变量值包括mOrientation。

[0015] 可选的,确定反向滑动事件发生,包括:在列表控件的方向属性为横向时,根据滑动操作产生的横向位移确定滑动操作是否发生反向;或者,在列表控件的方向属性为纵向时,根据滑动操作产生的纵向位移确定滑动操作是否发生反向。

[0016] 可选的,确定反向滑动事件发生,包括:在列表控件的方向属性为横向时,计算滑动操作产生的横向位移是否发生由负值变为正值,或由正值变为负值的变化;当滑动操作产生的横向位移发生由负值变为正值,或由正值变为负值的变化时,确定反向滑动事件发生;或者,在列表控件的方向属性为纵向时,计算滑动操作产生的纵向位移是否发生由负值变为正值,或由正值变为负值的变化;当滑动操作产生的纵向位移发生由负值变为正值,或由正值变为负值的变化时,确定反向滑动事件发生。

[0017] 可选的,视图系统还用于计算滑动操作的横向位移或纵向位移。

[0018] 可选的,列表控件为有序内容的集合,用于横向或纵向滑动,列表控件包括下述中的任一项:应用界面中的控件、设置界面中的控件、电子书阅读界面中的控件或文档界面中的控件。

[0019] 第二方面,本申请实施例提出一种数据处理方法,应用于终端设备,该方法包括:接收用户输入的滑动操作;确定反向滑动事件发生,反向滑动事件为滑动操作的滑动方向发生反向的事件;丢弃缓存队列中的部分或全部帧,缓存队列用于存储绘制完成的帧,绘制完成的帧与滑动操作相对应;显示待显示帧,待显示帧为执行丢弃动作后得到的待显示的帧。

[0020] 可选的,在缓存队列中存在第一帧的情况下,丢弃缓存队列中的部分或全部帧,第一帧为反向滑动事件发生时正在绘制的帧。

[0021] 可选的,在缓存队列中存在第一帧的情况下,丢弃缓存队列中的部分或全部帧,包括:终端设备中的显示合成进程确认缓存队列接收到第一帧时,丢弃缓存队列中部分或全部帧;或者,显示合成进程从缓存队列获取待合成的帧时,丢弃缓存队列中部分或全部帧。

[0022] 可选的,显示合成进程从缓存队列获取待合成的帧时,丢弃缓存队列中部分或全部帧,具体为:每当周期信号到来时,显示合成进程确定缓存队列中是否包含有第一帧;当

缓存队列中包含有第一帧的情况下,显示合成进程丢弃缓存队列中部分或全部帧。

[0023] 可选的,丢弃缓存队列中的部分或全部帧,包括:丢弃缓存队列中的M帧,其中,M小于或等于N,N为预设的常量;或者,丢弃缓存队列中的位于第一帧之前的帧;或者,丢弃第一帧以及缓存队列中的位于第一帧之前的帧;或者,丢弃缓存队列中的全部帧。

[0024] 第三方面,本申请实施例提供一种终端设备,终端设备也可以称为终端(terminal)、用户设备(user equipment,UE)、移动台(mobile station,MS)、移动终端(mobile terminal,MT)等。终端设备可以是手机(mobile phone)、智能电视、穿戴式设备、平板电脑(Pad)、带无线收发功能的电脑、虚拟现实(virtual reality,VR)终端设备、增强现实(augmented reality,AR)终端设备、工业控制(industrial control)中的无线终端、无人驾驶(self-driving)中的无线终端、远程手术(remote medical surgery)中的无线终端、智能电网(smart grid)中的无线终端、运输安全(transportation safety)中的无线终端、智慧城市(smart city)中的无线终端、智慧家庭(smart home)中的无线终端等等。

[0025] 该终端设备包括:触摸屏,其中,触摸屏包括触控面板和显示屏;一个或多个处理器;存储器;多个应用程序;以及一个或多个计算机程序,其中一个或多个计算机程序被存储在存储器中,一个或多个计算机程序包括指令,当指令被终端设备执行时,使得终端设备执行以下步骤:接收用户输入的滑动操作;确定滑动操作对应的控件为列表控件;确定反向滑动事件发生,反向滑动事件为滑动操作的滑动方向发生反向的事件;在反向滑动事件发生时,为第一帧添加标识,第一帧为反向滑动事件发生时正在绘制的帧;在缓存队列中存在携带标识的第一帧的情况下,丢弃缓存队列中的部分或全部帧,缓存队列用于存储绘制完成的帧,绘制完成的帧与滑动操作相对应;合成待显示帧,待显示帧为执行丢弃动作后得到的待显示的帧;显示待显示帧。

[0026] 可选的,终端设备具体用于执行以下步骤:确认缓存队列接收到携带标识的第一帧时,丢弃缓存队列中部分或全部帧;或者,从缓存队列获取待合成的帧时,丢弃缓存队列中部分或全部帧。

[0027] 可选的,终端设备具体用于执行以下步骤:在周期信号到来时,确定缓存队列中是否包含有携带标识的第一帧;在缓存队列中包含有携带标识的第一帧的情况下,丢弃缓存队列中部分或全部帧。

[0028] 可选的,终端设备具体用于执行以下步骤:丢弃缓存队列中的M帧,其中,M小于或等于N,N为预设的常量;或者,丢弃缓存队列中的位于携带标识的第一帧之前的帧;或者,丢弃携带标识的第一帧以及缓存队列中的位于携带标识的第一帧之前的帧;或者,丢弃缓存队列中的全部帧。

[0029] 可选的,当终端设备丢弃的帧为缓存队列中的部分帧时,待显示帧包括丢弃缓存队列中的部分帧后,缓存队列中剩余的帧;或者,当终端设备丢弃的帧为缓存队列中的全部帧时,待显示帧包括丢弃缓存队列中的全部帧后,缓存队列中新增加的渲染后的帧。

[0030] 可选的,终端设备包括图像绘制模块、图像渲染模块和图像合成模块,终端设备还用于响应于滑动操作,图像绘制模块用于对滑动操作对应的帧进行绘制;图像渲染模块用于对绘制后的帧进行渲染;图像渲染模块用于将渲染后的帧存储在缓存队列;每当周期信号到来时,终端设备用于从缓存队列中选择第二帧;终端设备用于通过图像合成模块对第二帧进行合成。

[0031] 可选的,终端设备还用于执行以下步骤:确定滑动操作对应的控件为列表控件之后,确定列表控件的方向属性,列表控件的方向属性包括:横向和纵向。

[0032] 可选的,终端设备包括视图系统,视图系统用于确定滑动操作对应的对象的实例化为列表控件;视图系统用于根据对象使用的变量值或接口函数的返回值,确定列表控件的方向属性为横向或纵向;其中,接口函数包括`getOrientation()`,变量值包括`mOrientation`。

[0033] 可选的,终端设备具体用于执行以下步骤:在列表控件的方向属性为横向时,根据滑动操作产生的横向位移确定滑动操作是否发生反向;或者,在列表控件的方向属性为纵向时,根据滑动操作产生的纵向位移确定滑动操作是否发生反向。

[0034] 可选的,终端设备,具体用于执行以下步骤:在列表控件的方向属性为横向时,计算滑动操作产生的横向位移是否发生由负值变为正值,或由正值变为负值的变化;当滑动操作产生的横向位移发生由负值变为正值,或由正值变为负值的变化时,确定反向滑动事件发生;或者,在列表控件的方向属性为纵向时,计算滑动操作产生的纵向位移是否发生由负值变为正值,或由正值变为负值的变化;当滑动操作产生的纵向位移发生由负值变为正值,或由正值变为负值的变化时,确定反向滑动事件发生。

[0035] 可选的,视图系统还用于计算滑动操作的横向位移或纵向位移。

[0036] 可选的,列表控件为有序内容的集合,用于横向或纵向滑动,列表控件包括下述中的任一项:应用界面中的控件、设置界面中的控件、电子书阅读界面中的控件或文档界面中的控件。

[0037] 第四方面,本申请实施例提供一种终端设备,终端设备包括:触摸屏,其中,触摸屏包括触控面板和显示屏;一个或多个处理器;存储器;多个应用程序;以及一个或多个计算机程序,其中一个或多个计算机程序被存储在存储器中,一个或多个计算机程序包括指令,当指令被终端设备执行时,使得终端设备执行以下步骤:接收用户输入的滑动操作;确定反向滑动事件发生,反向滑动事件为滑动操作的滑动方向发生反向的事件;丢弃缓存队列中的部分或全部帧,缓存队列用于存储绘制完成的帧,绘制完成的帧与滑动操作相对应;显示待显示帧,待显示帧为执行丢弃动作后得到的待显示的帧。

[0038] 可选的,终端设备具体用于执行以下步骤:在缓存队列中存在第一帧时,丢弃缓存队列中的部分或全部帧,第一帧为反向滑动事件发生时正在绘制的帧。

[0039] 可选的,终端设备具体用于执行以下步骤:确认缓存队列接收到携带标识的第一帧时,丢弃缓存队列中部分或全部帧;或者,从缓存队列获取待合成的帧时,丢弃缓存队列中部分或全部帧。

[0040] 可选的,终端设备具体用于执行以下步骤:每当周期信号到来时,确定缓存队列中是否包含有携带标识的第一帧;当缓存队列中包含有第一帧的情况下,丢弃缓存队列中部分或全部帧。

[0041] 可选的,终端设备具体用于执行以下步骤:丢弃缓存队列中的M帧,其中,M小于或等于N,N为预设的常量;或者,丢弃缓存队列中的位于第一帧之前的帧;或者,丢弃第一帧以及缓存队列中的位于第一帧之前的帧;或者,丢弃缓存队列中的全部帧。

[0042] 第五方面,本申请实施例提供一种计算机可读存储介质,计算机可读存储介质存储有计算机程序.计算机程序被处理器执行时实现如第一方面或第二方面所述的方法。

[0043] 第六方面,本申请实施例提供一种计算机程序产品,计算机程序产品包括计算机程序,当计算机程序被运行时,使得计算机执行如第一方面或第二方面所述的方法。

[0044] 应当理解的是,本申请的第二方面至第六方面与本申请的第一方面的技术方案相对应,各方面及对应的可行实施方式所取得的有益效果相似,不再赘述。

### 附图说明

[0045] 图1为本申请实施例提供的终端设备硬件系统结构示意图;

[0046] 图2为本申请实施例提供的终端设备软件系统结构示意图;

[0047] 图3为本申请实施例提供的一种场景示意图;

[0048] 图4为本申请实施例提供的一种终端设备界面显示处理流程示意图

[0049] 图5为本申请实施例提供的另一种终端设备界面显示处理流程示意图;

[0050] 图6为本申请实施例提供的一种设置显示触摸操作的界面示意图;

[0051] 图7为本申请实施例提供的一种滑动时界面示意图;

[0052] 图8为本申请实施例提供的一种降低屏幕显示的响应时延方法流程示意图;

[0053] 图9为本申请实施例提供的一种滑动显示的数据处理方法流程示意图;

[0054] 图10为本申请实施例提供的一种通常的反向滑动操作时帧显示的处理方法流程示意图;

[0055] 图11为本申请实施例提供的一种滑动显示处理方法流程示意图;

[0056] 图12为本申请实施例提供的一种滑动显示处理方法流程示意图;

[0057] 图13为本申请实施例提供的一种滑动显示处理方法流程示意图;

[0058] 图14为本申请实施例提供的一种数据处理方法流程示意图;

[0059] 图15为本申请实施例提供的一种终端设备确定反向滑动操作的方法的流程示意图;

[0060] 图16为本申请实施例提供的一种数据处理装置的结构示意图;

[0061] 图17为本申请实施例提供的一种数据处理装置的硬件结构示意图。

### 具体实施方式

[0062] 为了便于清楚描述本申请实施例的技术方案,在本申请的实施例中,采用了“第一”、“第二”等字样对功能和作用基本相同的相同项或相似项进行区分。例如,第一芯片和第二芯片仅仅是为了区分不同的芯片,并不对其先后顺序进行限定。本领域技术人员可以理解“第一”、“第二”等字样并不对数量和执行次序进行限定,并且“第一”、“第二”等字样也并不限定一定不同。

[0063] 需要说明的是,本申请实施例中,“示例性的”或者“例如”等词用于表示作例子、例证或说明。本申请中被描述为“示例性的”或者“例如”的任何实施例或设计方案不应被解释为比其他实施例或设计方案更优选或更具优势。确切而言,使用“示例性的”或者“例如”等词旨在以具体方式呈现相关概念。

[0064] 本申请实施例中,“至少一个”是指一个或者多个,“多个”是指两个或两个以上。“和/或”,描述关联对象的关联关系,表示可以存在三种关系,例如,A和/或B,可以表示:单独存在A,同时存在A和B,单独存在B的情况,其中A,B可以是单数或者复数。字符“/”一般表

示前后关联对象是一种“或”的关系。“以下至少一项(个)”或其类似表达,是指的这些项中的任意组合,包括单项(个)或复数项(个)的任意组合。例如,a,b,或c中的至少一项(个),可以表示:a,b,c,a-b,a-c,b-c,或a-b-c,其中a,b,c可以是单个,也可以是多个。

[0065] 本申请实施例提供的滑动显示方法,可以应用在具备显示功能的电子设备中。

[0066] 电子设备包括终端设备,终端设备也可以称为终端(terminal)、用户设备(user equipment,UE)、移动台(mobile station,MS)、移动终端(mobile terminal,MT)等。终端设备可以是手机(mobile phone)、智能电视、穿戴式设备、平板电脑(Pad)、带无线收发功能的电脑、虚拟现实(virtual reality,VR)终端设备、增强现实(augmented reality,AR)终端设备、工业控制(industrial control)中的无线终端、无人驾驶(self-driving)中的无线终端、远程手术(remote medical surgery)中的无线终端、智能电网(smart grid)中的无线终端、运输安全(transportation safety)中的无线终端、智慧城市(smart city)中的无线终端、智慧家庭(smart home)中的无线终端等等。本申请的实施例对终端设备所采用的具体技术和具体设备形态不做限定。

[0067] 为了更好地理解本申请实施例,下面对本申请实施例的终端设备的结构进行介绍:

[0068] 图1示出了终端设备100的结构示意图。终端设备100可以包括处理器110,外部存储器接口120,内部存储器121,通用串行总线(universal serial bus,USB)接口130,充电管理模块140,电源管理模块141,电池142,天线1,天线2,移动通信模块150,无线通信模块160,音频模块170,扬声器170A,受话器170B,麦克风170C,耳机接口170D,传感器模块180,按键190,马达191,指示器192,摄像头193,显示屏194,以及用户标识模块(subscriber identification module,SIM)卡接口195等。其中传感器模块180可以包括压力传感器180A,陀螺仪传感器180B,气压传感器180C,磁传感器180D,加速度传感器180E,距离传感器180F,接近光传感器180G,指纹传感器180H,温度传感器180J,触摸传感器180K,环境光传感器180L,骨传导传感器180M等。

[0069] 可以理解的是,本申请实施例示意的结构并不构成对终端设备100的具体限定。在本申请另一些实施例中,终端设备100可以包括比图示更多或更少的部件,或者组合某些部件,或者拆分某些部件,或者不同的部件布置。图示的部件可以以硬件,软件或软件和硬件的组合实现。

[0070] 处理器110可以包括一个或多个处理单元,例如:处理器110可以包括应用处理器(application processor,AP),调制解调处理器,图形处理器(graphics processing unit,GPU),图像信号处理器(image signal processor,ISP),控制器,视频编解码器,数字信号处理器(digital signal processor,DSP),基带处理器,和/或神经网络处理器(neural-network processing unit,NPU)等。其中,不同的处理单元可以是独立的器件,也可以集成在一个或多个处理器中。

[0071] 控制器可以根据指令操作码和时序信号,产生操作控制信号,完成取指令和执行指令的控制。

[0072] 处理器110中还可以设置存储器,用于存储指令和数据。在一些实施例中,处理器110中的存储器为高速缓冲存储器。该存储器可以保存处理器110刚用过或循环使用的指令或数据。如果处理器110需要再次使用该指令或数据,可从存储器中调用。避免了重复存取,

减少了处理器110的等待时间,因而提高了系统的效率。

[0073] 在一些实施例中,处理器110可以包括一个或多个接口。接口可以包括集成电路(inter-integrated circuit,I2C)接口,集成电路内置音频(inter-integrated circuitsound,I2S)接口,脉冲编码调制(pulse code modulation,PCM)接口,通用异步收发传输器(universal asynchronous receiver/transmitter,UART)接口,移动产业处理器接口(mobile industry processor interface,MIPI),通用输入输出(general-purposeinput/output,GPIO)接口,用户标识模块(subscriber identity module,SIM)接口,和/或通用串行总线(universal serial bus,USB)接口等。

[0074] I2C接口是一种双向同步串行总线,包括一根串行数据线(serial data line,SDA)和一根串行时钟线(derail clock line,SCL)。在一些实施例中,处理器110可以包含多组I2C总线。处理器110可以通过不同的I2C总线接口分别耦合触摸传感器180K,充电器,闪光灯,摄像头193等。例如:处理器110可以通过I2C接口耦合触摸传感器180K,使处理器110与触摸传感器180K通过I2C总线接口通信,实现终端设备100的触摸功能。

[0075] I2S接口可以用于音频通信。在一些实施例中,处理器110可以包含多组I2S总线。处理器110可以通过I2S总线与音频模块170耦合,实现处理器110与音频模块170之间的通信。在一些实施例中,音频模块170可以通过I2S接口向无线通信模块160传递音频信号,实现通过蓝牙耳机接听电话的功能。

[0076] PCM接口也可以用于音频通信,将模拟信号抽样,量化和编码。在一些实施例中,音频模块170与无线通信模块160可以通过PCM总线接口耦合。在一些实施例中,音频模块170也可以通过PCM接口向无线通信模块160传递音频信号,实现通过蓝牙耳机接听电话的功能。I2S接口和PCM接口都可以用于音频通信。

[0077] UART接口是一种通用串行数据总线,用于异步通信。该总线可以为双向通信总线。它将要传输的数据在串行通信与并行通信之间转换。在一些实施例中,UART接口通常被用于连接处理器110与无线通信模块160。例如:处理器110通过UART接口与无线通信模块160中的蓝牙模块通信,实现蓝牙功能。在一些实施例中,音频模块170可以通过UART接口向无线通信模块160传递音频信号,实现通过蓝牙耳机播放音乐的功能。

[0078] MIPI接口可以被用于连接处理器110与显示屏194,摄像头193等外围器件。MIPI接口包括摄像头串行接口(camera serial interface,CSI),显示屏串行接口(displayserial interface,DSI)等。在一些实施例中,处理器110和摄像头193通过CSI接口通信,实现终端设备100的拍摄功能。处理器110和显示屏194通过DSI接口通信,实现终端设备100的显示功能。

[0079] GPIO接口可以通过软件配置。GPIO接口可以被配置为控制信号,也可被配置为数据信号。在一些实施例中,GPIO接口可以用于连接处理器110与摄像头193,显示屏194,无线通信模块160,音频模块170,传感器模块180等。GPIO接口还可以被配置为I2C接口,I2S接口,UART接口,MIPI接口等。

[0080] USB接口130是符合USB标准规范的接口,具体可以是Mini USB接口,Micro USB接口,USB Type C接口等。USB接口130可以用于连接充电器为终端设备100充电,也可以用于终端设备100与外围设备之间传输数据。也可以用于连接耳机,通过耳机播放音频。该接口还可以用于连接其他电子设备,例如AR设备等。

[0081] 可以理解的是,本申请实施例示意的各模块间的接口连接关系,是示意性说明,并不构成对终端设备100的结构限定。在本申请另一些实施例中,终端设备100也可以采用上述实施例中不同的接口连接方式,或多种接口连接方式的组合。

[0082] 充电管理模块140用于从充电器接收充电输入。其中,充电器可以是无线充电器,也可以是有线充电器。在一些有线充电的实施例中,充电管理模块140可以通过USB接口130接收有线充电器的充电输入。在一些无线充电的实施例中,充电管理模块140可以通过终端设备100的无线充电线圈接收无线充电输入。充电管理模块140为电池142充电的同时,还可以通过电源管理模块141为终端设备供电。

[0083] 电源管理模块141用于连接电池142,充电管理模块140与处理器110。电源管理模块141接收电池142和/或充电管理模块140的输入,为处理器110,内部存储器121,显示屏194,摄像头193,和无线通信模块160等供电。电源管理模块141还可以用于监测电池容量,电池循环次数,电池健康状态(漏电,阻抗)等参数。在其他一些实施例中,电源管理模块141也可以设置于处理器110中。在另一些实施例中,电源管理模块141和充电管理模块140也可以设置于同一个器件中。

[0084] 终端设备100的无线通信功能可以通过天线1,天线2,移动通信模块150,无线通信模块160,调制解调处理器以及基带处理器等实现。

[0085] 天线1和天线2用于发射和接收电磁波信号。终端设备100中的天线可用于覆盖单个或多个通信频带。不同的天线还可以复用,以提高天线的利用率。例如:可以将天线1复用为无线局域网的分集天线。在另外一些实施例中,天线可以和调谐开关结合使用。

[0086] 移动通信模块150可以提供应用在终端设备100上的包括2G/3G/4G/5G等无线通信的解决方案。移动通信模块150可以包括至少一个滤波器,开关,功率放大器,低噪声放大器(low noise amplifier,LNA)等。移动通信模块150可以由天线1接收电磁波,并对接收的电磁波进行滤波,放大等处理,传送至调制解调处理器进行解调。移动通信模块150还可以对经调制解调处理器调制后的信号放大,经天线1转为电磁波辐射出去。在一些实施例中,移动通信模块150的至少部分功能模块可以被设置于处理器110中。在一些实施例中,移动通信模块150的至少部分功能模块可以与处理器110的至少部分模块被设置在同一个器件中。

[0087] 调制解调处理器可以包括调制器和解调器。其中,调制器用于将待发送的低频基带信号调制成中高频信号。解调器用于将接收的电磁波信号解调为低频基带信号。随后解调器将解调得到的低频基带信号传送至基带处理器处理。低频基带信号经基带处理器处理后,被传递给应用处理器。应用处理器通过音频设备(不限于扬声器170A,受话器170B等)输出声音信号,或通过显示屏194显示图像或视频。在一些实施例中,调制解调处理器可以是独立的器件。在另一些实施例中,调制解调处理器可以独立于处理器110,与移动通信模块150或其他功能模块设置在同一个器件中。

[0088] 无线通信模块160可以提供应用在终端设备100上的包括无线局域网(wireless local area networks,WLAN)(如无线保真(wireless fidelity,Wi-Fi)网络),蓝牙(blueetooth,BT),全球导航卫星系统(global navigation satellite system,GNSS),调频(frequency modulation,FM),近距离无线通信技术(near field communication,NFC),红外技术(infrared,IR)等无线通信的解决方案。无线通信模块160可以是集成至少一个通信处理模块的一个或多个器件。无线通信模块160经由天线2接收电磁波,将电磁波

信号调频以及滤波处理,将处理后的信号发送到处理器110。无线通信模块160还可以从处理器110接收待发送的信号,对其进行调频,放大,经天线2转为电磁波辐射出去。

[0089] 在一些实施例中,终端设备100的天线1和移动通信模块150耦合,天线2和无线通信模块160耦合,使得终端设备100可以通过无线通信技术与网络以及其他设备通信。无线通信技术可以包括全球移动通讯系统(global system for mobile communications, GSM),通用分组无线服务(general packet radio service,GPRS),码分多址接入(codedivision multiple access,CDMA),宽带码分多址(wideband code division multipleaccess,WCDMA),时分码分多址(time-division code division multiple access,TD-SCDMA),长期演进(long term evolution,LTE),BT,GNSS,WLAN,NFC,FM,和/或IR技术等。GNSS可以包括全球卫星定位系统(global positioning system,GPS),全球导航卫星系统(global navigation satellite system,GLONASS),北斗卫星导航系统(beidou navigation satellite system,BDS),准天顶卫星系统(quasi-zenith satellitesystem,QZSS)和/或星基增强系统(satellite based augmentation systems, SBAS)。

[0090] 终端设备100通过GPU,显示屏194,以及应用处理器等实现显示功能。GPU为图像处理的微处理器,连接显示屏194和应用处理器。GPU用于执行数学和几何计算,用于图形渲染。处理器110可包括一个或多个GPU,其执行程序指令以生成或改变显示信息。

[0091] 显示屏194用于显示图像、显示视频和接收滑动操作等。显示屏194包括显示面板。显示面板可以采用液晶显示屏(liquid crystal display,LCD),有机发光二极管(organic light-emittingdiode,OLED),有源矩阵有机发光二极体或主动矩阵有机发光二极体(active-matrixorganic light emitting diod,AMOLED),柔性发光二极管(flex light-emittingdiode,FLED),Miniled,MicroLed,Micro-oLed,量子点发光二极管(quantum dot lightemitting diodes,QLED)等。在一些实施例中,终端设备100可以包括1个或N个显示屏194,N为大于1的正整数。

[0092] 终端设备100可以通过ISP,摄像头193,视频编解码器,GPU,显示屏194以及应用处理器等实现拍摄功能。

[0093] ISP用于处理摄像头193反馈的数据。例如,拍照时,打开快门,光线通过镜头被传递到摄像头感光元件上,光信号转换为电信号,摄像头感光元件将电信号传递给ISP处理,转化为肉眼可见的图像。ISP还可以对图像的噪点,亮度,肤色进行算法优化。ISP还可以对拍摄场景的曝光,色温等参数优化。在一些实施例中,ISP可以设置在摄像头193中。

[0094] 摄像头193用于捕获静态图像或视频。物体通过镜头生成光学图像投射到感光元件。感光元件可以是电荷耦合器件(charge coupled device,CCD)或互补金属氧化物半导体(complementary metal-oxide-semiconductor,CMOS)光电晶体管。感光元件把光信号转换成电信号,之后将电信号传递给ISP转换成数字图像信号。ISP将数字图像信号输出到DSP加工处理。DSP将数字图像信号转换成标准的RGB,YUV等格式的图像信号。在一些实施例中,终端设备100可以包括1个或N个摄像头193,N为大于1的正整数。

[0095] 数字信号处理器用于处理数字信号,除了可以处理数字图像信号,还可以处理其他数字信号。例如,当终端设备100在频点选择时,数字信号处理器用于对频点能量进行傅里叶变换等。



[0096] 视频编解码器用于对数字视频压缩或解压缩。终端设备100可以支持一种或多种视频编解码器。这样,终端设备100可以播放或录制多种编码格式的视频,例如:动态图像专家组(moving picture experts group,MPEG)1,MPEG2,MPEG3,MPEG4等。

[0097] NPU为神经网络(neural-network,NN)计算处理器,通过借鉴生物神经网络结构,例如借鉴人脑神经元之间传递模式,对输入信息快速处理,还可以不断的自学习。通过NPU可以实现终端设备100的智能认知等应用,例如:图像识别,人脸识别,语音识别,文本理解等。

[0098] 外部存储器接口120可以用于连接外部存储卡,例如Micro SD卡,实现扩展终端设备100的存储能力。外部存储卡通过外部存储器接口120与处理器110通信,实现数据存储功能。例如将音乐,视频等文件保存在外部存储卡中。

[0099] 内部存储器121可以用于存储计算机可执行程序代码,可执行程序代码包括指令。内部存储器121可以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(比如声音播放功能,图像播放功能等)等。存储数据区可存储终端设备100使用过程中所创建的数据(比如音频数据,电话本等)等。此外,内部存储器121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,通用闪存存储器(universal flash storage,UFS)等。处理器110通过运行存储在内部存储器121的指令,和/或存储在设置于处理器中的存储器的指令,执行终端设备100的各种功能应用以及数据处理。

[0100] 终端设备100可以通过音频模块170,扬声器170A,受话器170B,麦克风170C,耳机接口170D,以及应用处理器等实现音频功能。例如音乐播放,录音等。

[0101] 音频模块170用于将数字音频信息转换成模拟音频信号输出,也用于将模拟音频输入转换为数字音频信号。音频模块170还可以用于对音频信号编码和解码。在一些实施例中,音频模块170可以设置于处理器110中,或将音频模块170的部分功能模块设置于处理器110中。

[0102] 扬声器170A,也称“喇叭”,用于将音频电信号转换为声音信号。终端设备100可以通过扬声器170A收听音乐,或收听免提通话。

[0103] 受话器170B,也称“听筒”,用于将音频电信号转换成声音信号。当终端设备100接听电话或语音信息时,可以通过将受话器170B靠近人耳接听语音。

[0104] 麦克风170C,也称“话筒”,“传声器”,用于将声音信号转换为电信号。当拨打电话或发送语音信息时,用户可以通过人嘴靠近麦克风170C发声,将声音信号输入到麦克风170C。终端设备100可以设置至少一个麦克风170C。在另一些实施例中,终端设备100可以设置两个麦克风170C,除了采集声音信号,还可以实现降噪功能。在另一些实施例中,终端设备100还可以设置三个,四个或更多麦克风170C,实现采集声音信号,降噪,还可以识别声音来源,实现定向录音功能等。

[0105] 耳机接口170D用于连接有线耳机。耳机接口170D可以是USB接口130,也可以是3.5mm的开放移动电子设备平台(open mobile terminal platform,OMTP)标准接口,美国蜂窝电信工业协会(cellular telecommunications industry association of the USA,CTIA)标准接口。

[0106] 压力传感器180A用于感受压力信号,可以将压力信号转换成电信号。在一些实施

例中,压力传感器180A可以设置于显示屏194。压力传感器180A的种类很多,如电阻式压力传感器,电感式压力传感器,电容式压力传感器等。电容式压力传感器可以是包括至少两个具有导电材料的平行板。当有力作用于压力传感器180A,电极之间的电容改变。终端设备100根据电容的变化确定压力的强度。当有触摸操作作用于显示屏194,终端设备100根据压力传感器180A检测触摸操作强度。终端设备100也可以根据压力传感器180A的检测信号计算触摸的位置。在一些实施例中,作用于相同触摸位置,但不同触摸操作强度的触摸操作,可以对应不同的操作指令。

[0107] 陀螺仪传感器180B可以用于确定终端设备100的运动姿态。在一些实施例中,可以通过陀螺仪传感器180B确定终端设备100围绕三个轴(即,x、y和z轴)的角速度。陀螺仪传感器180B可以用于拍摄防抖。示例性的,当按下快门,陀螺仪传感器180B检测终端设备100抖动的角度,根据角度计算出镜头模组需要补偿的距离,让镜头通过反向运动抵消终端设备100的抖动,实现防抖。陀螺仪传感器180B还可以用于导航,体感游戏场景。

[0108] 气压传感器180C用于测量气压。在一些实施例中,终端设备100通过气压传感器180C测得的气压值计算海拔高度,辅助定位和导航。

[0109] 磁传感器180D包括霍尔传感器。终端设备100可以利用磁传感器180D检测翻盖皮套的开合。在一些实施例中,当终端设备100是翻盖机时,终端设备100可以根据磁传感器180D检测翻盖的开合。进而根据检测到的皮套的开合状态或翻盖的开合状态,设置翻盖自动解锁等特性。

[0110] 加速度传感器180E可检测终端设备100在各个方向上(一般为三轴)加速度的大小。当终端设备100静止时可检测出重力的大小及方向。还可以用于识别终端设备姿态,应用于横竖屏切换,计步器等应用程序。

[0111] 距离传感器180F,用于测量距离。终端设备100可以通过红外或激光测量距离。在一些实施例中,拍摄场景,终端设备100可以利用距离传感器180F测距以实现快速对焦。

[0112] 接近光传感器180G可以包括例如发光二极管(LED)和光检测器,例如光电二极管。发光二极管可以是红外发光二极管。终端设备100通过发光二极管向外发射红外光。终端设备100使用光电二极管检测来自附近物体的红外反射光。当检测到充分的反射光时,可以确定终端设备100附近有物体。当检测到不充分的反射光时,终端设备100可以确定终端设备100附近没有物体。终端设备100可以利用接近光传感器180G检测用户手持终端设备100贴近耳朵通话,以便自动熄灭屏幕达到省电的目的。接近光传感器180G也可用于皮套模式,口袋模式自动解锁与锁屏。

[0113] 环境光传感器180L用于感知环境光亮度。终端设备100可以根据感知的环境光亮度自适应调节显示屏194亮度。环境光传感器180L也可用于拍照时自动调节白平衡。环境光传感器180L还可以与接近光传感器180G配合,检测终端设备100是否在口袋里,以防误触。

[0114] 指纹传感器180H用于采集指纹。终端设备100可以利用采集的指纹特性实现指纹解锁,访问应用锁,指纹拍照,指纹接听来电等。

[0115] 温度传感器180J用于检测温度。在一些实施例中,终端设备100利用温度传感器180J检测的温度,执行温度处理策略。例如,当温度传感器180J上报的温度超过阈值,终端设备100执行降低位于温度传感器180J附近的处理器的性能,以便降低功耗实施热保护。在另一些实施例中,当温度低于另一阈值时,终端设备100对电池142加热,以避免低温导致终

端设备100异常关机。在其他一些实施例中,当温度低于又一阈值时,终端设备100对电池142的输出电压执行升压,以避免低温导致的异常关机。

[0116] 触摸传感器180K,也称“触控器件”。触摸传感器180K可以设置于显示屏194,由触摸传感器180K与显示屏194组成触摸屏,也称“触控屏”。触摸传感器180K用于检测作用于其上或附近的触摸操作。触摸传感器可以将检测到的触摸操作传递给应用处理器,以确定触摸事件类型。可以通过显示屏194提供与触摸操作相关的视觉输出。在另一些实施例中,触摸传感器180K也可以设置于终端设备100的表面,与显示屏194所处的位置不同。

[0117] 骨传导传感器180M可以获取振动信号。在一些实施例中,骨传导传感器180M可以获取人体声部振动骨块的振动信号。骨传导传感器180M也可以接触人体脉搏,接收血压跳动信号。在一些实施例中,骨传导传感器180M也可以设置于耳机中,结合成骨传导耳机。音频模块170可以基于骨传导传感器180M获取的声部振动骨块的振动信号,解析出语音信号,实现语音功能。应用处理器可以基于骨传导传感器180M获取的血压跳动信号解析心率信息,实现心率检测功能。

[0118] 按键190包括开机键,音量键等。按键190可以是机械按键。也可以是触摸式按键。终端设备100可以接收按键输入,产生与终端设备100的用户设置以及功能控制有关的键信号输入。

[0119] 马达191可以产生振动提示。马达191可以用于来电振动提示,也可以用于触摸振动反馈。例如,作用于不同应用程序(例如拍照,音频播放等)的触摸操作,可以对应不同的振动反馈效果。作用于显示屏194不同区域的触摸操作,马达191也可对应不同的振动反馈效果。不同的应用场景(例如:时间提醒,接收信息,闹钟,游戏等)也可以对应不同的振动反馈效果。触摸振动反馈效果还可以支持自定义。

[0120] 指示器192可以是指示灯,可以用于指示充电状态,电量变化,也可以用于指示消息,未接来电,通知等。

[0121] SIM卡接口195用于连接SIM卡。SIM卡可以通过插入SIM卡接口195,或从SIM卡接口195拔出,实现和终端设备100的接触和分离。终端设备100可以支持1个或N个SIM卡接口,N为大于1的正整数。SIM卡接口195可以支持Nano SIM卡, Micro SIM卡, SIM卡等。同一个SIM卡接口195可以同时插入多张卡。多张卡的类型可以相同,也可以不同。SIM卡接口195也可以兼容不同类型的SIM卡。SIM卡接口195也可以兼容外部存储卡。终端设备100通过SIM卡和网络交互,实现通话以及数据通信等功能。在一些实施例中,终端设备100采用eSIM,即:嵌入式SIM卡。eSIM卡可以嵌在终端设备100中,不能和终端设备100分离。

[0122] 终端设备100的软件系统可以采用分层架构,事件驱动架构,微核架构,微服务架构,或云架构,等。本申请实施例以分层架构的Android系统为例,示例性说明终端设备100的软件结构。

[0123] 图2是本申请实施例的终端设备的软件结构框图。

[0124] 分层架构将软件分成若干个层,每一层都有清晰的角色和分工。层与层之间通过软件接口通信。在一些实施例中,将Android系统分为四层,从上至下分别为应用层,应用程序框架层,安卓运行时(Android runtime)和系统库,以及内核层。

[0125] 应用层可以包括一系列应用程序包。

[0126] 如图2所示,应用程序包可以包括电话、邮箱、日历、相机等应用程序。

[0127] 应用程序框架层为应用层的应用程序提供应用编程接口(application programming interface,API)和编程框架。应用程序框架层包括一些预先定义的函数。

[0128] 如图2所示,应用程序框架层可以包括输入系统、活动管理器、位置管理器、包管理器、通知管理器、资源管理器、电话管理器和视图系统等。

[0129] 输入系统用于管理输入设备的程序。例如,输入系统可以确定鼠标点击操作、键盘输入操作和触摸滑动等输入操作。

[0130] 活动管理器用于管理各个应用程序的生命周期以及导航回退功能。负责Android的主线程创建,各个应用程序的生命周期的维护。

[0131] 位置管理器用于为应用程序提供位置服务,包括查询上一个已知位置、注册和注销来自某个周期性的位置更新等。

[0132] 包管理器用于系统内的程序管理,例如:应用程序安装、卸载和升级等。

[0133] 通知管理器使应用程序可以在状态栏中显示通知信息,可以用于传达告知类型的消息,可以短暂停留后自动消失,无需用户交互。比如通知管理器被用于告知下载完成,消息提醒等。通知管理器还可以是以图表或者滚动条文本形式出现在系统顶部状态栏的通知,例如后台运行的应用程序的通知,还可以是以对话框形式出现在屏幕上的通知。例如在状态栏提示文本信息,发出提示音,终端设备振动,指示灯闪烁等。

[0134] 资源管理器为应用程序提供各种资源,比如本地化字符串,图标,图片,布局文件,视频文件等等。

[0135] 电话管理器用于管理移动设备功能,包括:手机通话状态、获取电话信息(设备、sim卡、网络信息),监听电话状态以及调用电话拨号器拨打电话

[0136] 视图系统包括可视控件,例如显示文字的控件,显示图片的控件等。视图系统可用于构建应用程序。显示界面可以由一个或多个视图组成的。例如,包括短信通知图标的显示界面,可以包括显示文字的视图以及显示图片的视图。

[0137] Android runtime包括核心库和虚拟机。Android runtime负责安卓系统的调度和管理。

[0138] 核心库包含两部分:一部分是java语言需要调用的功能函数,另一部分是安卓的核心库。

[0139] 应用层和应用程序框架层运行在虚拟机中。虚拟机将应用层和应用程序框架层的java文件执行为二进制文件。虚拟机用于执行对象生命周期的管理,堆栈管理,线程管理,安全和异常的管理,以及垃圾回收等功能。

[0140] 系统库可以包括多个功能模块。例如:图像绘制模块、图像渲染模块、图像合成模块、函数库、和输入处理库等。

[0141] 图像绘制模块用于二维或三维图像的绘制。图像渲染模块用于二维或三维图像的渲染。图像合成模块用于二维或三维图像的合成。

[0142] 可能的实现方式中,应用通过图像绘制模块对图像进行绘制,然后应用通过图像渲染模块对绘制后的图像进行渲染,然后应用将渲染后的图像发送显示合成进程的到缓存队列中。每当vsync信号到来时,显示合成进程(例如,surface flinger)从缓存队列中按顺序获取待合成的一帧图像,然后通过图像合成模块进行图像合成。

[0143] 函数库提供C语言中所使用的宏、类型定义、字符串操作函数、数学计算函数以及

输入输出函数等

[0144] 输入处理库用于处理输入设备的库,可以实现鼠标、键盘和触摸输入处理等。

[0145] 内核层是硬件和软件之间的层。内核层至少包含触控面板驱动、LCD/LED屏幕驱动、显示驱动、蓝牙驱动、WIFI驱动、键盘驱动、共用存储器驱动和相机驱动等。

[0146] 硬件可以是音频设备、蓝牙设备、相机设备、传感器设备等。

[0147] 下面结合应用程序启动或应用程序中发生界面切换的场景,示例性说明终端设备100软件以及硬件的工作流程。

[0148] 当触控面板中触摸传感器180K接收到触摸操作时,内核层将触摸操作加工成原始输入事件(包括触摸坐标,触摸力度,触摸操作的时间戳等信息)。原始输入事件被存储在内核层。内核层通过输入处理库将原始输入事件上报至应用程序框架层的输入系统。应用程序框架层的输入系统解析该原始输入事件的信息(包括:操作类型和报点位置等)和根据当前焦点确定焦点应用,并将解析后的信息发送至焦点应用。焦点可以是触摸操作中触碰点或者鼠标点击操作中点击位置。焦点应用为终端设备前台运行的应用或者触摸操作中触碰位置对应的应用。焦点应用根据解析后的原始输入事件的信息(例如,报点位置)确定该原始输入事件所对应的控件。

[0149] 以该触摸操作是触摸滑动操作,该触摸滑动操作所对应的控件为微信应用的列表控件为例,微信应用通过应用程序框架层的视图系统,调用系统库中图像绘制模块对图像进行绘制,图像渲染模块对绘制后的图像进行渲染。微信应用将渲染后的图像发送至显示合成进程的缓存队列中。通过系统库中图像合成模块将显示合成进程中渲染后的图像合成为微信列表界面。显示合成进程通过内核层的LCD/LED屏幕驱动,使得LCD/LED屏幕显示微信应用的相应列表界面。

[0150] 为了便于理解,示例的给出部分与本申请实施例相关概念的说明以供参考。

[0151] 1、帧:是指界面显示中最小单位的单幅画面。一帧可以理解为一副静止的画面,快速连续地显示多个相连的帧可以形成物体运动的假象。帧率是指在1秒钟时间里刷新图片的帧数,也可以理解为终端设备中图形处理器每秒钟刷新画面的次数。高的帧率可以得到更流畅和更逼真的动画。每秒钟帧数越多,所显示的动作就会越流畅。

[0152] 需要说明的是,界面显示帧前通常需要经过绘制、渲染、合成等过程。

[0153] 2、帧绘制:是指显示界面的图片绘制。显示界面可以由一个或多个视图组成,各个视图可以由视图系统的可视控件绘制,各个视图由子视图组成,一个子视图对应视图中的一个部件,例如,其中的一个子视图对应图片视图中的一个符号。

[0154] 3、帧渲染:是将绘制后的视图进行着色操作或增加3D效果等。例如:3D效果可以是灯光效果、阴影效果和纹理效果等。

[0155] 4、帧合成:是将多个上述一个或多个渲染后的视图合成为显示界面的过程。

[0156] 下面结合附图对本申请实施例提供的应用场景进行说明。图3为本申请实施例提供的应用场景示意图。

[0157] 终端设备可以在图3中的a所示的社交应用的界面,或在图3中的b所示的设置相关界面中,图3中的c所示的文档界面,图3中的d所示的商品浏览界面,等接收用户上滑操作或下滑操作。终端设备还可以在图3中的e所示的界面中,图3中的f所示的电子书籍界面,等接收到用户左滑操作或右滑动操作。当终端设备接收到用户的滑动操作时,终端设备基于滑动

操作进行帧绘制、渲染、合成等过程,对滑动操作对应的内容进行显示。

[0158] 下面根据图4-图8对终端设备的滑动时界面显示的过程进行说明。

[0159] 图4为一种终端设备界面显示处理流程示意图。图4中帧1、帧2和帧3显示的内容与不同的界面相对应。图4中垂直同步信号(vetical synchronization, vsync)用于控制帧的绘制渲染、合成等进程的起始。vsync的周期时间与帧率相关。示例性的,当终端设备的系统刷新帧率为60帧/秒(frames per second, fps)时,vsync的周期时间为16.6毫秒(millisecond, ms)。垂直同步信号(vsync)也可以称为周期信号。

[0160] 具体的,结合图4,当终端设备接收到用户在应用界面内向上滑动的操作时,该滑动操作依次对应于帧1、帧2和帧3,终端设备中的应用通过应用程序框架层的视图系统,依次对帧1、帧2和帧3进行绘制渲染。帧1绘制渲染完成后,终端设备的应用调用系统库的显示合成进程,例如,surface flinger,在显示合成进程进行合成。帧1完成合成后,终端设备可以通过调用内核层启动LCD/LED屏幕驱动,显示帧1对应的内容。帧2和帧3类似于帧1的过程也进行合成和显示,此处不再赘述。但是,如图4所示,帧1、帧2和帧3从绘制至显示有2个vsync周期的响应延迟。若终端设备的系统刷新帧率为60fps,当每帧从绘制到显示,相差滞后2个vsync周期时,终端设备的响应时延为 $2*16.6\text{ms}$ ,导致界面显示内容滞后。

[0161] 需要说明的是,图4所示的绘制过程中,帧1、帧2和帧3的合成在1个vsync周期完成,不会造成缓存的积压,而在终端设备的实际应用中,可能会由于终端设备的系统负载大或网络差等其他原因,导致终端设备合成帧的时间增加。缓存中可能积压多个等待合成的帧,使得响应时延增加,界面显示滞后更加严重。

[0162] 示例性的,图5为另一种终端设备界面显示处理流程示意图。图5中帧1、帧2、帧3、帧4和帧5显示的内容与不同的界面相对应。以终端设备的系统刷新帧率为60fps为例,图5中1个vsync周期为16.6ms。

[0163] 具体的,结合图5,当终端设备接收到用户向上滑动的操作时,该滑动操作依次对应于帧1、帧2、帧3、帧4和帧5,终端设备对图5中的帧1、帧2、帧3、帧4和帧5分别进行绘制渲染、合成和显示,具体可以参照图4对应的描述。

[0164] 与图4不同的是,图5中帧1由于各种原因导致合成时间增加到3个vsync周期的时间,例如,原因包括图形绘制的复杂或者系统负载受到下载数据等影响存在波动等。示例性的,以缓存队列长度为3帧为例,帧2、帧3、帧4和帧5绘制完成后依次放入缓存队列中等待合成,则图5中帧1、帧2、帧3、帧4和帧5的从绘制至显示均有4个vsync周期的响应延迟。当每帧从绘制到显示,相差滞后4个vsync周期时,终端设备的响应时延为 $4*16.6\text{ms}$ ,终端设备显示的滞后性更为明显。

[0165] 终端设备显示的滞后性,可以体现在用户在显示屏中的触点与手机中显示的内容存在错位的现象,下面结合图6和图7对界面显示的触摸操作的触点与手指在屏幕实际触点的对比对响应时延进行描述。当终端设备的响应时延越高时,界面显示的触摸操作的触点位置与手指在屏幕实际触点位置距离越大。

[0166] 示例性的,图6为设置显示触摸操作的界面示意图。当终端设备在图6中的a所示的主界面中,接收到用户打开终端设备的设置应用程序601的操作时,终端设备可以进入到图6中的b所示的设置界面。如图6中的b所示,该设置界面中可以包括设置标签栏602以及设置项,例如,设置项可以包括下述的一种或多种:系统和更新、通知、生物识别和密码、应用、电

池、存储、安全、隐私、健康使用手机或其他类型的设置项。

[0167] 当用户在图6中的b所示的设置界面中通过点击、触摸等操作触发系统和更新控件603时,终端设备接收到用户设置系统和更新603的操作,终端设备进入到如图6中的c所示的设置系统和更新的界面。该设置应用程序的界面中可以包括系统和更新标签栏604,以及系统和更新项,例如,系统和更新项可以包括下述的一种或多种:软件更新、系统单行方式、语言和输入法、日期和时间、手机克隆、备份和恢复、显示触摸操作或其他类型的应用程序项。

[0168] 当用户在图6中的c所示的系统和更新界面中通过点击、触摸等操作打开显示触摸操作控件605时,终端设备开启显示触摸操作,则后续当用户在显示屏中触摸时,终端设备可以显示用户点击或触摸屏幕时的触点。

[0169] 示例性的,图7为终端设备开启显示触摸操作后的一种滑动时界面示意图。

[0170] 如图7所示,当用户手指触摸屏幕并向上滑动时,终端设备界面显示的内容发生变化。例如,用户在图7中的a所示的界面中执行上滑操作,终端设备依次显示如图7b-d所示的界面。

[0171] 具体的,终端设备在如图7中的a所示的设置界面中,接收到用户向上滑动的触摸操作。此时,界面显示的触摸操作的触点701位置与手指在屏幕实际触点702位置一致。

[0172] 由于终端设备存在响应时延,界面显示滞后,在后续的界面显示中,触摸操作的触点位置与手指在屏幕实际触点位置出现偏差。示例性的,图7中的b所示的界面中,触摸操作的触点703位置与手指在屏幕实际触点704位置出现偏差。图7中的c所示的界面显示时,触摸操作的触点705位置与手指在屏幕实际触点706位置出现偏差。

[0173] 终端设备在界面显示过程中,可能受到的系统负载或其他原因的影响,导致终端设备合成帧的时间增加,使得响应时延增加,界面显示滞后增加。

[0174] 例如,终端设备在合成图7中的d所示的界面时,系统负载较大,图7中的d所示的界面合成时间较长,导致时延增加。图7中的d所示的界面显示时,触摸操作的触点707位置与手指在屏幕实际触点708位置偏差增大。

[0175] 为了减少终端设备的显示滞后,可能的设计中,提出了降低屏幕显示的响应时延方法,下面结合图8,说明对游戏场景中应用的降低屏幕显示的响应时延方法进行说明。

[0176] 终端设备可以提供用户界面,并在用户界面中接收用户的首次点击操作,进而执行下述的步骤。

[0177] S801、终端设备在应用程序框架层的输入系统中解析用户首次点击操作的信息。

[0178] 例如,应用程序框架层的输入系统解析该首次点击操作的信息,得到该首次操作的操作类型和报点位置等,根据当前焦点确定焦点应用为游戏应用,并将解析后的信息发送至游戏应用。

[0179] S802、终端设备在应用程序框架层的视图系统判断待显示界面是否为游戏场景和满足条件限制。

[0180] 示例性的,条件限制可以是终端设备是否进入游戏加速状态或者终端设备的屏幕显示的画质要求和帧数要求等。

[0181] 本申请实施例中,当终端设备确定待显示界面为游戏界面和满足条件限制时,终端设备执行S803-S809。示例性的,终端设备可以并行执行S803和S805,本申请实施例对

S803和S805的先后顺序不做限定。

[0182] 当终端设备确定待显示界面不是游戏界面或者不满足条件限制时,终端设备执行S805-S809。下面对S803-S809进行说明。

[0183] S803、当终端设备确定待显示界面为游戏界面和满足条件限制时,通知系统库的图像合成模块清空缓存队列中的待显示界面。

[0184] S804、图像合成模块丢弃缓存历史帧。

[0185] 其中,图像合成模块可以运行在显示合成进程中。

[0186] S805、应用程序框架层将解析后的信息给游戏应用,游戏应用对用户点击操作进行输入处理。

[0187] 终端设备通过游戏应用输入处理可以确定首次点击的待显示界面的大小、显示内容等相关信息。

[0188] S806、游戏应用调用系统库的图像绘制模块对首次点击的待显示界面进行图像绘制。

[0189] S807、游戏应用调用系统库的图像渲染模块对首次点击的待显示界面进行图像渲染。

[0190] S808、游戏应用调用系统库的图像合成模块对首次点击的待显示界面进行图像合成。

[0191] S809、终端设备通过LCD或LED显示首次点击的待显示界面。

[0192] 这样,终端设备通过丢弃缓存队列中的帧,优先合成并显示用户首次点击的待显示界面,缩短终端设备的响应时延。

[0193] 但是,上述方式不能应用于用户滑动显示屏的场景。由于滑动过程中,点击一直存在,终端设备丢弃首次点击前的历史帧,并不能在用户的滑动中,减少显示延迟。并且,图8所示的方法直接丢弃缓存队列中全部的帧,还可能会造成界面显示中丢帧情况,导致画面显示不流畅。

[0194] 可以理解的是,在用户保持同一个方向滑动时,如果丢帧可能导致内容丢失,显示屏界面显示内容不连续,给用户造成画面跳跃的视觉效果,因此,在用户保持同一个方向滑动时,通常不进行丢帧处理,显示滞后的现象存在。

[0195] 但是,如果存在反向滑动操作,该反向滑动操作可以指用户执行了与之前的滑动操作方向相反的滑动操作,例如之前滑动操作方向是从上向下滑动,突然转变为从下向上滑动,则可以认为用户进行了反向滑动操作,由于反向滑动操作后的显示内容与反向滑动操作之前的内容存在重复,以及有些内容可能未显示,未显示的内容用户并不关注,则在用户执行反向滑动操作时,如果进行适应的丢帧,例如丢弃重复的帧和/或未显示的帧,不仅可以有效改善显示时延,还不会对用户的视觉造成跳跃。

[0196] 有鉴于此,本申请实施例提出一种数据处理方法,终端设备接收并响应用户在显示屏上的滑动操作,当终端设备接收到用户反向滑动操作时,通过丢弃缓存队列中的全部或部分帧以降低反向滑动后界面显示的反应时延。

[0197] 示例性的,图9为本申请实施例提供的一种滑动显示的数据处理方法流程示意图,以用户在设置列表界面中触摸滑动为例,如图9所示,该方法包括:

[0198] 终端设备中的触控面板接收到用户的滑动操作的报点信息,并将该滑动操作上报



至应用程序框架层的输入系统。

[0199] S901、终端设备输入用户滑动操作的报点信息。

[0200] 具体的,终端设备中的应用程序框架层的输入系统接收用户滑动操作的报点信息,并将报点信息发送至视图系统。

[0201] 本申请实施例中,报点用于指示用户操作的位置,方便终端设备判断界面显示的内容。报点可以是用户在触摸屏幕时,手指与屏幕的接触位置。或者用户通过鼠标等点击时鼠标在界面中的位置。或者用户使用手写笔触发屏幕时,手写笔与屏幕的接触位置。当用户手指、手写笔或者鼠标移动时,手指、手写笔与屏幕接触位置或者界面中鼠标位置也随之移动,报点位置改变。

[0202] 可能的实现方式中,S901之后,终端设备可以利用视图系统通过报点信息计算位移,以及根据图像绘制模块绘制图像,根据图像渲染模块渲染绘制后的图像,根据图像合成模块合成渲染后的图像,并对合成的图像进行显示。

[0203] 本申请实施例中,S901之后可以执行S902-S908。

[0204] S902、终端设备确定待显示界面是否为列表场景。

[0205] 具体的,终端设备中的视图系统可以确认待显示界面是否为列表场景。其中,列表场景可以理解为显示界面中包括列表控件的场景。

[0206] 本申请实施例中,列表控件为有序内容的集合,用于横向或纵向滑动,列表控件可以包括下述中的任一项:应用界面中的控件、设置界面中的控件、电子书阅读界面中的控件或文档界面中的控件,等。示例性的,列表控件可以是图3中的a所示的社交应用的界面中可以跟随用户手指滑动而滑动的部分,或在图3中的b所示的设置相关界面中可以跟随用户手指滑动而滑动的部分,图3中的c所示的文档界面中可以跟随用户手指滑动而滑动的部分,或者图3中的d所示的商品浏览界面中可以跟随用户手指滑动而滑动的部分。

[0207] 例如,视图系统根据焦点确认焦点对应的对象的实例化是否为列表控件,进而确认待显示界面是否为列表场景。本申请实施例中,列表场景可以包括能够接收上下滑动或者左右滑动的列表控件。例如,列表控件可以包括:朋友圈界面中用于横向或纵向滑动的控件、设置界面中用于横向或纵向滑动的控件、电子书阅读界面中用于横向或纵向滑动的控件和word文档界面中用于横向或纵向滑动的控件等。具体的,列表控件还可以分为横向列表控件或者纵向列表控件。

[0208] S903、终端设备确定用户操作是否为反向滑动。

[0209] 具体的,终端设备中的视图系统可以确定用户操作是否为反向滑动。

[0210] 具体的,终端设备中的视图系统可以计算报点的位移,并根据位移确定用户操作是否为反向滑动。

[0211] 例如,纵向列表控件中,如果基于报点的位移,得到纵坐标发生了反向变化,则可以认为存在反向滑动。横向列表控件中,如果基于报点的位移,得到横坐标发生了反向变化,则可以认为存在反向滑动。

[0212] S904、当终端设备确定用户操作为反向滑动时,对识别出反向滑动时正在进行绘制的帧增加标识。

[0213] 具体的,当设置应用通过应用程序框架层的视图系统确定用户操作为反向滑动时,设置应用可以调用系统库的图像绘制模块对识别出反向滑动时正在绘制的帧增加标

识。

[0214] 可以理解的是,当终端设备在列表场景中确定用户操作为反向滑动时,反向滑动操作后的显示内容与反向滑动操作之前的内容可能存在重复,终端设备的缓存中可能存在相同内容的待显示的帧。为识别出反向滑动时正在进行绘制的帧增加标识,则后续终端设备可以基于该标识执行对待合成的帧的丢弃等,进而可以减少终端设备反向滑动后的界面显示的响应时延。

[0215] S905、终端设备对增加标识的帧进行图像渲染。

[0216] 设置应用调用系统库的图像渲染模块对增加标识的帧进行渲染,并发送至显示合成进程。

[0217] S906、终端设备丢弃缓存中部分或全部帧。

[0218] 终端设备丢弃缓存中增加标识的帧和/或增加标识的帧之前的历史帧,可以缩短反向滑动操作后待显示界面的响应时延。其中,终端设备丢弃增加标识的帧之前的历史帧可能存在下述几种实现方式:

[0219] 可能的方式一,显示合成进程在确认缓存队列接收到帧时,判断接收的帧是否为增加标识的帧。如果接收到的帧是增加标识的帧,显示合成进程判断缓存队列中是否有缓存帧。如果缓存队列中有缓存帧,则丢弃缓存中全部或部分缓存帧。如果缓存队列中没有缓存帧,则不处理。如果接收到的不是增加标识的帧,显示合成进程不做处理,正常合成缓存队列中的缓存帧。

[0220] 可能的方式二,显示合成进程在从缓存队列中获取待合成图像帧时,先判断缓存队列中是否存在增加标识的帧。如果缓存队列中有增加标识的帧,终端设备丢弃缓存中全部或部分缓存帧。如果缓存队列中没有增加标识的帧,终端设备正常合成并显示该帧。

[0221] S907、终端设备对缓存队列中剩余的帧进行图像合成。

[0222] 示例性的,显示合成进程中的图像合成模块对缓存队列中剩余的帧进行图像合成。

[0223] 可能的实现方式中,每当周期信号到来时,显示合成进程从缓存队列中选择第二帧,对第二帧进行合成。第二帧为缓存队列中最早进入缓存队列的帧。

[0224] S908、终端设备通过LCD或LED显示增加标识的待显示界面。

[0225] 示例性的,设置应用调用内核层启动LCD/LED屏幕驱动,显示增加标识的待显示界面。综上,终端设备通过对报点的位移判断用户操作方向,对反向操作对应的界面增加标识,并丢弃缓存队列中部分或全部帧,缩短终端设备的响应时延,减小显示的滞后性。

[0226] 下面结合图10、图11、图12和图13所示的终端设备界面显示处理流程示意图,对通常的反向滑动操作时帧显示的处理方法与图9所示的方法进行对比说明。示例性的,在图10、图11和图12中,帧0、帧1、帧2、帧3、帧4、帧5和帧6所对应的内容为内容1、内容2、内容3、内容4、内容3、内容2和内容1。帧3为拐点。帧2和帧4分别在拐点前和拐点后,两者对应的内容相同。在帧4绘制时,帧1在合成,界面显示帧0。可以理解的是,帧0可以从帧1开始合成时显示,持续显示到帧1送显。

[0227] 图10为通常的反向滑动操作时帧显示的处理方法流程示意图。在图10中,当帧4绘制完成时,缓存队列中帧2、帧3和帧4待合成。由于未对缓存队列中的帧进行处理,帧1、帧2、帧3、帧4、帧5和帧6按顺序合成和显示,帧4、帧5和帧6的响应时延为4个vsync周期。界面显

示的内容依次为内容1、内容2、内容3、内容4、内容3、内容2和内容1。

[0228] 图11为图9所示的方法对应的丢弃全部缓存帧的滑动显示处理方法流程示意图。图11中,在帧4绘制时,终端设备识别到反向滑动,对帧4增加标识,此时帧1在合成,界面显示帧0。当帧4绘制完成时,缓存队列中帧2、帧3和帧4待合成。终端设备的缓存队列在接收到帧4时,显示合成进程检测到标识,丢弃缓存中增加标识的帧之前的历史帧(即帧2和帧3)。待帧1合成后,终端设备直接合成并显示帧4。终端设备的合成和显示顺序变为帧1、帧4、帧5和帧6。相较于图10的处理方法,图11中帧4和帧5的响应时延降低,变为2个vsync周期。界面显示的内容依次为内容1、内容2、内容3、内容2和内容1,显示不会有跳跃。

[0229] 图12为图9所示的方法对应的丢弃部分缓存帧的滑动显示处理方法流程示意图。图11中,在帧4绘制时,终端设备识别到反向滑动,对帧4增加标识,此时帧1在合成,界面显示帧0。当帧4绘制完成时,缓存队列中帧2、帧3和帧4待合成。终端设备的显示合成进程在接收到帧4时,检测到标识,丢弃未合成的帧3。待帧1合成后,终端设备合成帧2、帧4、帧5和帧6。终端设备的合成和显示顺序变为帧1、帧2、帧4、帧5和帧6。相较于图10的处理方法,图12中帧4和帧5的响应时延降低,变为3个vsync周期。界面显示的内容依次为内容1、内容2、内容3、内容3、内容2和内容1,显示内容连续,不会有跳跃。

[0230] 图13为图9所示的方法对应的丢弃部分缓存帧的滑动显示处理方法流程示意图。图11中,在帧4绘制时,终端设备识别到反向滑动,对帧4增加标识,此时帧1在合成,界面显示帧0。当帧4绘制完成时,缓存队列中帧2、帧3和帧4待合成。终端设备的显示合成进程在接收到帧4时,检测到标识,丢弃增加标识的帧和增加标识的帧之前的历史帧(即帧3和帧4)。待帧1合成后,终端设备合成帧2、帧5和帧6。终端设备的合成和显示顺序变为帧1、帧2、帧5和帧6。相较于图10的处理方法,图12中帧5的响应时延降低,变为2个vsync周期。界面显示的内容依次为内容1、内容2、内容3、内容2和内容1,显示内容连续,不会有跳跃。

[0231] 下面结合图14对本申请实施例提供的滑动显示方法具体实现进行描述。

[0232] 图14为本申请实施例提供的滑动显示方法流程示意图。

[0233] S1401、终端设备在显示屏上接收滑动操作。

[0234] 可能的实现方式中,终端设备可以接收用户通过手或者手写笔在触摸屏上的滑动操作。或者,终端设备可以接收用户通过鼠标滚动或者拖拽的滑动操作。

[0235] S1402、终端设备基于滑动操作确定反向滑动事件发生。

[0236] 本申请实施例中,反向滑动事件是指滑动方向发生改变的操作对应的事件。示例性的,在横向列表场景中,反向滑动可以是左往右滑变为从右往左滑,或者从右往左滑变为从左往右滑。在纵向列表场景中,反向滑动可以是下往上滑变为从上往下滑,或者从上往下滑变为从下往上滑。可以理解的是,终端设备中有纵向列表和横向列表两种使用场景。终端设备需要对列表方向进行判断,进而确定滑动方向。示例性的,终端设备通过对对象的实例化判断当前场景是否为列表场景。比如,终端设备通过判断对象的实例化是否是列表视图(listview)或循环视图(recyclerview)等列表控件,如果对象的实例化是列表视图(listview)或循环视图(recyclerview)等列表控件,则确定是列表场景。

[0237] 或者,终端设备可以通过判断当前视图的布局,判断列表方向。示例性的,终端设备可以通过接口函数或变量值,获取当前视图的布局,以判断列表方向是横向或是纵向。接口函数可以是getOrientation()和变量值mOrientation等。

[0238] 示例性的,终端设备通过接口函数的参数返回值判断列表方向。

[0239] 可能的实现方式一中,接口函数为getOrientation(),当函数返回值为0时表示横向列表,函数返回值为1时表示纵向列表。或者,当函数返回值为1时表示横向列表,函数返回值为0时表示纵向列表。

[0240] 可能的方式二中,通过变量值mOrientation判断,当变量值为0时表示横向列表,变量值为1时表示纵向列表。或者,当变量值为1时表示横向列表,变量值为0时表示纵向列表。

[0241] 当列表方向确定后,终端设备可以通过滑动操作的位置信息例如触点坐标等,确定反向滑动事件。示例性的,终端设备可以计算滑动操作前后的手指触屏位置,确定滑动方向,进而确定反向滑动事件。

[0242] S1403、终端设备丢弃缓存队列中的全部或部分帧;全部或部分帧与反向滑动事件对应的位置有关。

[0243] 可以理解的是,终端设备在进行界面合成的过程中,可能由于系统负载或其他原因导致某帧合成较慢,导致新的帧缓存在缓存队列中。在确定反向滑动事件后,缓存队列中的帧可能存在显示内容相同的界面,以及有些内容可能未显示,未显示的内容用户并不关注,终端设备有选择丢弃全部或部分帧,例如丢弃重复的帧和/或未显示的帧,可以在保障界面显示的连贯性,降低终端设备界面显示的响应时延。

[0244] 可能的实现方式一中,因为丢弃帧过多可能造成画面显示不连续,终端设备在确定反向滑动事件后,可以根据帧率动态配置可丢弃的最多缓存数量。例如,当显示的帧率为60pfs时,缓存队列中最多存在3个帧,最多丢弃缓存中反向滑动事件的位置附近的2个帧或1个帧等。当显示的帧率高于60pfs时,缓存队列中最多可达5个帧,全部丢弃或者设置丢弃缓存中反向滑动事件的位置附近的4个帧或3个帧等。

[0245] 这样,终端设备可以在满足画面显示连续的情况下,尽可能的降低反向滑动后待显示界面的响应时延。

[0246] 可能的实现方式二中,终端设备根据帧的内容确定丢弃缓存中的部分帧。示例性的,帧可以具有权重,终端设备可以优先丢弃当前放入缓存队列中的权重较低的帧,等。

[0247] 这样,终端设备可以保障丢弃帧后,重要的帧能够得到显示。

[0248] 可能的实现方式三中,终端设备也可以丢弃反向滑动事件发生时,缓存中的全部帧,尽可能的降低反向滑动后待显示界面的响应时延。

[0249] S1404、终端设备显示待显示帧。

[0250] 可能的实现方式中,待显示帧可以包括终端设备在执行S1403的丢弃动作时已合成的帧。终端设备显示已合成的帧,这样,终端设备可以不用等待该已合成的帧的合成时间,进一步降低反向滑动后待显示界面的响应时延。

[0251] 待显示帧也可以为终端设备执行丢弃动作后得到的待显示的帧。例如,如果终端设备丢弃缓存队列中的部分帧,则待显示帧可以包括丢弃缓存队列中的部分帧后,缓存队列中剩余的帧,如果终端设备丢弃缓存队列中的全部帧,则待显示帧可以包括丢弃缓存队列中的全部帧后,缓存队列中新增加的渲染后的帧。

[0252] 可能的实现方式中,每当周期信号到来时,显示合成进程从缓存队列中选择第二帧,对第二帧进行合成。第二帧为缓存队列中最早进入缓存队列的帧。

[0253] 综上,终端设备在确定用户的反向滑动操作后,通过丢弃缓存中的全部或部分帧,缩短待显示界面的响应时延。

[0254] 在图14对应的实施例的基础上,上述S1402可以包括S14001、S14002和S14003。

[0255] S14001、终端设备获取滑动操作在显示屏的位置信息(即报点信息)。

[0256] 示例性的,当终端设备中包括触摸屏时,终端设备可以通过传感器获取用户通过手或者手写笔的位置,进而获取滑动操作在显示屏的位置信息。当终端设备包括鼠标时,终端设备可以通过获取鼠标位置,获取滑动操作在显示屏的位置信息。

[0257] S14002、终端设备确定滑动操作对应的界面包括列表控件。

[0258] S14003、根据滑动操作在显示屏的位置信息的变动情况,确定反向滑动事件发生。

[0259] 可以理解的是,上述终端设备中列表控件可以对应纵向列表和横向列表。

[0260] 可能的实现方式一中,在滑动操作作用于显示屏中的纵向列表时,通过滑动操作在显示屏的坐标信息,可以反映反向滑动事件发生,例如,滑动操作对应的相邻两帧的纵坐标差值(或理解为纵坐标位移)从正值变为负值,或者从负值变为正值的情况下,确定反向滑动事件发生。

[0261] 可能的实现方式二中,在滑动操作作用于显示屏中的横向列表,通过滑动操作在显示屏的坐标信息,可以反映反向滑动事件发生,例如,滑动操作对应的相邻两帧的横坐标差值(或理解为横坐标位移)从正值变为负值,或者从负值变为正值的情况下,确定反向滑动事件发生。这样,终端设备可以根据滑动操作的位置信息精确判断终端设备的反向滑动事件。

[0262] 可能的实现方式三中,终端设备根据滑动方向确定反向滑动事件的发生。

[0263] 下面对终端设备根据滑动方向确定反向滑动操作的方法进行说明。示例性的,如图15所示,该方法包括:

[0264] S1501、终端设备接收报点输入。

[0265] 其中,报点输入可以是用户在屏幕中的滑动(move)事件产生的。

[0266] S1502、终端设备判断列表方向。

[0267] 当列表为横向时,终端设备通过执行S1503-S1506确定反向滑动事件发生。当列表为纵向时,终端设备执行S1507-S1510,确定反向滑动事件发生。

[0268] 下面对S1503-S1506进行说明。

[0269] S1503、终端设备计算横坐标位移。

[0270] 示例性的,计算方式可以为 $dx = newX - lastX$ 或者 $dx = lastX - newX$

[0271] S1504、终端设备根据横坐标位移正负确定当前方向。

[0272] 可以理解的是,终端设备通过滑动操作对应的报点坐标变化识别滑动事件。横向列表场景中,滑动操作的方向与横坐标位移变化相对应。示例性的,终端设备根据 $dx = newX - lastX$ 计算的横坐标位移为正或者根据 $dx = lastX - newX$ 计算的横坐标位移为负时,滑动操作的当前方向为向右滑动。终端设备根据 $dx = newX - lastX$ 计算的横坐标位移为负或者根据 $dx = lastX - newX$ 计算的横坐标位移为正时,滑动操作的当前方向为向左滑动。

[0273] S1505、判断是否与历史方向相同。

[0274] 历史方向为上一次终端设备确定的滑动方向。当S1504中的当前方向与历史方向相同时,终端设备确定反向滑动事件未发生。当S1504中的当前方向与历史方向不同时,终

端设备确定反向滑动事件发生。这样,终端设备可以准确确定反向滑动事件,进而基于反向滑动事件降低终端设备界面显示的响应时延。

[0275] S1506、若方向不同,终端设备确定反向滑动事件发生,执行上述S1403-S1404。

[0276] 下面对S1507-S1510进行说明。

[0277] S1507、终端设备计算纵坐标位移。

[0278] 示例性的,计算方式可以为 $dy = newY - lastY$ 或者 $dy = lastY - newY$ 。

[0279] S1508、根据纵坐标位移正负确定当前方向。

[0280] 可以理解的是,纵向列表场景中,滑动操作的方向与纵坐标位移变化相对应。示例性的,终端设备根据 $dy = newY - lastY$ 计算的横坐标位移为正或者根据 $dy = lastY - newY$ 计算的横坐标位移为负时,滑动操作的当前方向为向上滑动。终端设备根据 $dy = newY - lastY$ 计算的横坐标位移为负或者根据 $dy = lastY - newY$ 计算的横坐标位移为正时,滑动操作的当前方向为向下滑动。

[0281] S1509、判断是否与历史方向相同。

[0282] S1510、若方向不同,终端设备确定反向滑动事件发生,执行上述S1403-S1404。

[0283] S1511、终端设备更新当前滑动方向。

[0284] 这样,终端设备可以基于滑动方向,重新判定是否有新的反向滑动操作事件发生。在上述S1505、S1506、S1509或S1510,终端设备执行S1511更新该滑动方向为当前滑动方向。

[0285] 在图14对应的实施例的基础上,上述S1403可以包括,S14004、S14005和S14006。

[0286] S14004、在基于滑动操作确定反向滑动事件发生时,为正在绘制的帧添加标识。

[0287] 示例性的,终端设备在正在绘制的帧添加一个整形变量,例如,int变量中的 $flag = 1$ 。

[0288] S14005、在合成队列中识别到添加标识的帧。

[0289] 可能的实现方式中,终端设备通过对标识进行解析,识别添加标识的帧。

[0290] S14006、在第一帧传入合成队列时,丢弃部分或全部帧。第一帧为反向滑动事件发生时,正在绘制的帧,即添加了标识的帧。

[0291] 本申请实施例中,合成队列可以称为缓存队列。

[0292] 可能的实现方式一中,终端设备在第一帧传入合成队列时,终端设备丢弃第一帧之前的帧。

[0293] 示例性的,帧1-帧9为连续的显示界面,帧1-帧9对应的显示内容分别是内容1、内容2、内容3、内容4、内容5、内容4、内容3、内容2和内容1;根据帧6对应用户滑动报点的位移,帧5为拐点,正在绘制的为帧6,为帧6添加标识,记作第一帧,第一帧之前的帧为帧1-帧5。当帧6传入缓存队列时,缓存队列中有帧3、帧4和帧5。此时,终端设备可以丢弃帧4和帧5,待显示帧为帧3和帧6。待显示帧为缓存队列中剩余的帧,即帧3和帧6,第二帧为帧3。终端设备先合成显示帧3,再合成显示帧6。终端设备的显示顺序为帧1、帧2、帧3、帧6、帧7、帧8和帧9,显示屏显示内容为内容1、内容2、内容3、内容4、内容3、内容2和内容1。可能的实现方式二中,终端设备在第一帧传入合成队列时,终端设备丢弃第一帧,以及第一帧之前的帧。

[0294] 示例性的,帧1-帧9为连续的显示界面,帧1-帧9对应的显示内容分别是内容1、内容2、内容3、内容4、内容5、内容4、内容3、内容2和内容1;根据帧6对应用户滑动报点的位移,帧6为第一帧,第一帧之前的帧为帧1-帧5。当帧6传入缓存队列时,缓存队列中有帧3、帧4和

帧5。此时，终端设备丢弃帧6和帧5，缓存队列中有帧3和帧4，待显示帧为缓存队列中剩余的帧，即帧3和帧4，第二帧为帧3。终端设备先合成显示帧3，再合成显示帧4。终端设备的显示顺序为帧1、帧2、帧3、帧4、帧7、帧8和帧9，显示屏显示内容为内容1、内容2、内容3、内容4、内容3、内容2和内容1。

[0295] 示例性的，帧1-帧9为连续的显示界面，帧1-帧9对应的显示内容分别是内容1、内容2、内容3、内容4、内容5、内容4、内容3、内容2和内容1；根据帧6对应用户滑动报点的位移，帧6为第一帧，第一帧之前的帧为帧1-帧5。当帧6传入缓存队列时，缓存队列中有帧3、帧4和帧5。此时，终端设备丢弃帧3、帧4、帧5和帧6，缓存队列中没有缓存帧。待显示帧为缓存队列中新增加的渲染后的帧，即帧7。终端设备的显示顺序为帧1、帧2、帧7、帧8和帧9，显示屏显示内容为内容1、内容2、内容3、内容2和内容1。

[0296] 可能的实现方式三中，终端设备在第一帧传入合成队列时，终端设备丢弃第一帧前的一帧。

[0297] 示例性的，帧1-帧9为连续的显示界面，帧1-帧9对应的显示内容分别是内容1、内容2、内容3、内容4、内容5、内容4、内容3、内容2和内容1；根据帧6对应用户滑动报点的位移，帧6为第一帧，第一帧前的一帧为帧5。当帧6传入缓存队列时，缓存队列中有帧3、帧4和帧5。此时，终端设备丢弃帧5，缓存队列中有帧3、帧4和帧6，待显示帧为缓存队列中剩余的帧，即帧3、帧4和帧6，第二帧为帧3。终端设备先合成显示帧3，再合成显示帧4，然后合成显示帧6。终端设备的显示顺序为帧1、帧2、帧3、帧4、帧6、帧7、帧8和帧9，显示屏显示内容为内容1、内容2、内容3、内容4、内容4、内容3、内容2和内容1。

[0298] 这样，丢弃的内容与当前放入的帧内容相连，或者丢弃的内容是之前还没有显示的内容，可以保障终端设备界面显示的连贯性。

[0299] 上面已对本申请实施例的设备数据处理方法进行了说明，下面对本申请实施例提供的执行上述设备数据处理方法的终端设备进行描述。本领域技术人员可以理解，方法和装置可以相互结合和引用，本申请实施例提供的终端设备可以执行上述设备回连方法中的步骤。

[0300] 如图16所示，图16示出了本申请实施例提供的一种数据处理装置的结构示意图，该数据处理装置可以是本申请实施例中的终端设备。该数据处理装置包括：触摸屏1601，其中，触摸屏包括触控面板和显示屏；一个或多个处理器1602；存储器1603；多个应用程序；以及一个或多个计算机程序，其中一个或多个计算机程序被存储在存储器1603中，一个或多个计算机程序包括指令，当指令被数据处理装置执行时，使得数据处理装置执行以下步骤：接收用户输入的滑动操作；确定滑动操作对应的控件为列表控件；确定反向滑动事件发生，反向滑动事件为滑动操作的滑动方向发生反向的事件；在反向滑动事件发生时，为第一帧添加标识，第一帧为反向滑动事件发生时正在绘制的帧；在缓存队列中存在携带标识的第一帧的情况下，丢弃缓存队列中的部分或全部帧，缓存队列用于存储绘制完成的帧，绘制完成的帧与滑动操作相对应；合成待显示帧，待显示帧为执行丢弃动作后得到的待显示的帧；显示待显示帧。

[0301] 可选的，数据处理装置具体用于执行以下步骤：确认缓存队列接收到携带标识的第一帧时，丢弃缓存队列中部分或全部帧；或者，从缓存队列获取待合成的帧时，丢弃缓存队列中部分或全部帧。

[0302] 可选的,数据处理装置具体用于执行以下步骤:在周期信号到来时,确定缓存队列中是否包含有携带标识的第一帧;在缓存队列中包含有携带标识的第一帧的情况下,丢弃缓存队列中部分或全部帧。

[0303] 可选的,数据处理装置具体用于执行以下步骤:丢弃缓存队列中的M帧,其中,M小于或等于N,N为预设的常量;或者,丢弃缓存队列中的位于携带标识的第一帧之前的帧;或者,丢弃携带标识的第一帧以及缓存队列中的位于携带标识的第一帧之前的帧;或者,丢弃缓存队列中的全部帧。

[0304] 可选的,当数据处理装置丢弃的帧为缓存队列中的部分帧时,待显示帧包括丢弃缓存队列中的部分帧后,缓存队列中剩余的帧;或者,当数据处理装置丢弃的帧为缓存队列中的全部帧时,待显示帧包括丢弃缓存队列中的全部帧后,缓存队列中新增加的渲染后的帧。

[0305] 可选的,数据处理装置包括图像绘制模块、图像渲染模块和图像合成模块,数据处理装置还用于响应于滑动操作,图像绘制模块用于对滑动操作对应的帧进行绘制;图像渲染模块用于对绘制后的帧进行渲染;图像渲染模块用于将渲染后的帧存储在缓存队列;每当周期信号到来时,数据处理装置用于从缓存队列中选择第二帧;数据处理装置通过图像合成模块对第二帧进行合成。

[0306] 可选的,终端设备还用于执行以下步骤:确定滑动操作对应的控件为列表控件之后,确定列表控件的方向属性,列表控件的方向属性包括:横向和纵向。

[0307] 可选的,终端设备包括视图系统,视图系统用于确定滑动操作对应的对象的实例化为列表控件;视图系统用于根据对象使用的变量值或接口函数的返回值,确定列表控件的方向属性为横向或纵向;其中,接口函数包括getOrientation(),变量值包括mOrientation。

[0308] 可选的,终端设备具体用于执行以下步骤:在列表控件的方向属性为横向时,根据滑动操作产生的横向位移确定滑动操作是否发生反向;或者,在列表控件的方向属性为纵向时,根据滑动操作产生的纵向位移确定滑动操作是否发生反向。

[0309] 可选的,终端设备,具体用于执行以下步骤:在列表控件的方向属性为横向时,计算滑动操作产生的横向位移是否发生由负值变为正值,或由正值变为负值的变化;当滑动操作产生的横向位移发生由负值变为正值,或由正值变为负值的变化时,确定反向滑动事件发生;或者,在列表控件的方向属性为纵向时,计算滑动操作产生的纵向位移是否发生由负值变为正值,或由正值变为负值的变化;当滑动操作产生的纵向位移发生由负值变为正值,或由正值变为负值的变化时,确定反向滑动事件发生。

[0310] 可选的,视图系统还用于计算滑动操作的横向位移或纵向位移。

[0311] 可选的,列表控件为有序内容的集合,用于横向或纵向滑动,列表控件包括下述中的任一项:应用界面中的控件、设置界面中的控件、电子书阅读界面中的控件或文档界面中的控件。

[0312] 图17为本申请实施例提供的一种数据处理装置的硬件结构示意图。请参见图17,该装置包括:存储器1701、处理器1702和接口电路1703。该装置还可以包括显示屏1704,其中,存储器1701、处理器1702、接口电路1703和显示屏1704可以通信;示例性的,存储器1701、处理器1702、接口电路1703和显示屏1704可以通过通信总线通信,存储器1701用于存



储计算机执行指令,由处理器1702来控制执行,并由接口电路1703来执行通信,从而实现本申请下述实施例提供的数据处理方法。

[0313] 可能的实现方式中,本申请实施例中的计算机执行指令也可以称之为应用程序代码,本申请实施例对此不作具体限定。

[0314] 可选的,接口电路1703还可以包括发送器和/或接收器。可选的,上述处理器1702可以包括一个或多个CPU,还可以是其他通用处理器、数字信号处理器(digital signal processor,DSP)、专用集成电路(application specific integrated circuit,ASIC)等。通用处理器可以是微处理器或者该处理器也可以是任何常规的处理器等。结合本申请所公开的方法的步骤可以直接体现为硬件处理器执行完成,或者用处理器中的硬件及软件模块组合执行完成。

[0315] 本申请实施例提供一种终端设备,终端设备用于接收用户输入的滑动操作;终端设备还用于确定反向滑动事件发生,反向滑动事件为滑动操作的滑动方向发生反向的事件;终端设备还用于丢弃缓存队列中的部分或全部帧,缓存队列用于存储绘制完成的帧,绘制完成的帧与滑动操作相对应;终端设备还用于显示待显示帧,待显示帧为执行丢弃动作后得到的待显示的帧。

[0316] 可选的,终端设备具体用于,在缓存队列中存在第一帧时,丢弃缓存队列中的部分或全部帧,第一帧为反向滑动事件发生时正在绘制的帧。

[0317] 可选的,终端设备包括显示合成进程,显示合成进程用于确认缓存队列接收到携带标识的第一帧时,丢弃缓存队列中部分或全部帧;或者,显示合成进程用于从缓存队列获取待合成的帧时,丢弃缓存队列中部分或全部帧。

[0318] 可选的,每当周期信号到来时,显示合成进程具体用于确定缓存队列中是否包含有携带标识的第一帧;当缓存队列中包含有第一帧的情况下,显示合成进程具体用于丢弃缓存队列中部分或全部帧。

[0319] 可选的,丢弃缓存队列中的部分或全部帧,包括:丢弃缓存队列中的M帧,其中,M小于或等于N,N为预设的常量;或者,丢弃缓存队列中的位于第一帧之前的帧;或者,丢弃第一帧以及缓存队列中的位于第一帧之前的帧;或者,丢弃缓存队列中的全部帧。

[0320] 本实施例的装置对应地可用于执行上述方法实施例中执行的步骤,其实现原理和技术效果类似,此处不再赘述。

[0321] 本申请实施例还提供了一种计算机可读存储介质。上述实施例中描述的方法可以全部或部分地通过软件、硬件、固件或者其任意组合来实现。如果在软件中实现,则功能可以作为一个或多个指令或代码存储在计算机可读介质上或者在计算机可读介质上传输。计算机可读介质可以包括计算机存储介质和通信介质,还可以包括任何可以将计算机程序从一个地方传送到另一个地方的介质。存储介质可以是可由计算机访问的任何目标介质。

[0322] 一种可能的实现方式中,计算机可读介质可以包括RAM,ROM,只读光盘(compact disc read-only memory,CD-ROM)或其它光盘存储器,磁盘存储器或其它磁存储设备,或目标于承载的任何其它介质或以指令或数据结构的形式存储所需的程序代码,并且可由计算机访问。而且,任何连接被适当地称为计算机可读介质。例如,如果使用同轴电缆,光纤电缆,双绞线,数字用户线(Digital Subscriber Line,DSL)或无线技术(如红外,无线电和微波)从网站,服务器或其它远程源传输软件,则同轴电缆,光纤电缆,双绞线,DSL或诸如红

外,无线电和微波之类的无线技术包括在介质的定义中。如本文所使用的磁盘和光盘包括光盘,激光盘,光盘,数字通用光盘(Digital Versatile Disc,DVD),软盘和蓝光盘,其中磁盘通常以磁性方式再现数据,而光盘利用激光光学地再现数据。上述的组合也应包括在计算机可读介质的范围内。

[0323] 本申请实施例是参照根据本申请实施例的方法、设备(系统)、和计算机程序产品的流程图和/或方框图来描述的。应理解可由计算机程序指令实现流程图和/或方框图中的每一流程和/或方框、以及流程图和/或方框图中的流程和/或方框的结合。可提供这些计算机程序指令到通用计算机、专用计算机、嵌入式处理机或其他可编程数据处理设备的处理单元以产生一个机器,使得通过计算机或其他可编程数据处理设备的处理单元执行的指令产生用于实现在流程图一个流程或多个流程和/或方框图一个方框或多个方框中指定的功能的装置。

[0324] 以上的具体实施方式,对本发明的目的、技术方案和有益效果进行了进一步详细说明,所应理解的是,以上仅为本发明的具体实施方式而已,并不用于限定本发明的保护范围,凡在本发明的技术方案的基础之上,所做的任何修改、等同替换、改进等,均应包括在本发明的保护范围之内。

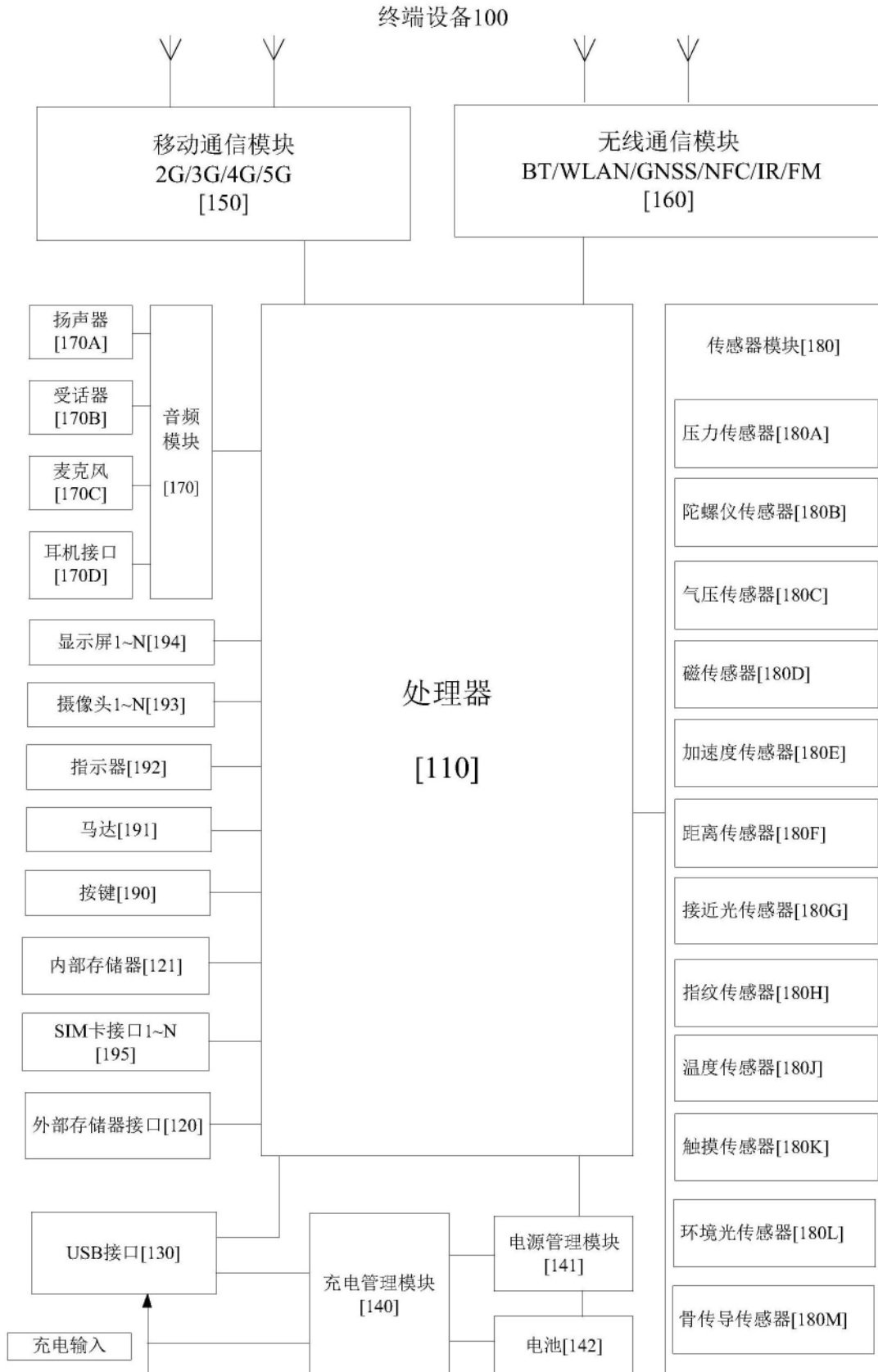


图1

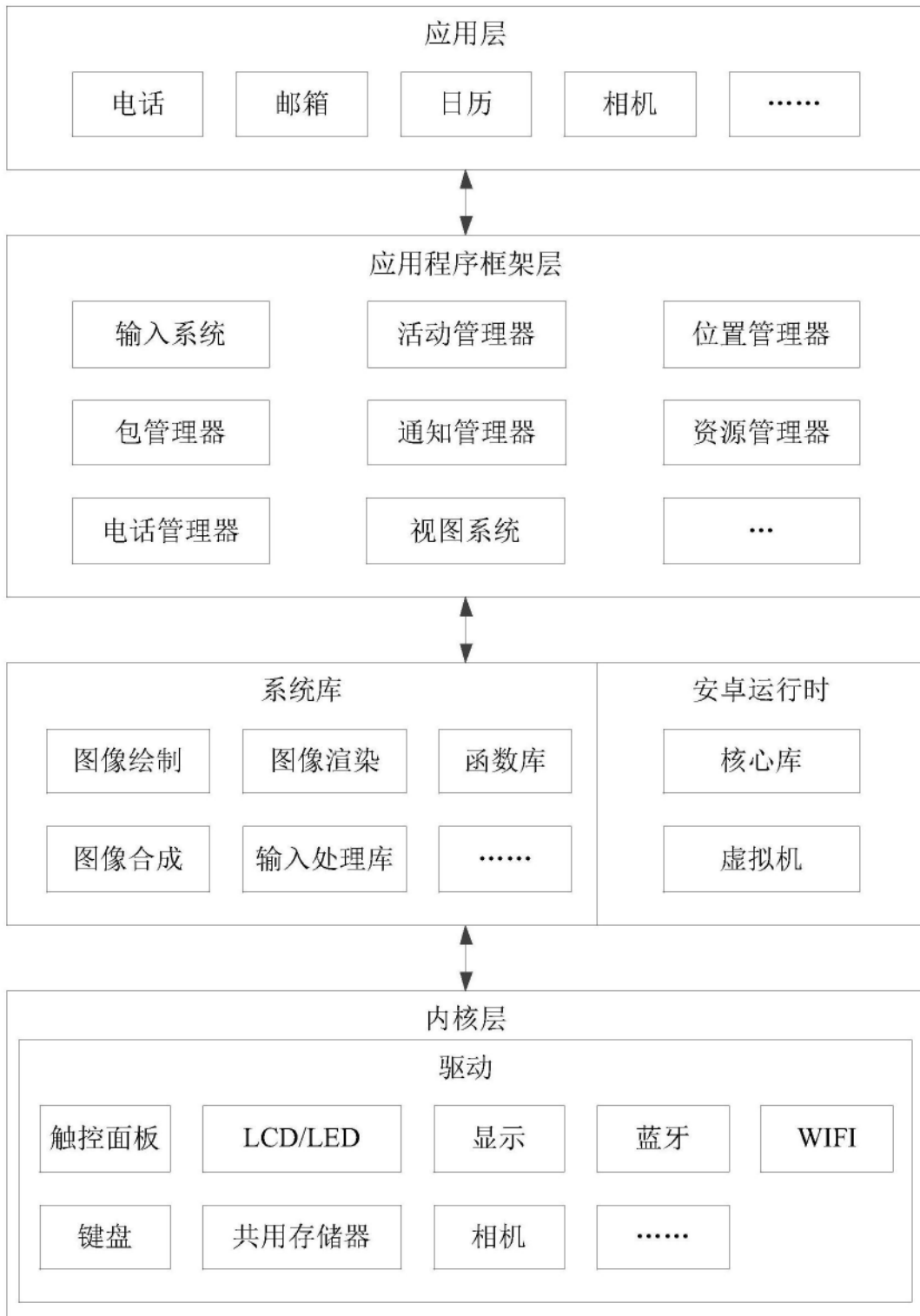


图2



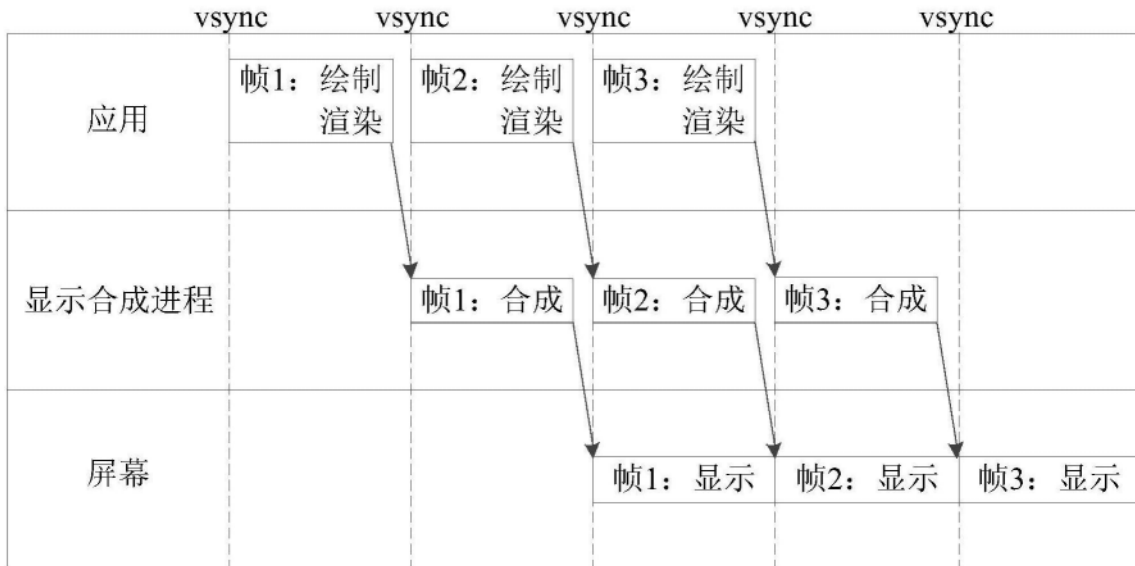


图4

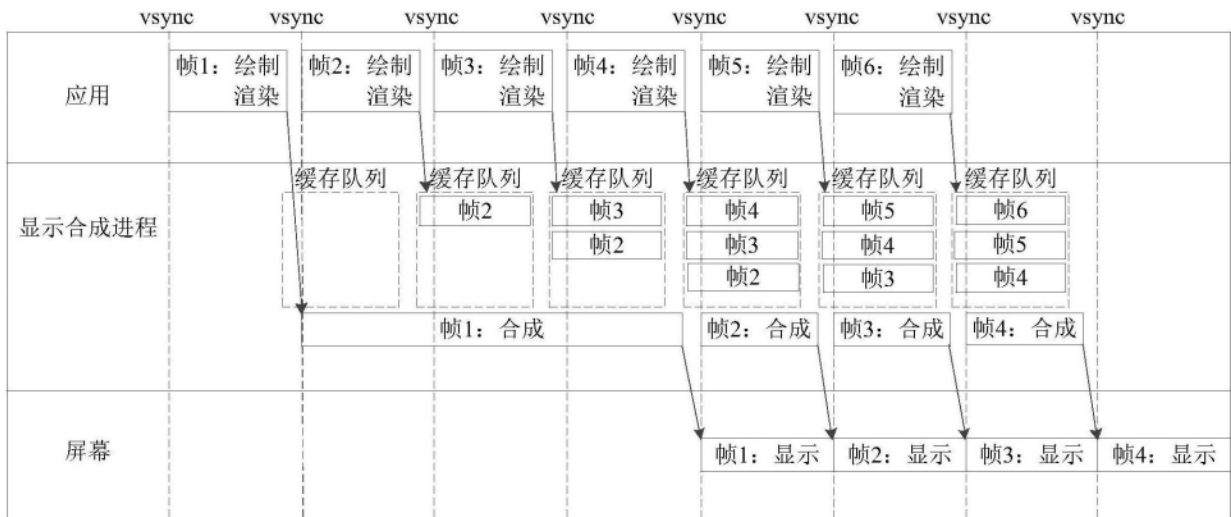


图5

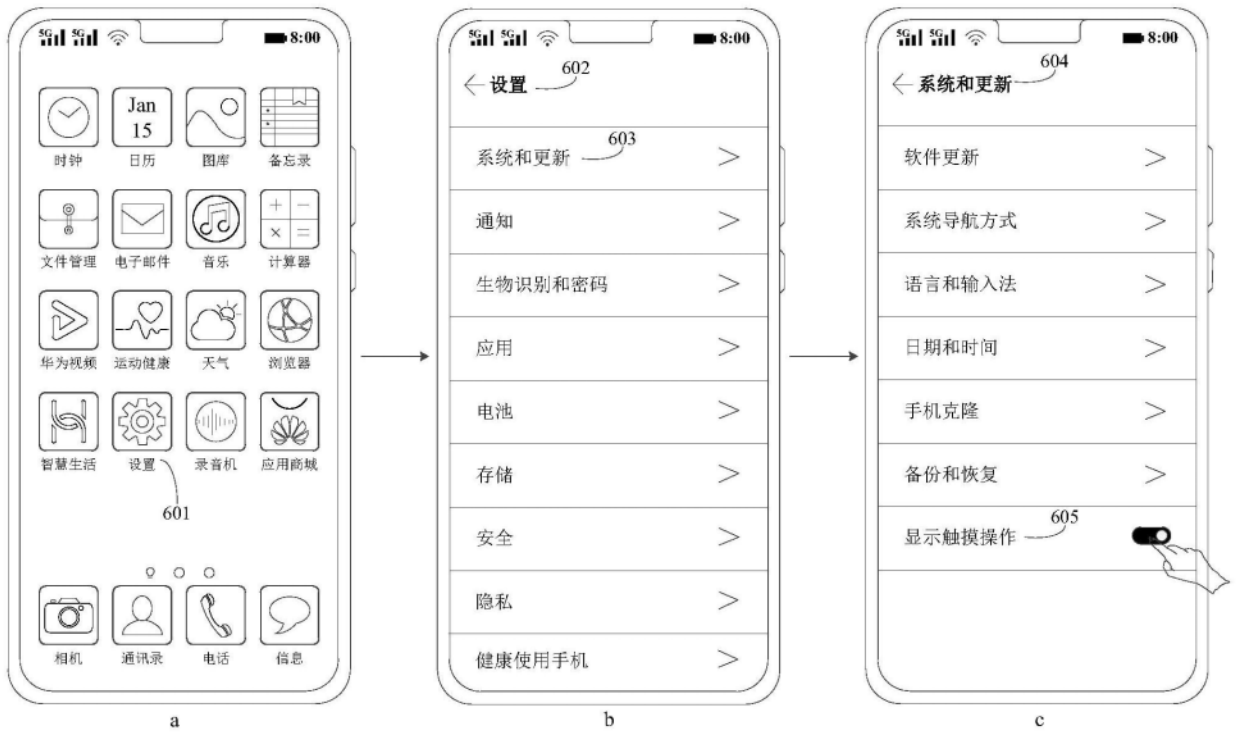


图6

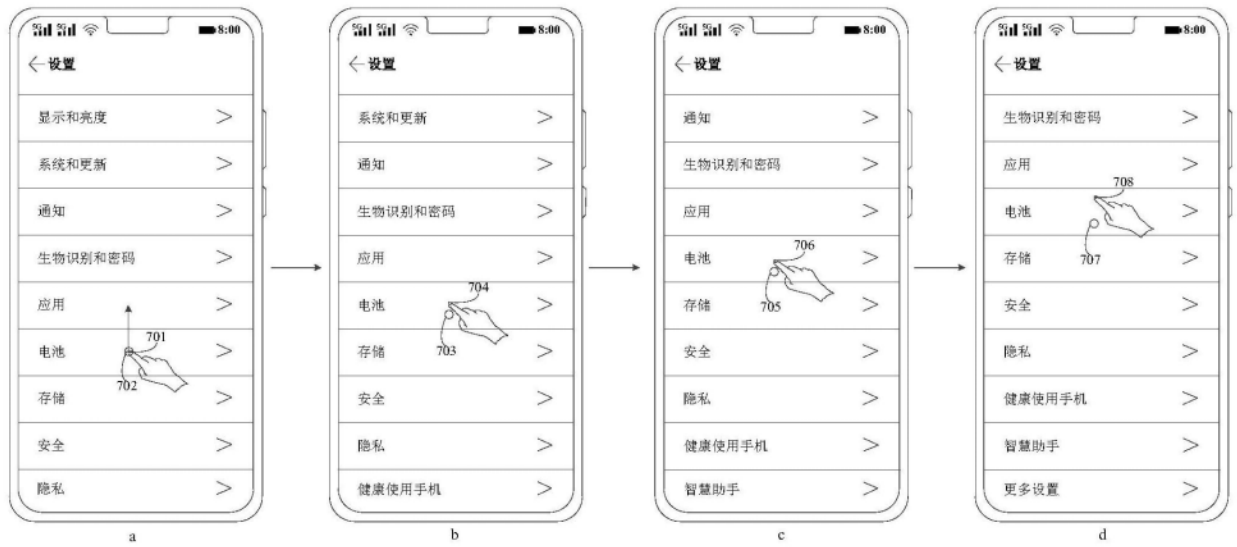


图7

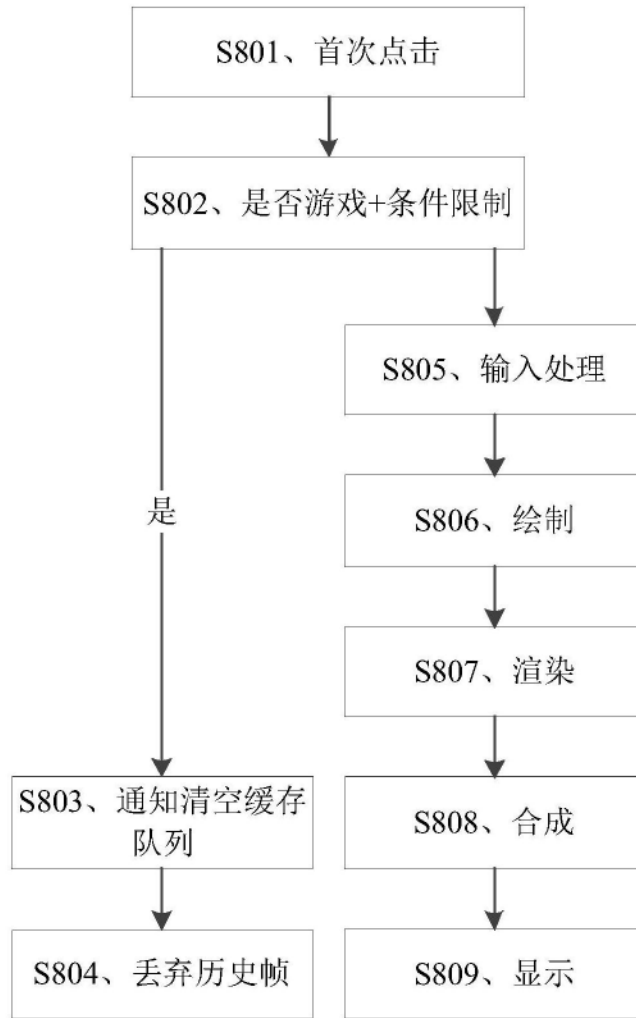


图8



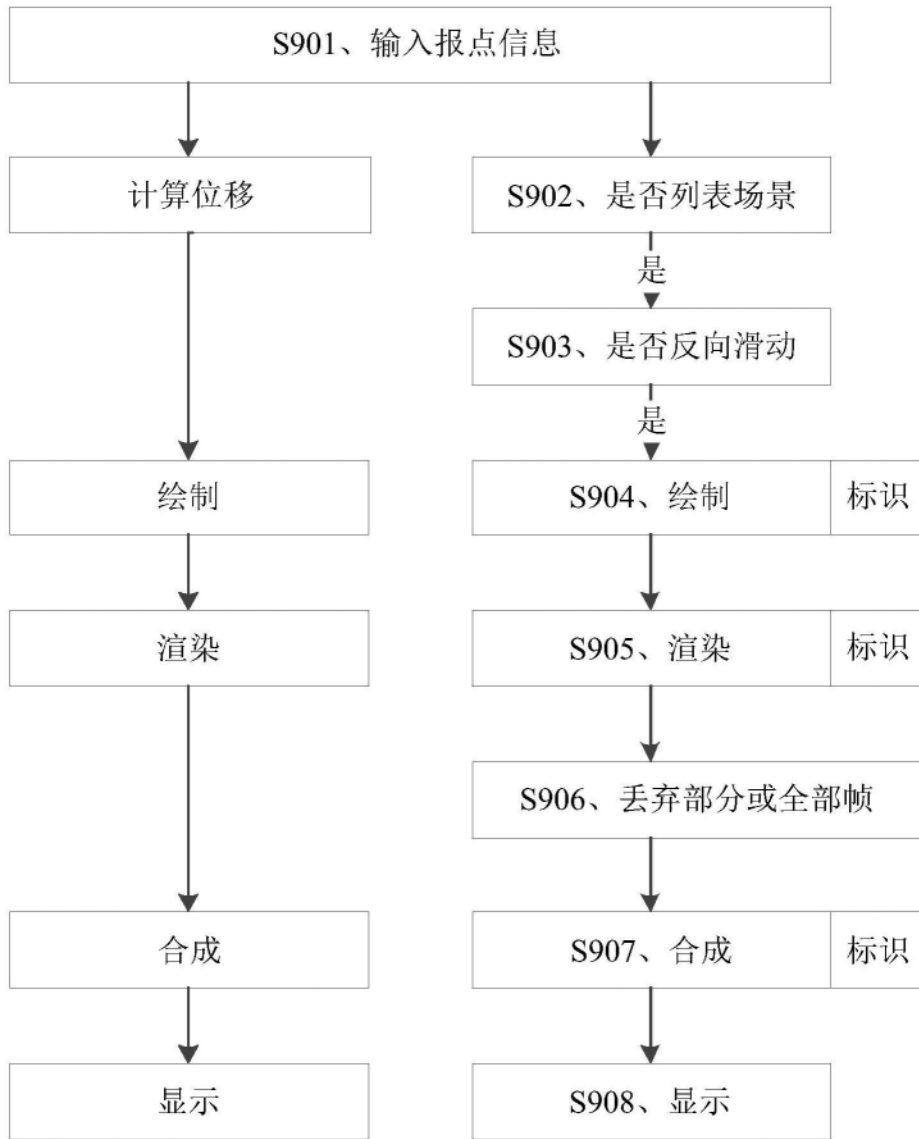


图9

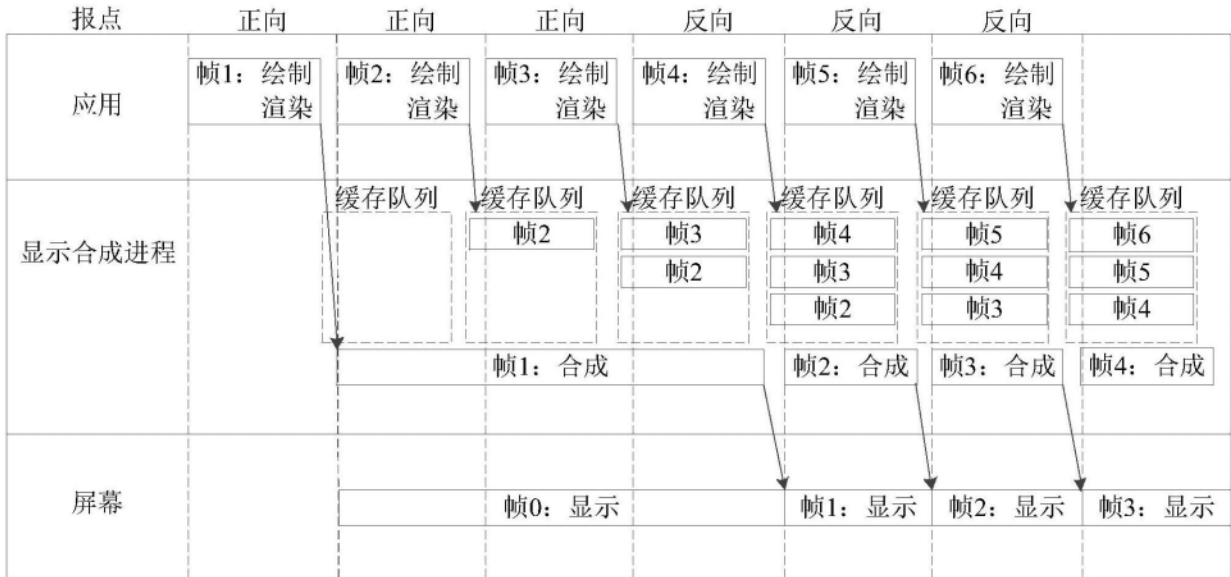


图10

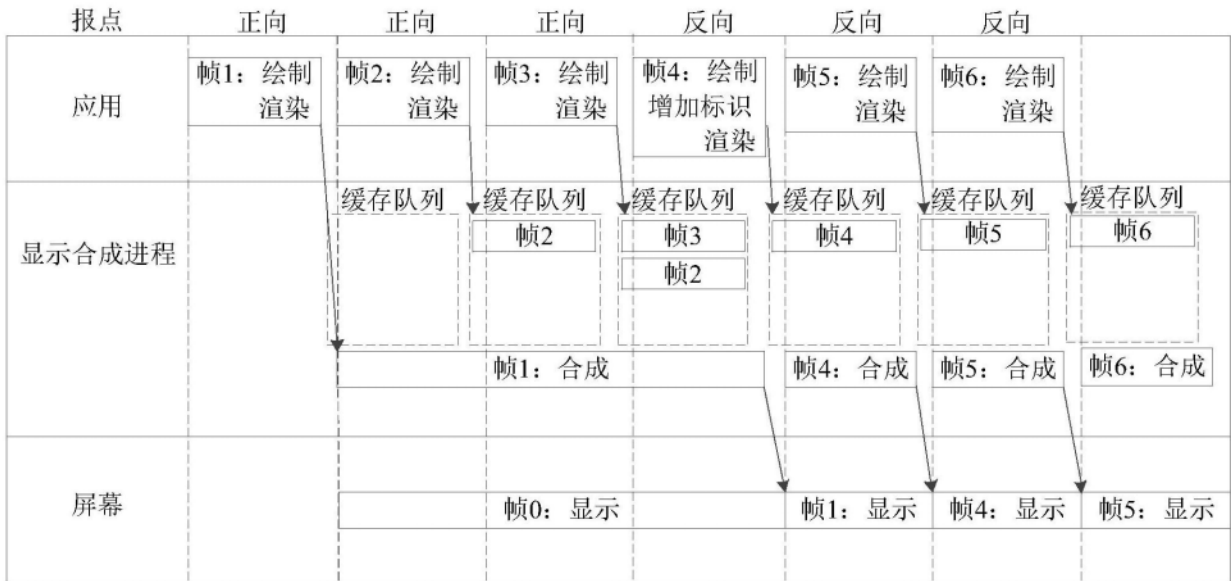


图11

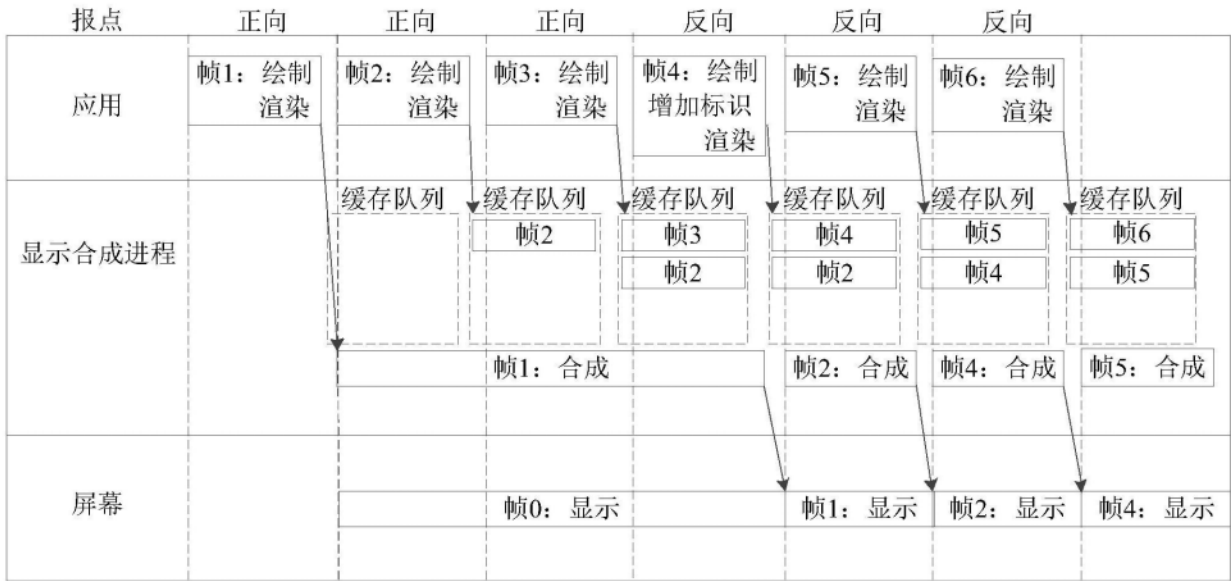


图12

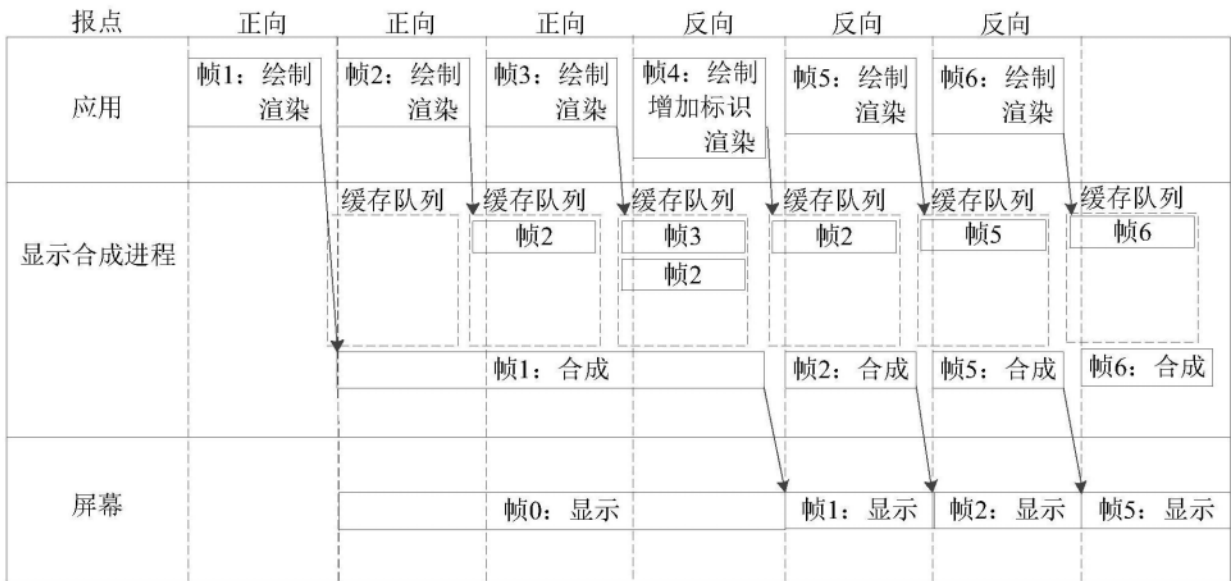


图13

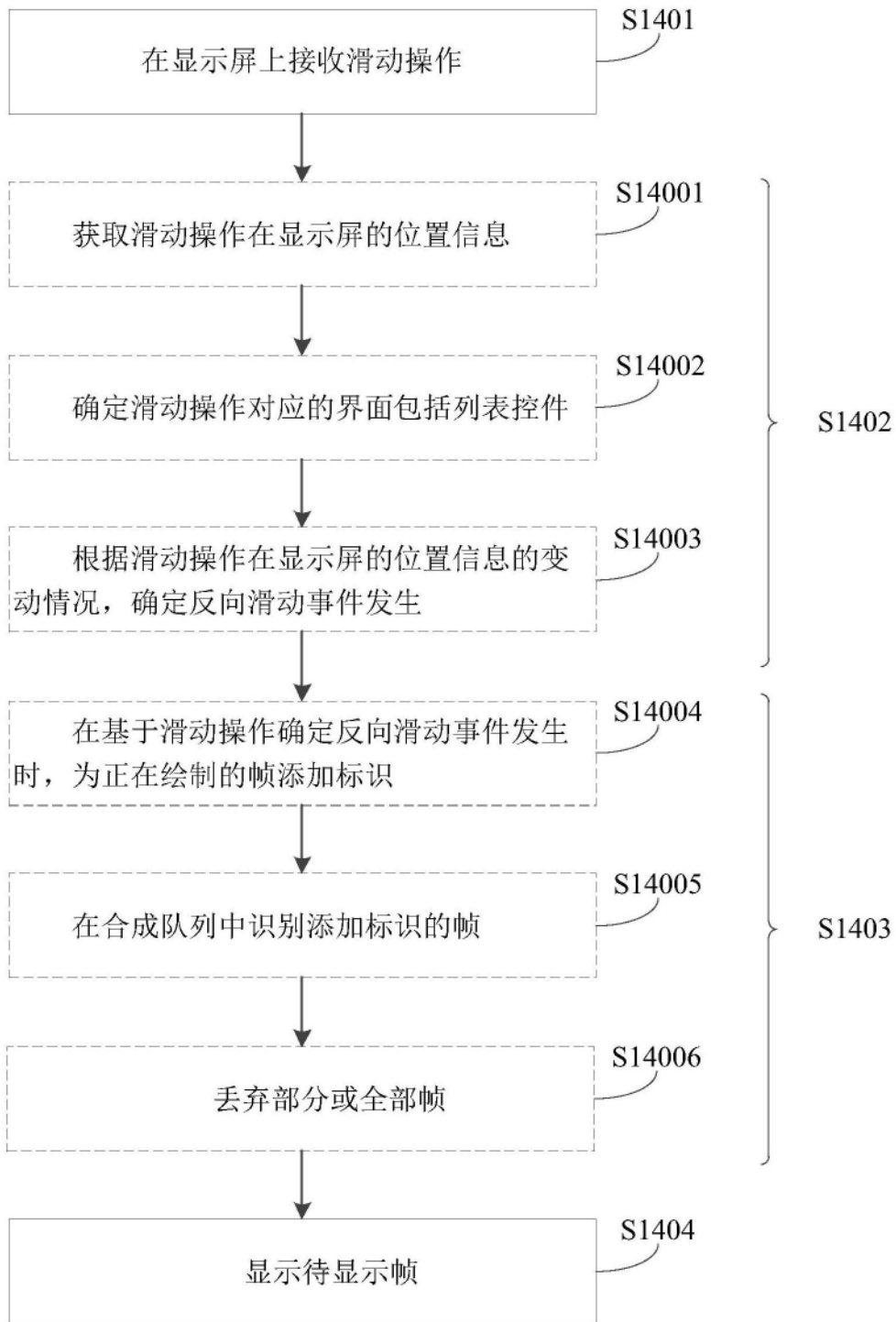


图14

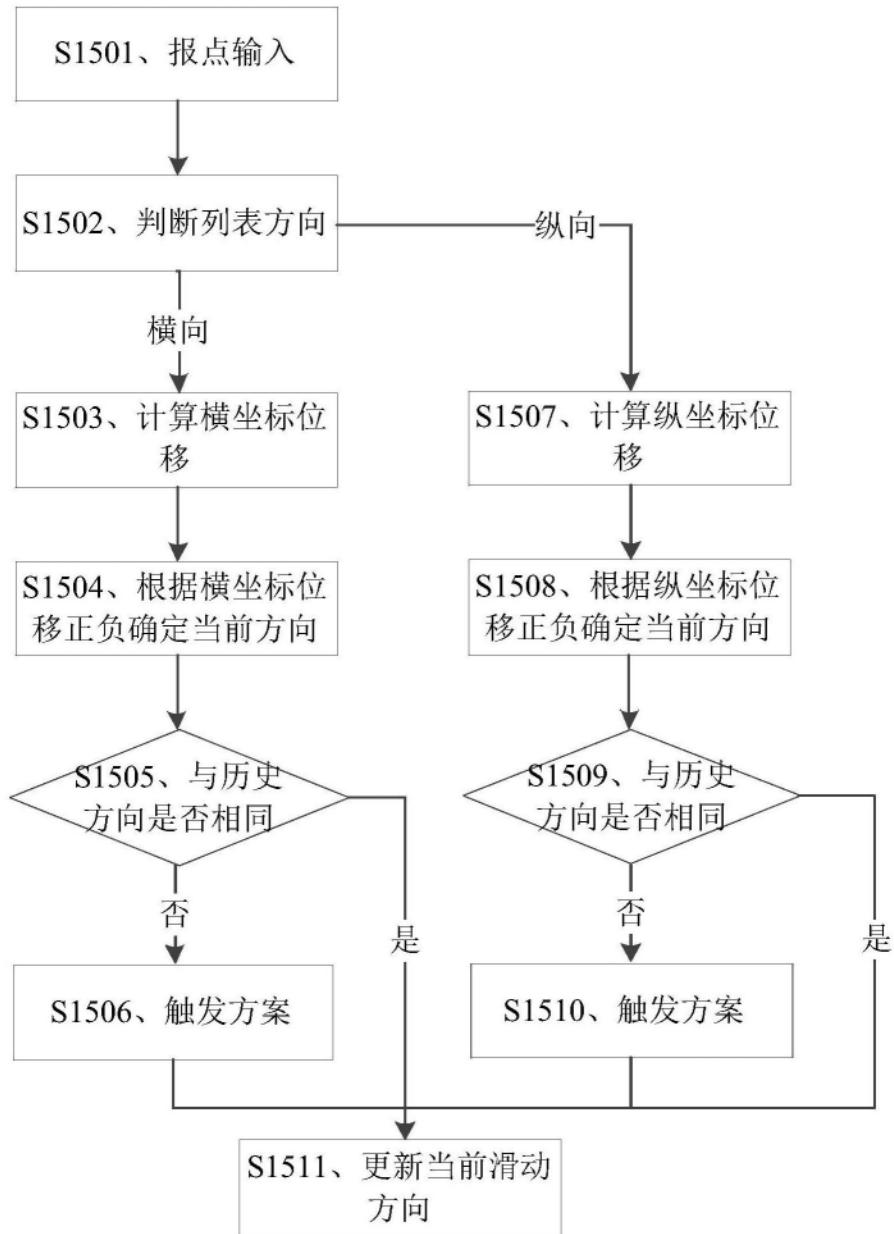


图15

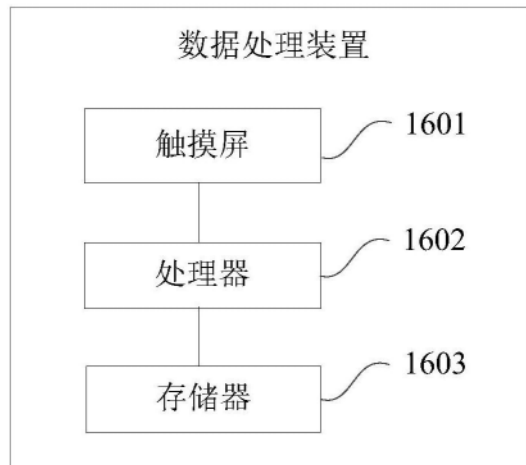


图16

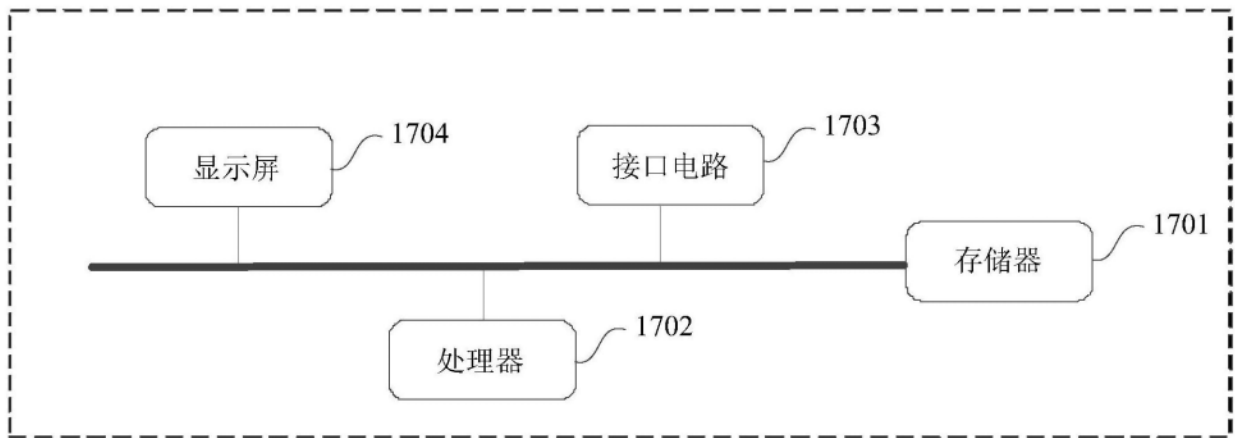


图17