



(12) 发明专利

(10) 授权公告号 CN 113053313 B

(45) 授权公告日 2022.08.02

(21) 申请号 202110327879.0

(22) 申请日 2021.03.26

(65) 同一申请的已公布的文献号  
申请公布号 CN 113053313 A

(43) 申请公布日 2021.06.29

(73) 专利权人 苏州科达科技股份有限公司  
地址 215011 江苏省苏州市高新区金山路  
131号

(72) 发明人 苗运齐 刘硕 贺晓巍

(74) 专利代理机构 上海隆天律师事务所 31282  
专利代理师 夏彬

(51) Int.Cl.  
G09G 3/3208 (2016.01)

(56) 对比文件

- CN 111798794 A, 2020.10.20
- CN 108766346 A, 2018.11.06
- CN 101650912 A, 2010.02.17
- CN 105869190 A, 2016.08.17
- CN 108958452 A, 2018.12.07
- CN 103325332 A, 2013.09.25
- CN 104423805 A, 2015.03.18
- US 2011096051 A1, 2011.04.28
- EP 2801969 A1, 2014.11.12
- CN 108735158 A, 2018.11.02

审查员 杜昕

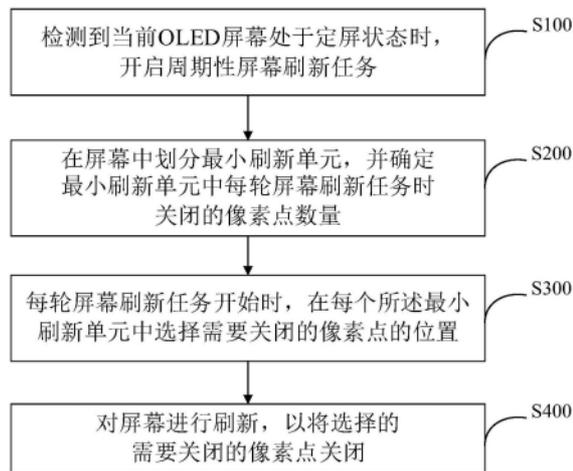
权利要求书3页 说明书17页 附图8页

(54) 发明名称

OLED屏幕残影减少方法、系统、设备及存储介质

(57) 摘要

本发明提供了一种OLED屏幕残影减少方法、系统、设备及存储介质,所述方法包括:检测到当前OLED屏幕处于定屏状态时,开启周期性屏幕刷新任务;在屏幕中划分最小刷新单元,并确定最小刷新单元中每轮屏幕刷新任务时关闭的像素点数量;每轮屏幕刷新任务开始时,在每个所述最小刷新单元中选择需要关闭的像素点的位置;对屏幕进行刷新,以将选择的需要关闭的像素点关闭。通过采用本发明,对应用本身没有设计要求,也不会强制退出当前的定屏图像,适用于各种场景中的OLED屏幕残影减少。



1. 一种OLED屏幕残影减少方法,其特征在于,包括如下步骤:

检测到当前OLED屏幕处于定屏状态时,开启周期性屏幕刷新任务,每个屏幕刷新周期包括多轮屏幕刷新任务;

在屏幕中划分最小刷新单元,并确定最小刷新单元中每轮屏幕刷新任务时关闭的像素点数量;

每轮屏幕刷新任务开始时,在每个所述最小刷新单元中选择需要关闭的像素点的位置;

对屏幕进行刷新,以将选择的需要关闭的像素点关闭;

所述在每个所述最小刷新单元中选择需要关闭的像素点的位置之后,还包括如下步骤:

对于本轮屏幕刷新任务,判断是否采用递进刷新方式;

如果是,则按照预设的拆分次数 $y$ ,将该轮屏幕刷新任务拆分为多次屏幕刷新子任务,递进关闭本轮被选中关闭的像素点,其中,对于第 $i$ 次刷新子任务,将上一轮或本轮被选中关系的像素点的位置所在的最小刷新单元的编号与所述拆分次数 $y$ 求余数,并根据余数与 $i-1$ 的比较结果控制该像素点在本次屏幕刷新子任务中是否关闭;

依次执行该轮屏幕刷新任务中的各次屏幕刷新子任务。

2. 根据权利要求1所述的OLED屏幕残影减少方法,其特征在于,所述在屏幕中划分最小刷新单元,并确定最小刷新单元中每轮刷新任务时关闭的像素点数量,包括如下步骤:

加载用户设定的屏幕刷新策略,所述屏幕刷新策略包括最小刷新单元的尺寸和最小刷新单元中每轮刷新任务时关闭的像素点比例和/或数量;

根据所述最小刷新单元的尺寸在屏幕中划分最小刷新单元;

根据所述屏幕刷新策略确定所述最小刷新单元中每轮刷新任务时关闭的像素点数量。

3. 根据权利要求2所述的OLED屏幕残影减少方法,其特征在于,所述加载用户设定的屏幕刷新策略之前,还包括如下步骤:

提供屏幕刷新策略配置页面给用户端;

在所述屏幕刷新策略配置页面中生成运行效果体验区,在所述运行效果体验区中采用包括多个像素点的像素块模拟最小刷新单元的一个像素点,并在所述运行效果体验区中模拟至少一个最小刷新单元;

接收用户设定的屏幕刷新策略后,根据用户设定的屏幕刷新策略,在所述运行效果体验区中进行刷新效果展示。

4. 根据权利要求1所述的OLED屏幕残影减少方法,其特征在于,所述开启周期性屏幕刷新任务之后,还包括如下步骤:

设置每轮屏幕刷新任务的间隔时间;

启动定时功能,根据所述间隔时间确定每轮屏幕刷新任务的时间,于每轮屏幕刷新任务的时间到达时,启动对应的屏幕刷新任务。

5. 根据权利要求1所述的OLED屏幕残影减少方法,其特征在于,在每个所述最小刷新单元中选择需要关闭的像素点的位置,包括如下步骤:

在每一轮屏幕刷新任务中,选择最小刷新单元中需要关闭的像素点的起始位置所对应的位置编号;

根据最小刷新单元中每轮屏幕刷新任务时关闭的像素点数量,选择最小刷新单元中除起始位置之外其他像素点的位置编号,包括:

如果最小刷新单元中每轮屏幕刷新任务时要关闭的像素点的数量 $a$ 占每个最小刷新单元中像素点总数量 $b$ 的比例小于等于 $1/2$ ,则在前一个位置的位置编号上加 $n$ ,作为后一个位置的位置编号;如果所述数量 $a$ 占所述总数量 $b$ 的比例大于 $1/2$ ,则在前一个位置的位置编号上加 $m$ ,作为后一个位置的位置编号, $m$ 、 $n$ 为预设系数, $m < n$ ;

如果得到的位置编号有超过所述总数量 $b$ 的,则将该位置编号与所述总数量 $b$ 取余,采用余数更新该位置编号。

6. 根据权利要求5所述的OLED屏幕残影减少方法,其特征在于,所述选择最小刷新单元中需要关闭的像素点的起始位置所对应的位置编号,包括如下步骤:

对于第一轮屏幕刷新任务,随机选择最小刷新单元中需要关闭的像素点的起始位置所对应的位置编号;

对于后续屏幕刷新任务,如果所述数量 $a$ 占所述总数量 $b$ 的比例小于等于 $1/2$ ,则在前一轮屏幕刷新任务中需要关闭的像素点的起始位置的位置编号上加 $q$ , $q$ 为预设系数,得到本轮屏幕刷新任务中需要关闭的像素点的起始位置的位置编号,如果所述数量 $a$ 占所述总数量 $b$ 的比例大于 $1/2$ ,则在前一轮屏幕刷新任务中需要关闭的像素点的起始位置的位置编号上加数量 $a$ ,得到本轮屏幕刷新任务中需要关闭的像素点的起始位置的位置编号;

如果得到的起始位置的位置编号大于所述总数量 $b$ ,则将得到的起始位置的位置编号与所述总数量 $b$ 取余,采用得到的余数更新本轮屏幕刷新任务中需要关闭的像素点的起始位置的位置编号。

7. 根据权利要求1所述的OLED屏幕残影减少方法,其特征在于,采用如下步骤确定在第 $i$ 次屏幕刷新子任务中关闭的像素点, $i \in (1, y)$ :

对于上一轮被选中关闭的像素点的位置,将其所在的最小刷新单元的编号与所述拆分次数 $y$ 求余数,如果余数大于 $i-1$ ,则保持该像素点在本次屏幕刷新子任务中继续关闭;

对于本轮被选中关闭的像素点的位置,将该像素点所在的最小刷新单元的编号与所述拆分次数 $y$ 求余数,如果余数小于等于 $i-1$ ,则将该像素点在本次屏幕刷新子任务中关闭。

8. 根据权利要求1所述的OLED屏幕残影减少方法,其特征在于,所述对屏幕进行刷新,以将选择的需要关闭的像素点关闭,包括如下步骤:

获取屏幕刷新数组,所述屏幕刷新数组中每个值对应于一个像素点;

将所述屏幕刷新数组中上一轮关闭的像素点对应的数值恢复为默认值,并将本轮需要关闭的像素点对应的数值设为预设像素关闭值;

基于所述屏幕刷新数组进行屏幕刷新,以将选择的需要关闭的像素点关闭。

9. 一种OLED屏幕残影减少系统,其特征在于,应用于权利要求1至8中任一项所述的OLED屏幕残影减少方法,所述系统包括:

状态监控模块,用于检测到当前OLED屏幕是否处于定屏状态;

任务调度模块,用于在屏幕处于定屏状态时,启动屏幕刷新模块,以开启周期性屏幕刷新任务,每个屏幕刷新周期包括多轮屏幕刷新任务;

屏幕刷新模块,用于在屏幕中划分最小刷新单元,并确定最小刷新单元中每轮屏幕刷新任务时关闭的像素点数量,每轮屏幕刷新任务开始时,在每个所述最小刷新单元中选择

需要关闭的像素点的位置,以及对屏幕进行刷新,以将选择的需要关闭的像素点关闭。

10. 一种OLED屏幕残影减少设备,其特征在于,包括:

处理器;

存储器,其中存储有所述处理器的可执行指令;

其中,所述处理器配置为经由执行所述可执行指令来执行权利要求1至8中任一项所述的OLED屏幕残影减少方法的步骤。

11. 一种计算机可读存储介质,用于存储程序,其特征在于,所述程序被处理器执行时实现权利要求1至8中任一项所述的OLED屏幕残影减少方法的步骤。

## OLED屏幕残影减少方法、系统、设备及存储介质

### 技术领域

[0001] 本发明涉及屏幕显示技术领域,尤其涉及一种OLED屏幕残影减少方法、系统、设备及存储介质。

### 背景技术

[0002] OLED (Organic Light-Emitting Diode,有机发光半导体) 屏幕是一种通过电流驱动有机薄膜本身来发光的显示技术,具有自发光的特性。

[0003] 当OLED屏幕在长期显示某个静止画面时,因整个图像会存在亮暗不均的情况,这时如果这些屏幕有机发光材料持续发光太久,不仅会加快屏幕衰老的速度,而且会在屏幕上出现该静态画面所残留的影像,该残留的影像一般俗称残影。残影的主要原因,就是定格图像的各个像素在屏幕上显示时,存在亮暗不均的差异,从而每个屏幕每个像素点的老化速度就有了差异,如果像素点老化差异的明显程度越高,那么屏幕的残影现象就越明显。

[0004] 目前的OLED电视机,如果用户长时间看某个电视台,这时该电视台徽标部分是长期不动的,如果这时切换到其它电视节目,那个徽标位置就会隐隐地看到先前一直在看的电视台徽标的形状。还有目前的Android智能手机,由于系统状态栏是长期不动的,主要用来显示电量、信号和时间等一些常规信息,在设备使用久了以后,这时让手机全屏显示一张纯色的图片,就可能在状态栏的位置隐约看到那些电量、信号等图标信息残留的痕迹。

[0005] 常见的残影有两种,一种是亮残影,另一种是暗残影。

[0006] 亮残影是指残留的影像比其它地方的影像显示要亮,亮残影在经过一段时间以后可以逐步恢复,但是会对显示效果产生影响,亮残影出现以后,如果重新启动设备,该残影现象可能会消失。暗残影指残留的影像比其它地方的影像显示暗,这种情况是发光材料的使用损耗衰减造成的,这种暗残影现象是彻底的物理伤害,出现暗残影后是不可恢复的,一般的残影现象都属于暗残影。要尽量做到使整个屏幕各像素点的显示差异不大,这样的话不仅在一定程度上解决屏幕残影的出现,而且可以有效延长屏幕的使用时间。

[0007] 现在的解决屏幕残影的技术,基本都是尽量避免显示静态画面过久,应用自身尽量避免出现亮暗分布不均衡的画面,在没有信息输入输出时,可以采用低亮度的动画屏保显示;可以使用深色模式来整体降低屏幕的亮度。如果设备出现了残影,可以让屏幕显示一张全白的图片,再将设备亮度调节到最亮,这样让设备显示长达数小时的时间,目的是让整个屏幕的各个像素老化程度达到基本同步,这样可以在一定程度上缓解残影所造成的使用影响。还有的方式是通过App自身降低屏幕图像的亮暗对比度。然而,如上的解决方式,要么因对App的设计要求提高了而不具备通用性,要么让屏幕通过屏保来退出当前的定屏图像而出现不满足客户使用场景的要求。

### 发明内容

[0008] 针对现有技术中的问题,本发明的目的在于提供一种OLED屏幕残影减少方法、系统、设备及存储介质,对应用本身没有设计要求,也不会强制退出当前的定屏图像,适用于

各种场景中的OLED屏幕残影减少。

[0009] 本发明实施例提供一种OLED屏幕残影减少方法,包括如下步骤:

[0010] 检测到当前OLED屏幕处于定屏状态时,开启周期性屏幕刷新任务,每个屏幕刷新周期包括多轮屏幕刷新任务;

[0011] 在屏幕中划分最小刷新单元,并确定最小刷新单元中每轮屏幕刷新任务时关闭的像素点数量;

[0012] 每轮屏幕刷新任务开始时,在每个所述最小刷新单元中选择需要关闭的像素点的位置;

[0013] 对屏幕进行刷新,以将选择的需要关闭的像素点关闭。

[0014] 本发明通过采用该OLED屏幕残影减少方法,在检测到屏幕处于定屏状态时,自动开启周期性屏幕刷新任务,以最小刷新单元为单位,在每个最小刷新单元中选择每轮需要关闭的像素点的位置,并据此对屏幕进行刷新,以关闭需要关闭的像素点,从而周期性地交替开启和关闭一定比例的像素点发光功能,使得整个屏幕的像素点老化程度可以达到基本同步,从而可以有效减少因为定屏而产生的屏幕烧屏残影现象。

[0015] 在一些实施例中,所述在屏幕中划分最小刷新单元,并确定最小刷新单元中每轮刷新任务时关闭的像素点数量,包括如下步骤:

[0016] 加载用户设定的屏幕刷新策略,所述屏幕刷新策略包括最小刷新单元的尺寸和最小刷新单元中每轮刷新任务时关闭的像素点比例和/或数量;

[0017] 根据所述最小刷新单元的尺寸在屏幕中划分最小刷新单元;

[0018] 根据所述屏幕刷新策略确定所述最小刷新单元中每轮刷新任务时关闭的像素点数量。

[0019] 在一些实施例中,所述加载用户设定的屏幕刷新策略之前,还包括如下步骤:

[0020] 提供屏幕刷新策略配置页面给用户端;

[0021] 在所述屏幕刷新策略配置页面中生成运行效果体验区,在所述运行效果体验区中采用包括多个像素点的像素块模拟最小刷新单元的一个像素点,并在运行效果体验区中模拟至少一个最小刷新单元;

[0022] 接收用户设定的屏幕刷新策略后,根据用户设定的屏幕刷新策略,在所述运行效果体验区中进行刷新效果展示。

[0023] 在一些实施例中,所述开启周期性屏幕刷新任务之后,还包括如下步骤:

[0024] 设置每轮屏幕刷新任务的间隔时间;

[0025] 启动定时功能,根据间隔时间确定每轮屏幕刷新任务的时间,于每轮屏幕刷新任务的时间到达时,启动对应的屏幕刷新任务。

[0026] 在一些实施例中,在每个所述最小刷新单元中选择需要关闭的像素点的位置,包括如下步骤:

[0027] 在每一轮屏幕刷新任务中,选择最小刷新单元中需要关闭的像素点的起始位置所对应的位置编号;

[0028] 根据最小刷新单元中每轮屏幕刷新任务时关闭的像素点数量,选择最小刷新单元中除起始位置之外其他像素点的位置编号,包括:

[0029] 如果最小刷新单元中每轮屏幕刷新任务时要关闭的像素点的数量 $a$ 占每个最小刷

新单元中像素点总数量 $b$ 的比例小于等于 $1/2$ ,则在前一个位置的位置编号上加 $n$ ,作为后一个位置的位置编号;如果所述数量 $a$ 占所述总数量 $b$ 的比例大于 $1/2$ ,则在前一个位置的位置编号上加 $m$ ,作为后一个位置的位置编号, $m$ 、 $n$ 为预设系数, $m < n$ ;

[0030] 如果得到的位置编号有超过数量 $b$ 的,则将该位置编号与数量 $b$ 取余,采用余数更新该位置编号。

[0031] 在一些实施例中,所述选择最小刷新单元中需要关闭的像素点的起始位置所对应的位置编号,包括如下步骤:

[0032] 对于第一轮屏幕刷新任务,随机选择最小刷新单元中需要关闭的像素点的起始位置所对应的位置编号;

[0033] 对于后续屏幕刷新任务,如果所述数量 $a$ 占所述总数量 $b$ 的比例小于等于 $1/2$ ,则在前一轮屏幕刷新任务中需要关闭的像素点的起始位置的位置编号上加 $q$ , $q$ 为预设系数,得到本轮屏幕刷新任务中需要关闭的像素点的起始位置的位置编号,如果所述数量 $a$ 占所述总数量 $b$ 的比例大于 $1/2$ ,则在前一轮屏幕刷新任务中需要关闭的像素点的起始位置的位置编号上加所述数量 $a$ ,得到本轮屏幕刷新任务中需要关闭的像素点的起始位置的位置编号;

[0034] 如果得到的起始位置的位置编号大于所述总数量 $b$ ,则将得到的起始位置的位置编号与所述总数量 $b$ 取余,采用得到的余数更新本轮屏幕刷新任务中需要关闭的像素点的起始位置的位置编号。

[0035] 在一些实施例中,所述在每个所述最小刷新单元中选择需要关闭的像素点的位置之后,还包括如下步骤:

[0036] 对于本轮屏幕刷新任务,判断是否采用递进刷新方式;

[0037] 如果是,则按照预设的拆分次数 $y$ ,将该轮屏幕刷新任务拆分为多次屏幕刷新子任务,递进关闭本轮被选中关闭的像素点;

[0038] 依次执行该轮屏幕刷新任务中的各次屏幕刷新子任务。

[0039] 在一些实施例中,

[0040] 采用如下步骤确定第 $i$ 次屏幕刷新子任务中关闭的像素点, $i \in (1, y)$ :

[0041] 对于上一轮被选中关闭的像素点的位置,将其所在的最小刷新单元的编号与所述拆分次数 $y$ 求余数,如果余数大于 $i-1$ ,则保持该像素点在本次屏幕刷新子任务中继续关闭;

[0042] 对于本轮被选中关闭的像素点的位置,将该像素点所在的最小刷新单元的编号与所述拆分次数 $y$ 求余数,如果余数小于等于 $i-1$ ,则将该像素点在本次屏幕刷新子任务中关闭。

[0043] 在一些实施例中,所述对屏幕进行刷新,以将选择的需要关闭的像素点关闭,包括如下步骤:

[0044] 获取屏幕刷新数组,所述屏幕刷新数组中每个值对应于一个像素点;

[0045] 将所述屏幕刷新数组中上一轮关闭的像素点对应的数值恢复为默认值,并将本轮需要关闭的像素点对应的数值设为预设像素关闭值;

[0046] 基于所述屏幕刷新数组进行屏幕刷新,以将选择的需要关闭的像素点关闭。

[0047] 本发明实施例还提供一种OLED屏幕残影减少系统,应用于所述的OLED屏幕残影减少方法,所述系统包括:

[0048] 状态监控模块,用于检测到当前OLED屏幕是否处于定屏状态;

[0049] 任务调度模块,用于在屏幕处于定屏状态时,启动屏幕刷新模块,以开启周期性屏幕刷新任务,每个屏幕刷新周期包括多轮屏幕刷新任务;

[0050] 屏幕刷新模块,用于在屏幕中划分最小刷新单元,并确定最小刷新单元中每轮屏幕刷新任务时关闭的像素点数量,每轮屏幕刷新任务开始时,在每个所述最小刷新单元中选择需要关闭的像素点的位置,以及对屏幕进行刷新,以将选择的需要关闭的像素点关闭。

[0051] 本发明通过采用该OLED屏幕残影减少系统,通过任务监控模块检测当前OLED屏幕是否处于定屏状态,并在检测到屏幕处于定屏状态时,通过任务调度模块自动开启周期性屏幕刷新任务,由屏幕刷新模块以最小刷新单元为单位,在每个最小刷新单元中选择每轮需要关闭的像素点的位置,并据此对屏幕进行刷新,以关闭需要关闭的像素点,从而周期性地交替开启和关闭一定比例的像素点发光功能,使得整个屏幕的像素点老化程度可以达到基本同步,从而可以有效减少因为定屏而产生的屏幕烧屏残影现象。

[0052] 本发明实施例还提供一种OLED屏幕残影减少设备,包括:

[0053] 处理器;

[0054] 存储器,其中存储有所述处理器的可执行指令;

[0055] 其中,所述处理器配置为经由执行所述可执行指令来执行所述的OLED屏幕残影减少方法的步骤。

[0056] 通过采用本发明所提供的OLED屏幕残影减少设备,所述处理器在执行所述可执行指令时执行所述的OLED屏幕残影减少方法,由此可以获得上述OLED屏幕残影减少方法的有益效果。

[0057] 本发明实施例还提供一种计算机可读存储介质,用于存储程序,所述程序被处理器执行时实现所述的OLED屏幕残影减少方法的步骤。

[0058] 通过采用本发明所提供的计算机可读存储介质,其中存储的程序在被执行时实现所述的OLED屏幕残影减少方法的步骤,由此可以获得上述OLED屏幕残影减少方法的有益效果。

## 附图说明

[0059] 通过阅读参照以下附图对非限制性实施例所作的详细描述,本发明的其它特征、目的和优点将会变得更明显。

[0060] 图1是本发明一实施例的OLED屏幕残影减少方法的流程图;

[0061] 图2是本发明一实施例的OLED屏幕残影减少系统的示意图;

[0062] 图3是本发明一具体实例的OLED屏幕残影减少系统的示意图;

[0063] 图4是本发明一具体实例的OLED屏幕残影减少方法的流程图;

[0064] 图5和图6是采用本发明的OLED屏幕残影减少方法的效果对比图,其中图5是原图,图6是采用该方法后的屏幕显示图;

[0065] 图7和图8是采用本发明的OLED屏幕残影减少方法的验证对比图,其中图7是原图,图8是采用该方法后的屏幕显示图;

[0066] 图9是本发明一实施例的OLED屏幕残影减少设备的结构示意图;

[0067] 图10是本发明一实施例的计算机存储介质的结构示意图。

## 具体实施方式

[0068] 现在将参考附图更全面地描述示例实施方式。然而，示例实施方式能够以多种形式实施，且不应被理解为限于在此阐述的实施方式；相反，提供这些实施方式使得本发明将全面和完整，并将示例实施方式的构思全面地传达给本领域的技术人员。在图中相同的附图标记表示相同或类似的结构，因而将省略对它们的重复描述。

[0069] 如图1所示，在一实施例中，本发明提供了一种OLED屏幕残影减少方法，包括如下步骤：

[0070] S100：检测到当前OLED屏幕处于定屏状态时，开启周期性屏幕刷新任务，每个屏幕刷新周期包括多轮屏幕刷新任务；

[0071] S200：在屏幕中划分最小刷新单元，并确定最小刷新单元中每轮屏幕刷新任务时关闭的像素点数量；

[0072] S300：每轮屏幕刷新任务开始时，在每个所述最小刷新单元中选择需要关闭的像素点的位置；

[0073] S400：对屏幕进行刷新，以将选择的需要关闭的像素点关闭。

[0074] 因OLED屏幕具有每个像素自发光特性，黑色部分的像素点是不发光的，那么在屏幕图像静止不动时，如果能够通过算法来周期性地关闭一定比例的像素点，从而被关闭的像素点就不会处于一直发光的状态，这样在设备定屏时，通过周期性交替地开启和关闭一定比例的像素点发光功能，那么整个屏幕的像素点老化程度可以达到基本同步，从而就可以有效解决屏幕的烧屏残影现象了。

[0075] 本发明在屏幕处于定屏状态并且持续一段时间后，利用OLED屏的每个像素自发光的特点，通过算法策略对屏幕图像进行周期性地刷新来关闭一定比例的像素点，使其在一定时间段内不再发光，在间隔一段时间以后，再关闭另一部分的像素点，这样经过几轮的循环，就可以让每部分的像素点都得到周期性的关闭，从而有效解决因不同像素点的使用差异而造成的屏幕烧屏残影现象了。

[0076] 因此，本发明通过采用该OLED屏幕残影减少方法，通过步骤S100在检测到屏幕处于定屏状态时，自动开启周期性屏幕刷新任务，通过步骤S200~S400以最小刷新单元为单位，在每个最小刷新单元中选择每轮需要关闭的像素点的位置，并据此对屏幕进行刷新，以关闭需要关闭的像素点，从而周期性地交替开启和关闭一定比例的像素点发光功能，使得整个屏幕的像素点老化程度可以达到基本同步，从而可以有效减少因为定屏而产生的屏幕烧屏残影现象。

[0077] 所述步骤S100中，对OLED屏幕进行定屏状态检测，具体地，监听当前处于前端显示的App是否有图像渲染，如果存在长时间没有图像渲染的定屏情况，就会开启周期性屏幕刷新任务，来关闭屏幕部分像素点。如果当前设备出现用户的某种操作行为，就会自动停止周期刷新功能，为了可以及时地停止周期刷新功能，这时需要从多个层面来监控具有该OLED屏幕的设备的运行状态。

[0078] 例如：

[0079] 1) 预设时间段内屏幕是否有亮灭状态改变；

[0080] 2) 预设时间段内是否有用户的触屏操作事件发生；

[0081] 3) 预设时间段内用户是否有按键操作发生；

[0082] 4) 预设时间段内屏幕是否有转屏事件发生;

[0083] 5) 预设时间段内屏幕图像是否有发生改变;

[0084] 只有在设备的多种状态都满足要求时,即在预设时间段内上述情况都没有发生,才能认定设备当前处于定屏状态,此时就可以开启周期刷新功能,否则就停止周期刷新功能。此处仅为判断定屏状态的一种方式,在其他可替代的实施方式中,也可以采用其他的条件来进行判断。

[0085] 在该实施例中,所述步骤S200:在屏幕中划分最小刷新单元,并确定最小刷新单元中每轮刷新任务时关闭的像素点数量,包括如下步骤:

[0086] 加载用户设定的屏幕刷新策略,所述屏幕刷新策略包括最小刷新单元的尺寸和最小刷新单元中每轮刷新任务时关闭的像素点比例和/或数量。

[0087] 此处最小刷新单元中每轮刷新任务时关闭的像素点比例为最小刷新单元中每轮刷新任务时关闭的像素点与该最小刷新单元中像素点总数的比例,也就等于每轮刷新任务中屏幕整体所要关闭的像素点数量占屏幕总像素点数量的比例。

[0088] 假设以一个长3个像素和宽2个像素的方块作为最小刷新单元,该最小刷新单元中有6个像素点,那么可以选取每次关闭30%比例的像素点,即关闭最下刷新单元中的2个像素点。假设每个最小刷新单元中像素点的位置序号从0到5进行编号,那么假设第一轮刷新关闭0和1这两个编号的像素点,第二轮刷新关闭2和3,第三轮4和5,这样每经过三轮为一个周期的循环,就可以将该最小刷新单元中的所有像素点都关闭一次,从而每经过三轮为一个周期的循环刷新就可以将屏幕中的所有像素点都刷新关闭一次。

[0089] 在每一轮的屏幕刷新任务中,这个比例的数值既不能太大也不能太小。如果比例的数值比较大,那么在屏幕刷新时,用户会明显感觉到屏幕图像有闪动的效果(俗称“闪屏”),从而屏幕就会出现周期性的闪屏现象,这一点对用户的体验造成很不好的影响。如果这个比例的值比较小,那么将所有像素点都刷新一遍所需要的时间就会太久,这样在每轮的刷新周期中,就会出现屏幕上被最早关闭的像素点需要等待很长的时间才能轮到下一轮的关闭刷新,这样就不能很好地解决因像素点的使用不均而造成的屏幕老化残影的问题了。综上,该比例的取值的大小需要合理选择,并且可以通过实际测试验证来选择能获得比较优异效果的比值,来尽可能减少屏幕残影问题。

[0090] 根据所述最小刷新单元的尺寸在屏幕中划分最小刷新单元;

[0091] 根据所述屏幕刷新策略确定所述最小刷新单元中每轮刷新任务时关闭的像素点数量。

[0092] 具体地,如果所述屏幕刷新策略包括所述最小刷新单元中每轮刷新任务时关闭的像素点数量,则基于所述屏幕刷新策略定义刷新过程中,所述最小刷新单元中每轮刷新任务时关闭的像素点数量。如果所述屏幕刷新策略包括所述最小刷新单元中每轮刷新任务时关闭的像素点比例,则根据每个最小刷新单元中的像素点总数乘以该像素点比例,得到刷新过程中,所述最小刷新单元中每轮刷新任务时关闭的像素点数量。

[0093] 因屏幕刷新功能需要使用到具体的刷新策略,例如:每轮刷新需要关闭多大比例的像素点,那么就需提供一个可供用户调节屏幕刷新策略的屏幕刷新策略配置页面,可以让用户根据自己的实际需要进行相应的配置。

[0094] 具体地,所述加载用户设定的屏幕刷新策略之前,还包括如下步骤:

[0095] 提供屏幕刷新策略配置页面给用户端,在所述屏幕刷新策略配置页面中接收用户配置的屏幕刷新策略。

[0096] 所述步骤S200中,在屏幕中划分最小刷新单元,即为根据最小刷新单元的尺寸,通过最小刷新单元将整个屏幕分割成若干块,假设屏幕的分辨率为1920\*1080,并且最小刷新单元是2\*3,即长为3个像素、宽为2个像素,那么可以通过该最小刷新单元会将屏幕分割成 $(1920/2) * (1080/3) = 960 * 360 = 345600$ 块,且每一块的大小都是2\*3,那么整个屏幕就可以不用以像素点为单位来进行处理,而是按照最小刷新单元来进行处理了,并且所有的最小刷新单元可以先按行然后再按列进行排序,假设排序的编号依次从0到345599,那么接下来的刷新重点就是对每一个刷新单元的刷新策略进行计算,获取每一个最小刷新单元中所要关闭的像素点位置,然后将要关闭的所有像素点位置存储在一个数组里面,最后对屏幕进行刷新来关闭数组中对应位置的像素点即可。

[0097] 在屏幕刷新策略配置页面还可以提供一个针对当前刷新策略的运行效果体验区,可以让用户实时查看当前屏幕刷新策略运行后的屏幕效果。为了让用户可以更清楚地理解最小刷新单元概念及每个刷新单元下所关闭的像素点的策略运行机制,会在该区域的每个最小刷新单元中用一个像素块来模拟一个像素点,即将若干个像素点模拟成一个像素点,这样用户看起来就更为直观。

[0098] 具体地,在屏幕刷新策略配置页面中生成运行效果体验区,在所述运行效果体验区中采用包括多个像素点的像素块模拟最小刷新单元的一个像素点,并在运行效果体验区中模拟至少一个最小刷新单元;接收用户设定的屏幕刷新策略后,根据用户设定的屏幕刷新策略,在所述运行效果体验区中进行刷新效果展示。

[0099] 例如,将长480像素和宽1080像素的一个长方形效果体验区分割成12个长120像素和宽270像素的长方形方块,将这样的每一个120\*270的长方形方块看作一个最小刷新单元,假设该最小刷新单元为2\*3,那么该最小刷新单元中每一个像素点实际代表 $(120/2) * (270/3) = 60 * 90 = 5400$ 个像素点,从而在该运行效果体验区,用户会更直观地感受到当前刷新策略下的屏幕刷新策略的运作机制。

[0100] 要想屏幕的所有像素点都被刷新关闭一次,那么需要经过多轮刷新才可以,至于要经过多少轮的刷新才能将屏幕的所有像素点都完整地刷新一次,这个时间取决于每轮刷新的时间间隔。具体地,在该实施例,所述步骤S100中开启周期性屏幕刷新任务之后,还包括如下步骤:

[0101] 设置每轮屏幕刷新任务的间隔时间;

[0102] 启动定时功能,于每轮屏幕刷新任务的时间到达时,启动对应的屏幕刷新任务。

[0103] 在开启屏幕刷新功能的时候,会开启一个用于定时刷新的TimerTask,该TimerTask会间隔一个固定时间来启动下一轮的刷新任务,该固定时间就是每一轮刷新的间隔时间,这样就可以实现屏幕定时刷新的功能了,但是在设备状态发生改变(例如:设备屏灭)的时候,要关闭该定时任务。

[0104] 在用户进行用户刷新策略配置时,还可以确认每轮屏幕刷新任务的时间间隔。如果每轮屏幕刷新任务的时间间隔所取的值比较大,这样要将屏幕所有像素点都刷新关闭一次的时间周期就会比较长,如若这样,那么在每一个刷新周期中,在第一轮刷新任务中被关闭的像素点,就需要等待较长的时间,才能在下一个屏幕刷新周期被关闭,这样就会同样出

现因屏幕像素点使用不均而出现屏幕老化残影的问题了。如果每轮屏幕刷新任务的时间间隔所取的值比较小,这样就会造成屏幕刷新过于频繁,因每次刷新都需要申请CPU和内存资源,而且这些额外的资源开销本不属于App业务范畴,如果刷新过于频繁,那么因功耗过高对设备的续航能力会造成过大的负面影响。综上,每轮屏幕刷新任务的时间间隔同样要取一个比较合适的值,并且可以通过实际的测试验证过程来选择一个具有优异效果的时间间隔值,来尽可能减少屏幕残影问题。既要能避免因屏幕像素点的使用不均造成的屏幕老化残影问题,又要不能让屏幕刷新过于频繁而造成额外功耗过高而影响设备的续航能力。

[0105] 在该实施例中,所述步骤S300:在每个所述最小刷新单元中选择需要关闭的像素点的位置,包括如下步骤:

[0106] 在每一轮屏幕刷新任务中,选择最小刷新单元中需要关闭的像素点的起始位置所对应的位置编号;

[0107] 根据最小刷新单元中每轮屏幕刷新任务时关闭的像素点数量,选择最小刷新单元中除起始位置之外其他像素点的位置编号,包括:

[0108] 如果最小刷新单元中每轮屏幕刷新任务时要关闭的像素点的数量 $a$ 占每个最小刷新单元中像素点总数量 $b$ 的比例小于等于 $1/2$ ,则在前一个位置的位置编号上加 $n$ ,作为后一个位置的位置编号;如果所述数量 $a$ 占所述总数量 $b$ 的比例大于 $1/2$ ,则在前一个位置的位置编号上加 $m$ ,作为后一个位置的位置编号, $m$ 、 $n$ 为预设系数, $m < n$ ,该实施例中 $n$ 选择为2, $m$ 选择为1;

[0109] 如果得到的位置编号有超过所述总数量 $b$ 的,则将该位置编号与所述总数量 $b$ 取余,采用余数更新该位置编号。

[0110] 具体地,在已经获取到了每个最小刷新单元中要关闭的像素点起始位置之后,在每个最小刷新单元中,所要关闭的像素点的后续位置,要通过该起始位置来获取。此时,如果每个最小刷新单元中要关闭的像素点数量占每个最小刷新单元中像素点数量的比例不超过一半,那么要关闭的像素点的后一个位置在上一个位置的基础上加2;如果每个最小刷新单元要关闭的像素点数量占每个最小刷新单元中像素点数量的比例超过一半,那么要关闭的像素点的后一个位置在上一个位置的基础上加1。例如,最小刷新单元为 $2 \times 3$ ,其中的6个像素点的编号为0到5,当前第一个最小刷新单元要关闭的像素点的起始位置编号为2;此时如果需要关闭的像素点数量为3(不大于一半),那么后面两个要关闭的像素点位置编号依次在上一个的基础上加2再取余,那么对应的值为4和0,从而该轮刷新中第一个最小刷新单元要关闭的像素点位置编号为2、4、0;如果需要关闭的像素点数量为4(大于一半),那么后续两个要关闭的像素点位置编号依次在上一个的基础上加1再取余,那么对应的值为3、4、5,从而要关闭的像素点位置编号为2、3、4、5。

[0111] 每个最小刷新单元中都需要关闭一定数量的像素点,而且每个刷新单元所要关闭的像素点数量是相同的,每轮刷新中要关闭的像素点的位置编号,它们相对于各自所属的最小刷新单元的位置编号可以相同也可以不同。假设最小刷新单元为 $2 \times 3$ ,并且该最小刷新单元中每个像素点位置的编号从0到5,此外每次刷新需要关闭的像素点数量为2,那么此时第一个最小刷新单元的位置编号可以是0和1,第二个最小刷新单元的位置编号可以是0和1或者2和3。如果每个最小刷新单元中所要关闭的像素点位置编号一样,那么这些被关闭的像素点位置基本上是成一行或一列的样式,那么通过屏幕刷新将这些像素点关闭后,屏幕

上会出现黑色的横向条纹或竖向条纹,这些条纹对用户的体验不是很好,此时可以将每个最小刷新单元中要关闭的像素点位置进行随机选取,由于位置是随机的,那么所有最小刷新单元中要被关闭的像素点位置就不会存在一定规律,即不会成一行或一列的样式,从而当屏幕上的这些位置随机的像素点被关闭后,就不会出现条纹的样式,而此时的屏幕会呈现一种磨砂的效果。

[0112] 具体地,在该实施例中,所述选择最小刷新单元中需要关闭的像素点的起始位置所对应的位置编号,包括如下步骤:

[0113] 对于第一轮屏幕刷新任务,随机选择最小刷新单元中需要关闭的像素点的起始位置所对应的位置编号;

[0114] 每个最小刷新单元中要关闭的像素点的初始位置,可以取不大于每个最小刷新单元中像素点数量的一个随机数,例如,最小刷新单元中的像素点数量为6,那么该随机数可以是0到5中的任何一个随机值,此时每个最小刷新单元中要关闭的像素点的初始位置都是从0到5中的任何一个随机值,然后每个最小刷新单元中其它要关闭的像素点位置都依据各自的起始随机值来通过计算获取,从而也可以确保要关闭的所有像素点的位置编号都基本随机。

[0115] 对于后续屏幕刷新任务,如果所述数量a占所述总数量b的比例小于等于 $1/2$ ,则在上一轮屏幕刷新任务中需要关闭的像素点的起始位置的位置编号上加q,q为预设系数,该实施例中q为1,得到本轮屏幕刷新任务中需要关闭的像素点的起始位置的位置编号,如果所述数量a占所述总数量b的比例大于 $1/2$ ,则在上一轮屏幕刷新任务中需要关闭的像素点的起始位置的位置编号上加所述数量a,得到本轮屏幕刷新任务中需要关闭的像素点的起始位置的位置编号;

[0116] 如果得到的起始位置的位置编号大于所述总数量b,则将得到的起始位置的位置编号与所述总数量b取余,采用得到的余数更新本轮屏幕刷新任务中需要关闭的像素点的起始位置的位置编号。

[0117] 即第二轮及以后的每一轮刷新中获取每个最小刷新单元所要关闭的像素点起始位置采用如下方式选取:

[0118] 针对某个具体刷新策略下的第一轮刷新,每个最小刷新单元已经通过随机策略确定了要关闭的像素点初始位置,那么在后续每一轮刷新任务中,每个最小刷新单元的起始位置怎么取值呢?可以采用如下的取值策略,如果每个最小刷新单元中,要关闭的像素点数量占每个最小刷新单元中像素点数量的比例不超过一半,那么在后续每一轮刷新任务中,每个最小刷新单元中要关闭的像素点起始位置在上一轮的基础上加1;如果每个最小刷新单元中要关闭的像素点数量占每个最小刷新单元中像素点数量的比例超过一半,那么后续每一轮刷新任务中,每个最小刷新单元中要关闭的像素点起始位置在上一轮的基础上,加每个最小刷新单元中所要关闭的像素点数量所对应的数值;如上两种加法以后得出的值,如果该值超过最小刷新单元的像素点数量,那么再跟每个最小刷新单元的像素点数量值进行取余运算,最后的余数就是当前这一轮刷新任务中该最小刷新单元所要关闭的像素点起始位置。例如每个最小刷新单元包含的像素点数量为6,每一轮刷新所要关闭的像素点数量为4,且上一轮刷新时,该最小刷新单元中所关闭的像素点的起始位置为3,那么当前这一轮刷新中该最小刷新单元所要关闭的像素点的起始位置为 $(3+4) \% 6 = 1$ ,以此类推,就可以取

出每一刷新轮中每个最小刷新单元所要关闭的像素点的起始位置了。

[0119] 在该实施例中,所述步骤S300:在每个所述最小刷新单元中选择需要关闭的像素点的位置之后,还包括如下步骤:

[0120] 对于本轮屏幕刷新任务,判断是否采用递进刷新方式;

[0121] 如果是,则按照预设的拆分次数 $y$ ,将该轮屏幕刷新任务拆分为多次屏幕刷新子任务,递进关闭本轮被选中关闭的像素点;

[0122] 依次执行该轮屏幕刷新任务中的各次屏幕刷新子任务。

[0123] 采用如下步骤确定第 $i$ 次屏幕刷新子任务中关闭的像素点, $i \in (1, y)$ :

[0124] 对于上一轮被选中关闭的像素点的位置,将其所在的最小刷新单元的编号与所述拆分次数 $y$ 求余数,如果余数大于 $i-1$ ,则保持该像素点在本次屏幕刷新子任务中继续关闭;

[0125] 对于本轮被选中关闭的像素点的位置,将该像素点所在的最小刷新单元的编号与所述拆分次数 $y$ 求余数,如果余数小于等于 $i-1$ ,则将该像素点在本次屏幕刷新子任务中关闭。

[0126] 例如,每一轮的刷新比例为30%,共拆分成7次子任务来递进刷新,那么第一次刷新时,针对上一轮被选中关闭的像素点位置,如果该位置所在的最小刷新单元的编号跟7求余后值大于0,那么该像素点位置继续关闭;针对这一轮被选中关闭的像素点位置,如果该位置所在的最小刷新单元的编号跟7求余后的值小于等于0,那么该像素点的位置会关闭,从而第一次刷新关闭的像素点中,这一轮被选中关闭的像素点中,只有所在的最小刷新单元的编号是7的倍数的会被关闭,即有6/7是上一轮所关闭的像素点,有1/7是这一轮要关闭的像素点;同理,在第二次刷新时,针对上一轮被选中关闭的像素点位置,如果该位置所在的最小刷新单元的编号跟7求余后值大于1,那么该像素点位置继续关闭,针对这一轮被选中关闭的像素点位置,如果该位置所在的最小刷新单元的编号跟7求余后的值小于等于1,那么该像素点的位置会关闭,这一轮被选中关闭的像素点中,所在的最小刷新单元的编号是7的倍数和与7取余数为1的会被关闭,即有5/7是上一轮所关闭的像素点,有2/7是这一轮要关闭的像素点;...;第六次刷新时,针对上一轮被选中关闭的像素点位置,如果该位置所在的最小刷新单元的编号跟7求余后值大于5,那么该像素点位置继续关闭,针对这一轮被选中关闭的像素点位置,如果该位置所在的最小刷新单元的编号跟7求余后的值小于等于5,那么该像素点的位置会关闭,这一轮被选中关闭的像素点中,最小刷新单元的编号是7的倍数和与7取余数为1、2、3、4的会被关闭,有1/7是上一轮所关闭的像素点,有6/7是这一轮要关闭的像素点;第七次刷新时,针对上一轮被选中关闭的像素点位置,由于所有编号跟7求余后都小于等于6,因此没有上一轮被选中关闭的像素点保持关闭,针对这一轮被选中关闭的像素点位置,如果该位置所在的最小刷新单元的编号跟7求余后的值小于等于6,那么该像素点的位置会关闭,也就是这一轮被选中关闭的像素点在第7次子任务均需要关闭,第7次子任务时,全部都是这一轮要关闭的像素点。通过如上所述的步骤,就可以获取到每次递进刷新时需要关闭的像素点位置了。

[0127] 在用户进行屏幕刷新策略的配置时,还可以进行如下配置:

[0128] (1) 确认需要将每轮刷新任务分为多少次子任务来执行:

[0129] 针对每一轮的屏幕刷新任务,在具体执行该刷新任务时,为了避免因一次刷新关闭太多像素点而造成屏幕闪屏的现象,需要将每一轮的刷新任务再次进行拆分为多次子任

务来执行,每个子任务所要关闭的像素点数量仅占该轮任务所要关闭的像素点数量的某一个比例。例如:在每一轮刷新任务中都将关闭屏幕30%的像素点,如果这个30%的比例在一次屏幕刷新就都关闭的话,那么此时用户会明显感觉到屏幕出现闪屏的现象,针对这种情况,可以选择将30%的比例再次分10次小的任务来刷新关闭,那么每次刷新只要关闭屏幕3%的像素点,从而10次小的任务中,每次屏幕刷新所关闭的像素点数量,跟上一轮刷新任务所关闭的像素点数量相比,都递进增加3%,即:第一次刷新任务跟上一轮刷新任务有3%不同,第二次跟上一轮有6%不同,第三次跟上一轮有9%不同,……,第10次跟上一轮有30%不同,这样通过累加递进的方式通过10次小的刷新任务,就可以完成30%的屏幕像素点刷新任务了,通过这种方式不但可以确保每一轮刷新,所关闭的像素点数量比例不会太少,并且也不会出现屏幕闪屏的现象。

[0130] (2) 确认每轮刷新任务中每次递进刷新的时间间隔

[0131] 物体的图像是成像在视网膜上的,人眼在观看物体图像时,视网膜上的图像由视神经输入人脑,然后人才能感觉到物体的像,但当物体移动时,视神经对物体的影像是不会立即消失的,而要延续0.1~0.4秒的时间,如果每次屏幕刷新时的间隔时间过短,那么此时的视觉记忆还停留在以前的图像上,而当前屏幕上的图像与人记忆中的图像就是出现差别太大的情况,从而就能明显感觉到图线的变化,也就间接出现屏幕闪屏的现象了,如果每一次屏幕刷新间隔是大概0.5秒的时间,就可以将图像微小变动的的时间间隔在与视觉暂留的时间保持基本一致,这样就可以让屏幕图像微小变化与人视觉上的变化同步起来,从而就不容易察觉到图像的明显变化了,也就不会让感觉到屏幕出现闪屏的现象了。

[0132] 目前测试验证下来,在每一轮刷新比例为30%的情况下,将每一轮的刷新任务拆分为7次递进刷新的子任务来实现,不会出现明显的闪屏现象;那么针对其它刷新比例的情况,可以采用如下两种方式来处理,一是所有的刷新比例都按照7次来进行拆解;二是以30%的比例所对应的7次拆解作为一个标准,其它比例的刷新任务,按照这个标准来进行拆解,如果刷新比例上调,那么所拆解的次数同比例增加,如果刷新比例下调,那么所拆解的次数也同比例减少,采取这样一种动态调整的方式也是可行的。

[0133] 在递进刷新过程中,还需要对递进刷新状态进行监控。由于在每次递进刷新时,两次刷新之间需间隔500ms,因人眼对图像视觉暂留的影响,每次递进刷新的间隔不能太快,否则也会出现闪屏的现象,但在这间隔的500ms时间内,设备的运行状态可能会发生改变,在状态改变时应立即停止后续的刷新操作并退出当前的屏幕刷新任务。如果该500ms的时间,使用的是线程sleep操作,此时该线程一旦进入sleep状态,那么该线程在该500ms内是不能跟外界进行通信的,此时即使设备状态发生了改变,也无法通知该线程停止当前的刷新行为,从而sleep方式不可取,但还有一种方式是线程wait/notify,该实施例可以在线程wait之前,通过handler delay500ms的方式发送一个message,然后再调用线程wait方法来让当前线程的运行暂停,在等待500ms后,收到notify通知后继续工作,如果在这500ms内出现设备状态改变需要停止刷新任务的情况,就可以提前发送message来唤醒被wait的线程继续工作来退出当前的刷新任务。

[0134] 通过之前的步骤,每次任务刷新所要关闭的像素点位置已经知道了,那么每次刷新时只要处理相应步骤来要刷新屏幕关闭像素点即可。具体地,所述步骤S400:对屏幕进行刷新,以将选择的需要关闭的像素点关闭,包括如下步骤:

[0135] 获取屏幕刷新数组,所述屏幕刷新数组中每个值对应于一个像素点;

[0136] 将所述屏幕刷新数组中上一轮关闭的像素点对应的数值恢复为默认值,并将本轮需要关闭的像素点对应的数值设为预设像素关闭值;

[0137] 基于所述屏幕刷新数组进行屏幕刷新,以将选择的需要关闭的像素点关闭。

[0138] 在屏幕刷新开始之前,先通过设备分辨率来初始化一个数组,例如,设备分辨率为 $1920*1080$ ,那么所初始化的数组大小为 $1920*1080$ ,并且将该数组的值初始化为 $0x00ffffff$ ,代表该位置的像素点是透明的,即每一个像素点默认不关闭,那么第一次刷新时,首先取出第一次刷新时要关闭的像素点位置,然后将如上数组中该位置所对应的数值设为 $0xff000000$ ,即关闭该位置的像素点,并且通过该数组创建一个bitmap位图,接下来将此bitmap位图赋值给imageview,然后windowmanager通过该imageview来刷新对应view,最后屏幕刷新时就会关闭如上数组定义中所对应的像素点了。在从第二次开始的每次刷新任务中,为避免内存损耗太大,需要重复使用如上已初始化的数组,在取出要关闭的像素点位置之前,要将数组中上一次刷新时已关闭的像素点位置的值修改为 $0x00ffffff$ ,即恢复成默认透明的状态,然后再取出此次所要关闭的像素点位置,接下来通过该位置的数值来修改数组,将该位置数组值设为 $0xff000000$ ,接下来的刷新步骤就跟第一次相同了。

[0139] 本发明的OLED屏幕残影减少方法还可以进一步包括功能验证的过程。由于屏幕刷新策略最终操作的是屏幕像素点,而当前设备的分辨率一般都比较,例如, $1080P$ 分辨率的设备,它的像素数量为 $1920*1080=2073600$ ,从而最终的刷新效果如果只是通过人为观察来判断,那么结果的可靠性是很难保证的,所以我们需要一个功能验证模块,它可以根据当前用户所选择的刷新策略来判断结果的正确性。该功能验证模块主要包括如下几个验证的点。

[0140] 1) 每轮刷新时所关闭的像素点比例是否满足刷新策略所设定的比例

[0141] 通过屏幕分辨率来计算所有像素点的数量,然后遍历每轮刷新中的每次递进刷新任务中的像素点数组,找出其中需要关闭的像素点的总数,最后通过要关闭的像素点总数和屏幕所有像素点总数相除,得出本次递进刷新中所要关闭的像素点占比是否符合策略预期。

[0142] 2) 一个刷新周期是否完整地覆盖了屏幕中的所有像素点

[0143] 通过每轮需要关闭的最小刷新单元中的像素点数量,可以计算出一个刷新周期共需要几轮刷新任务,将每一轮刷新任务中的像素点位置保存起来,假设每个周期需要三轮刷新,那么经过任意三轮后,如果已刷新关闭的像素点位置数量与屏幕总像素点数量相等,说明一个周期是可以覆盖屏幕中的所有像素点的。

[0144] 3) 每次递进刷新时是否是按照比例以累进增加的方式来刷新的

[0145] 在每次递进刷新时,找出该次与上一次所关闭的相同位置的像素点数量,如果这个数量的占比每次都是按照一个固定值在增加,说明是以递进累加的方式来刷新的。例如,将每轮刷新任务拆分为7次递进刷新,那么每次递进刷新的增加比例为 $1/7$ ,如果实际验证的增加比例正好满足 $1/7$ ,说明每次递进刷新时是按照比例累进增加的方式来刷新的。

[0146] 如图2所示,本发明实施例还提供一种OLED屏幕残影减少系统,应用于所述的OLED屏幕残影减少方法,所述系统包括:

[0147] 状态监控模块M100,用于检测到当前OLED屏幕是否处于定屏状态。

[0148] 具体地,所述状态监控模块M100可以实时地对灭屏状态、触屏状态、按键状态、转屏状态和图像状态进行综合监控,当监控到设备图像处于定屏状态时,就开始屏幕周期刷新功能,如果在执行屏幕周期刷新期间出现设备状态改变,此时会及时地退出周期性屏幕刷新功能;

[0149] 任务调度模块M200,用于在屏幕处于定屏状态时,启动屏幕刷新模块,以开启周期性屏幕刷新任务,每个屏幕刷新周期包括多轮屏幕刷新任务。

[0150] 具体地,当状态监控模块监控到设备处于定屏状态时,此时会通知任务调度模块M200来进行周期性屏幕刷新任务,任务调度模块主要是在状态监控模块M100和屏幕刷新模块M300之间做设备状态变化确认、过滤和排序的功能,因设备的状态变化可能会比较频繁,此时在开启或关闭屏幕刷新模块之前,会通过任务调度模块来进行状态的过滤和检查;

[0151] 屏幕刷新模块M300,用于在屏幕中划分最小刷新单元,并确定最小刷新单元中每轮屏幕刷新任务时关闭的像素点数量,每轮屏幕刷新任务开始时,在每个所述最小刷新单元中选择需要关闭的像素点的位置,以及对屏幕进行刷新,以将选择的需要关闭的像素点关闭。

[0152] 具体地,在开始屏幕刷新时,所述屏幕刷新模块M300会首先加载屏幕刷新策略,接下来获取每个最小刷新单元中要关闭的像素点的起始位置,然后通过该起始位置计算每个最小刷新单元中要关闭的其它像素点位置,最后通过递进累加的方法来进行屏幕的刷新工作,此外也会为下一轮的屏幕刷新任务开启一个定时器,该定时器会在下一轮任务刷新时间到达的时候开始工作,这样就可以实现基本的周期性屏幕刷新流程了。

[0153] 本发明通过采用该OLED屏幕残影减少系统,通过任务监控模块M100检测当前OLED屏幕是否处于定屏状态,并在检测到屏幕处于定屏状态时,通过任务调度模块M200自动开启周期性屏幕刷新任务,由屏幕刷新模块M300以最小刷新单元为单位,在每个最小刷新单元中选择每轮需要关闭的像素点的位置,并据此对屏幕进行刷新,以关闭需要关闭的像素点,从而周期性地交替开启和关闭一定比例的像素点发光功能,使得整个屏幕的像素点老化程度可以达到基本同步,从而可以有效减少因为定屏而产生的屏幕烧屏残影现象。

[0154] 如图3所示,为本发明一具体实例的OLED屏幕残影减少系统的结构示意图。在该实施例中,所述系统还进一步包括效果展示模块、功能验证模块和策略配置模块。其中,所述策略配置模块用于让用户可以定制自己的屏幕刷新策略,例如,选择最小刷新单元是2\*3还是3\*3的规格,选择每个最小刷新单元中需要关闭的像素点比例等。所述效果展示模块用于生成运行效果体验区,通过像素点模拟,可以让用户比较直观地感受到周期性屏幕刷新功能的运行原理和效果。

[0155] 所述功能验证模块主要用于功能验证,主要验证的内容包括:

[0156] (1) 每轮刷新时,所关闭的像素点数量比例与刷新策略所设定的比例是否相同。

[0157] (2) 每轮刷新当中的递进刷新方法,是否按照本轮刷新所要关闭的像素点数量以逐步累加的方式来进行刷新的。

[0158] (3) 每个刷新周期内是否把屏幕上的所有像素点都刷新关闭了一次。

[0159] 如图4所示,下面结合各个模块,具体介绍本发明一具体实例的OLED屏幕残影减少的实现方法。

[0160] 整个屏幕周期刷新流程分为十二个步骤,图中的①②③④⑤⑥⑦⑧⑨⑩⑪⑫代

表十二个步骤的执行顺序,接下来我们依次进行说明。

[0161] ①对应于步骤S100,状态监控模块一直在监控设备的状态,如果监控到设备即将关机的时候,会保存当前的策略状态,并停止屏幕刷新相关的任务。

[0162] ②对应于步骤S100,状态监控模块会对屏幕的灭屏状态、触屏的点击或滑动状态、按键的按下或弹起状态、设备的屏幕是否发生转屏和屏幕图像的内容是否发生改变来进行综合判断,以此来确认设备是否处于定屏状态。如果如上所述的条件有任何一项不满足要求,都不会认为设备处于定屏状态,只有如上所述的所有条件都满足要求时,才会认为设备处于定屏状态,这时才会开启屏幕的周期刷新任务。

[0163] ③在屏幕的周期刷新任务开始以后,如果设备的如下任何一个状态发生改变,即有发生屏幕灭屏行为、有触屏点击或滑动操作、有任何按键按下或弹起动作发生、设备的屏幕发现发生改变,此时都会立即停止屏幕的周期刷新任务。

[0164] ④对应于步骤S200,在屏幕的周期刷新任务开始以后,屏幕刷新模块会主动加载屏幕刷新策略,例如:最小刷新单元的尺寸、最小刷新单元中每轮要关闭的像素点数量,并通过该刷新策略来对屏幕进行刷新。如果在屏幕刷新过程中,用户通过策略配置界面对刷新策略进行了修改,此时新的刷新策略也会主动同步到屏幕刷新模块中,然后屏幕刷新模块会使用新的刷新策略来进行屏幕的刷新操作。

[0165] ⑤对应于步骤S200,屏幕刷新模块会依据最小刷新单元的尺寸,对当前屏幕的布局进行分割,屏幕本来是以像素点作为最小单元,要想对屏幕进行处理,必须依次遍历每个像素点,现在对屏幕处理的最小单位不再是像素点,而是屏幕刷新策略中所定义的最小刷新单元,那么最小刷新单元就可以作为屏幕遍历的最小单位了。

[0166] ⑥对应于步骤S200中像素点位置的选择,由于对屏幕遍历处理的最小单位是最小刷新单元,而且要求每轮刷新中,最小刷新单元中像素点位置要随机,从而在开始每次的屏幕刷新任务时,要对最小刷新单元中要关闭的像素点位置进行初始化,该初始化操作,是依据每个最小刷新单元中像素点的数量来取一个随机位置值,然后每个最小刷新单元中其它要关闭的像素点位置,再依据该随机值来通过算法计算生成。

[0167] ⑦一个完整的屏幕刷新周期是由多轮屏幕刷新任务组成的,而且每轮屏幕刷新之间都有一定的时间间隔,只有每轮屏幕刷新任务时间到了才会开启本轮的刷新任务,这种计时机制是通过系统的定时任务来实现的,计时机制在屏幕周期刷新功能开启的时候同步开启,在设备灭屏的时候会关闭,并且在屏幕再次点亮后也会重新开启该定时任务。

[0168] ⑧对应于步骤S200中选择每轮屏幕刷新任务中需要关闭的像素点位置,每一轮刷新任务开始以后,要首先获取该轮刷新任务所要关闭的像素点位置,这里的位置获取方式跟每个最小刷新单元中所要关闭的像素点数量比例有关。那么相对于最小刷新单元来讲,如果该比例大于一半,每一个要关闭的像素点位置索引是在前一个要关闭的像素点位置索引基础上加1;如果该比例不大于一半,那么每一个要关闭的像素点位置索引要在前一个要关闭的像素点位置索引基础上加2。如上加1或2以后得到的值,再和最小刷新单元中像素点数量进行求余操作,求余以后的值就是每个最小刷新单元中下一个要关闭的像素点位置。

[0169] ⑨对应于上述的递进刷新,在获取到每个最小刷新单元中要关闭的像素点位置后,此时要判断本轮刷新是否是该周期屏幕刷新任务开始以来的第一轮刷新,如果是第一轮刷新,那么就不需要开启递进累加的刷新功能,而是直接进行屏幕刷新;如果不是第一轮

刷新,那么需要开启递进累加的刷新功能,通过递进累加刷新,可以有效避免因一次性关闭太多像素点,而导致的屏幕闪屏现象。此时,可以根据每轮要关闭的像素点数量,占每个最小刷新单元中像素点数量的比例,来获取递进刷新的次数。如果比例越大,对应的刷新次数也越多,反之对应的刷新次数就越少;同时也可以将每轮的刷新次数设置为一个定值,将每轮刷新都拆分成一个固定次数的递进累加刷新来执行。

[0170] ⑩对应于步骤S400,当前已经获取到了所有要关闭的像素点位置,此时要进行刷新任务了,首先要根据设备分辨率初始化一个数组,并且将该数组的值初始化为0x00ffffff,然后通过要关闭的像素点位置值,将如上数组所对应位置的数值设为0xff000000,该值代表关闭该位置的像素点,然后通过该数组创建一个bitmap位图,并且将此bitmap位图赋值给imageview,然后windowmanager通过该imageview来刷新对应view,最后view屏幕刷新时就会关闭如上数组中所声明要关闭的像素点了。

[0171] ⑪对应于上述的递进刷新,每次刷新时所关闭的像素点位置,一部分是上一轮刷新中被关闭的像素点位置,另一部分是这一轮中要关闭的像素点位置。假如,每一轮的刷新比例为30%,共分7次来递进刷新,那么第一次刷新时,针对上一次被选中关闭的像素点位置,如果该位置所在的最小刷新单元的次序编号跟7求余后值大于0,那么该像素点位置继续关闭,针对这一次被选中关闭的像素点位置,如果该位置所在的最小刷新单元的次序编号跟7求余后的值小于等于0,那么该像素点的位置会关闭,从而第一次刷新关闭的像素点中,有6/7是上一轮刷新所关闭的像素点,有1/7是这一轮刷新要关闭的像素点,同理,第二次刷新时,有5/7是上一轮刷新所关闭的像素点,有2/7是这一轮刷新要关闭的像素点,……,第六次刷新时,有1/7是上一轮刷新所关闭的像素点,有6/7是这一轮刷新要关闭的像素点,第七次刷新时,全部都是这一轮要关闭的像素点。通过如上方法,将每一次刷新要关闭的像素点位置都获取到了,然后具体的刷新任务跟步骤10是一致的。

[0172] ⑫在每轮刷新中,如果递进累加刷新的循环次数结束了,那么该轮刷新任务也就要结束了,此时就要结束该轮的刷新任务,等待开始下一轮的刷新任务。

[0173] 通过如上12个步骤,在设备屏幕定屏时,周期刷新屏幕来关闭一定比例的像素点,并且在每一个刷新周期结束后,可以将屏幕上的所有像素点至少关闭一次,从而可以有效解决因屏幕图像定屏而导致的屏幕烧屏残影的现象了。

[0174] 采用本发明的OLED屏幕残影方法和系统后的效果对比可以参见图5和图6所示。其中,图5是图像的原始效果,图6是通过屏幕刷新随机关闭了30%像素点后的屏幕效果。从图6的显示效果可明显感觉到,虽然关闭了30%的像素点,但基本不会影像用户的视觉体验,而且此时只要用户对设备进行了任何操作,就会立即退出当前的磨砂效果,并切换到原始效果的图像。

[0175] 采用本发明的OLED屏幕残影方法和系统后的验证对比可以参见图7和图8。其中,该验证方法专门设置了图像对比度比较高的显示效果来验证功能的有效性,图7是没有开启屏幕周期刷新功能的原始图像,图8是开启了屏幕周期刷新功能的图像效果。该周期刷新功能会随机关闭30%像素点的图像,并且每间隔90秒进行下一轮的屏幕刷新任务。已经通过此验证方法进行了连续7\*24小时的有效性测试,整个测试过程完成后没有出现屏幕烧屏残影的现象,从而可以很好地说明该发明所提供的方法对解决因屏幕定屏导致的烧屏残影

问题具有积极的效果。

[0176] 本发明实施例还提供一种OLED屏幕残影减少设备,包括处理器;存储器,其中存储有所述处理器的可执行指令;其中,所述处理器配置为经由执行所述可执行指令来执行所述的OLED屏幕残影减少方法的步骤。

[0177] 所属技术领域的技术人员能够理解,本发明的各个方面可以实现为系统、方法或程序产品。因此,本发明的各个方面可以具体实现为以下形式,即:完全的硬件实施方式、完全的软件实施方式(包括固件、微代码等),或硬件和软件方面结合的实施方式,这里可以统称为“电路”、“模块”或“系统”。

[0178] 下面参照图9来描述根据本发明的这种实施方式的电子设备600。图9显示的电子设备600仅仅是一个示例,不应对本发明实施例的功能和使用范围带来任何限制。

[0179] 如图9所示,电子设备600以通用计算设备的形式表现。电子设备600的组件可以包括但不限于:至少一个处理单元610、至少一个存储单元620、连接不同系统组件(包括存储单元620和处理单元610)的总线630、显示单元640等。

[0180] 其中,所述存储单元存储有程序代码,所述程序代码可以被所述处理单元610执行,使得所述处理单元610执行本说明书上述电子处方流转处理方法部分中描述的根据本发明各种示例性实施方式的步骤。例如,所述处理单元610可以执行如图1中所示的步骤。

[0181] 所述存储单元620可以包括易失性存储单元形式的可读介质,例如随机存取存储单元(RAM) 6201和/或高速缓存存储单元6202,还可以进一步包括只读存储单元(ROM) 6203。

[0182] 所述存储单元620还可以包括具有一组(至少一个)程序模块6205的程序/实用工具6204,这样的程序模块6205包括但不限于:操作系统、一个或者多个应用程序、其它程序模块以及程序数据,这些示例中的每一个或某种组合中可能包括网络环境的实现。

[0183] 总线630可以为表示几类总线结构中的一种或多种,包括存储单元总线或者存储单元控制器、外围总线、图形加速端口、处理单元或者使用多种总线结构中的任意总线结构的局域总线。

[0184] 电子设备600也可以与一个或多个外部设备700(例如键盘、指向设备、蓝牙设备等)通信,还可与一个或者多个使得用户能与该电子设备600交互的设备通信,和/或与使得该电子设备600能与一个或多个其它计算设备进行通信的任何设备(例如路由器、调制解调器等等)通信。这种通信可以通过输入/输出(I/O)接口650进行。并且,电子设备600还可以通过网络适配器660与一个或者多个网络(例如局域网(LAN),广域网(WAN)和/或公共网络,例如因特网)通信。网络适配器660可以通过总线630与电子设备600的其它模块通信。应当明白,尽管图中未示出,可以结合电子设备600使用其它硬件和/或软件模块,包括但不限于:微代码、设备驱动器、冗余处理单元、外部磁盘驱动阵列、RAID系统、磁带驱动器以及数据备份存储系统等。

[0185] 通过采用本发明所提供的OLED屏幕残影减少设备,所述处理器在执行所述可执行指令时执行所述的OLED屏幕残影减少方法,由此可以获得上述OLED屏幕残影减少方法的有益效果。

[0186] 本发明实施例还提供一种计算机可读存储介质,用于存储程序,所述程序被处理器执行时实现所述的OLED屏幕残影减少方法的步骤。在一些可能的实施方式中,本发明的各个方面还可以实现为一种程序产品的形式,其包括程序代码,当所述程序产品在终端设

备上运行时,所述程序代码用于使所述终端设备执行本说明书上述电子处方流转处理方法部分中描述的根据本发明各种示例性实施方式的步骤。

[0187] 参考图10所示,描述了根据本发明的实施方式的用于实现上述方法的程序产品800,其可以采用便携式紧凑盘只读存储器(CD-ROM)并包括程序代码,并可以在终端设备,例如个人电脑上运行。然而,本发明的程序产品不限于此,在本文件中,可读存储介质可以是任何包含或存储程序的有形介质,该程序可以被指令执行系统、装置或者器件使用或者与其结合使用。

[0188] 所述程序产品可以采用一个或多个可读介质的任意组合。可读介质可以是可读信号介质或者可读存储介质。可读存储介质例如可以为但不限于电、磁、光、电磁、红外线、或半导体的系统、装置或器件,或者任意以上的组合。可读存储介质的更具体的例子(非穷举的列表)包括:具有一个或多个导线的电连接、便携式盘、硬盘、随机存取存储器(RAM)、只读存储器(ROM)、可擦式可编程只读存储器(EPROM或闪存)、光纤、便携式紧凑盘只读存储器(CD-ROM)、光存储器件、磁存储器件、或者上述的任意合适的组合。

[0189] 所述计算机可读存储介质可以包括在基带中或者作为载波一部分传播的数据信号,其中承载了可读程序代码。这种传播的数据信号可以采用多种形式,包括但不限于电磁信号、光信号或上述的任意合适的组合。可读存储介质还可以是可读存储介质以外的任何可读介质,该可读介质可以发送、传播或者传输用于由指令执行系统、装置或者器件使用或者与其结合使用的程序。可读存储介质上包含的程序代码可以用任何适当的介质传输,包括但不限于无线、有线、光缆、RF等等,或者上述的任意合适的组合。

[0190] 可以以一种或多种程序设计语言的任意组合来编写用于执行本发明操作的程序代码,所述程序设计语言包括面向对象的程序设计语言—诸如Java、C++等,还包括常规的过程式程序设计语言—诸如“C”语言或类似的设计语言。程序代码可以完全地在用户计算设备上执行、部分地在用户设备上执行、作为一个独立的软件包执行、部分在用户计算设备上部分在远程计算设备上执行、或者完全在远程计算设备或集群上执行。在涉及远程计算设备的情形中,远程计算设备可以通过任意种类的网络,包括局域网(LAN)或广域网(WAN),连接到用户计算设备,或者,可以连接到外部计算设备(例如利用因特网服务提供商来通过因特网连接)。

[0191] 通过采用本发明所提供的计算机可读存储介质,其中存储的程序在被执行时实现所述的OLED屏幕残影减少方法的步骤,由此可以获得上述OLED屏幕残影减少方法的有益效果。

[0192] 以上内容是结合具体的优选实施方式对本发明所作的进一步详细说明,不能认定本发明的具体实施只局限于这些说明。对于本发明所属技术领域的普通技术人员来说,在不脱离本发明构思的前提下,还可以做出若干简单推演或替换,都应当视为属于本发明的保护范围。

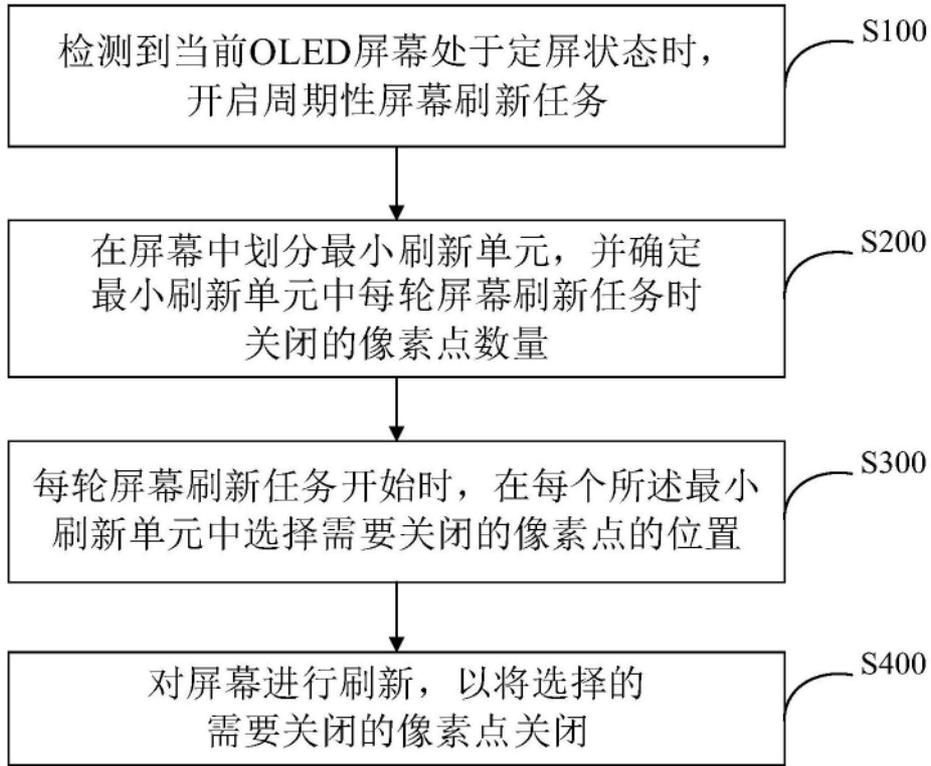


图1

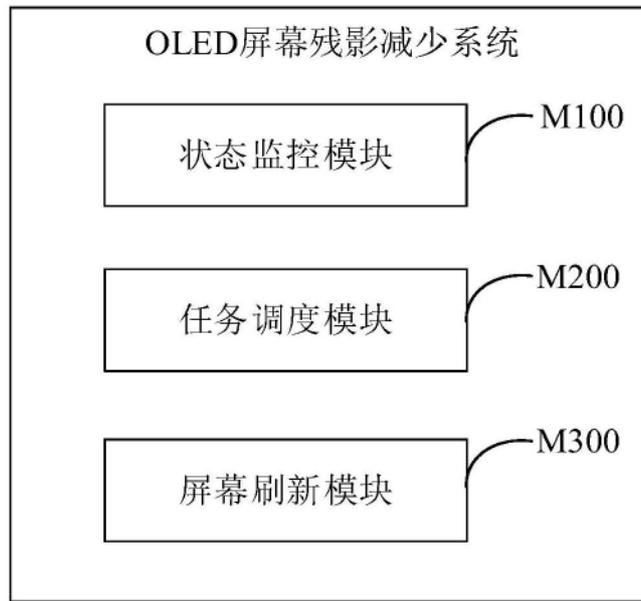


图2

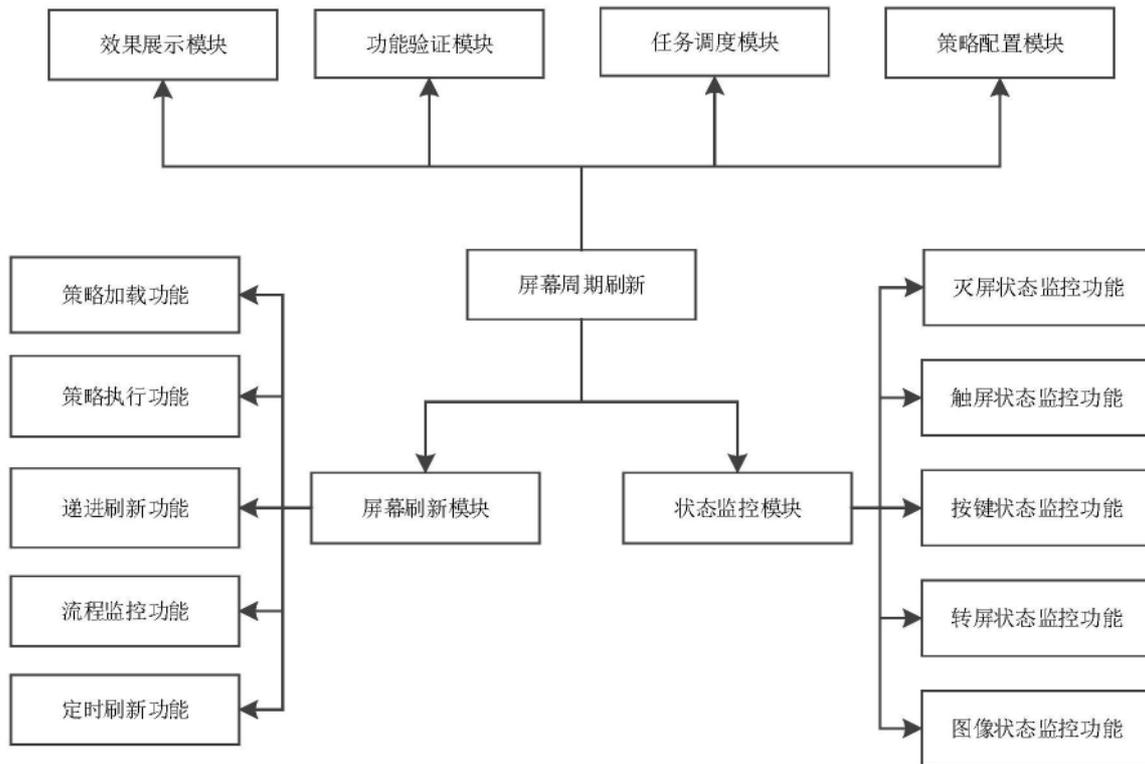


图3

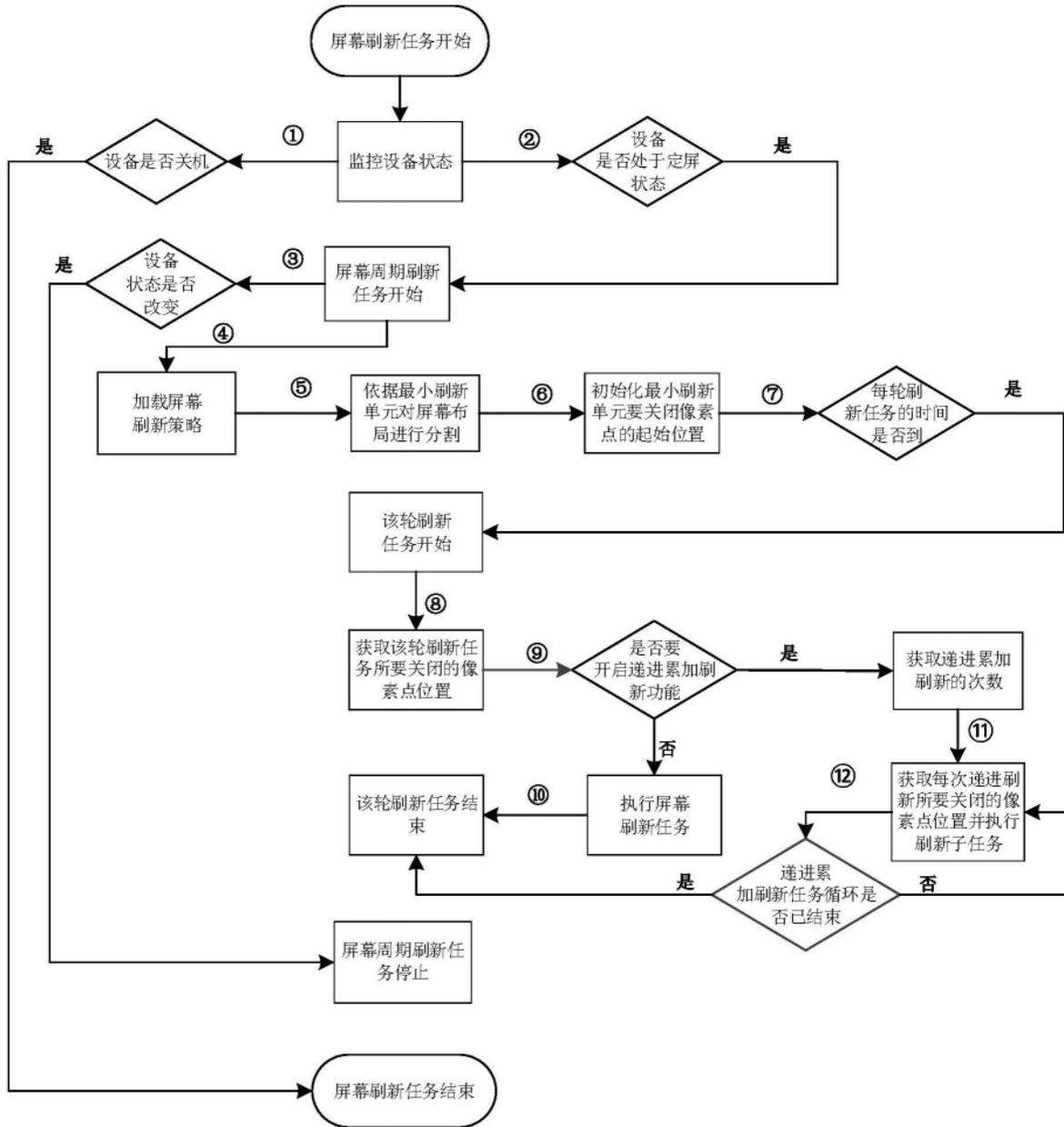


图4



图5

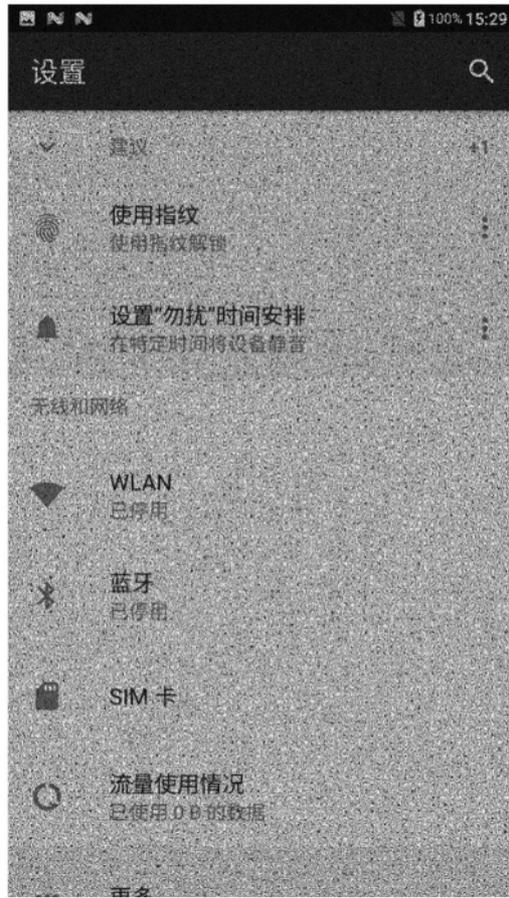


图6

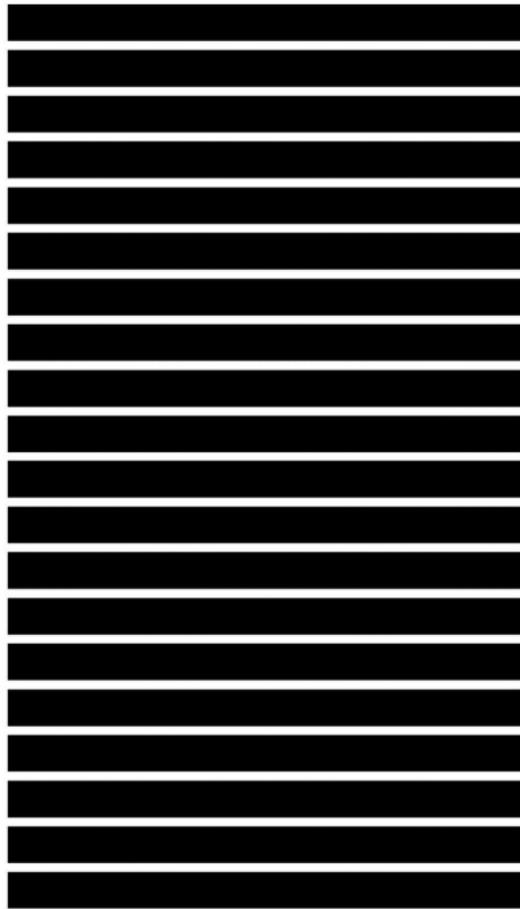


图7

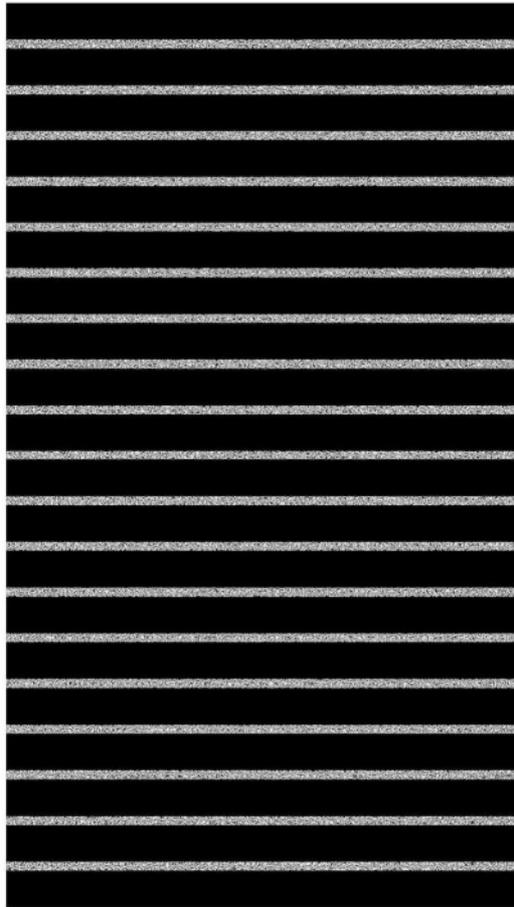


图8

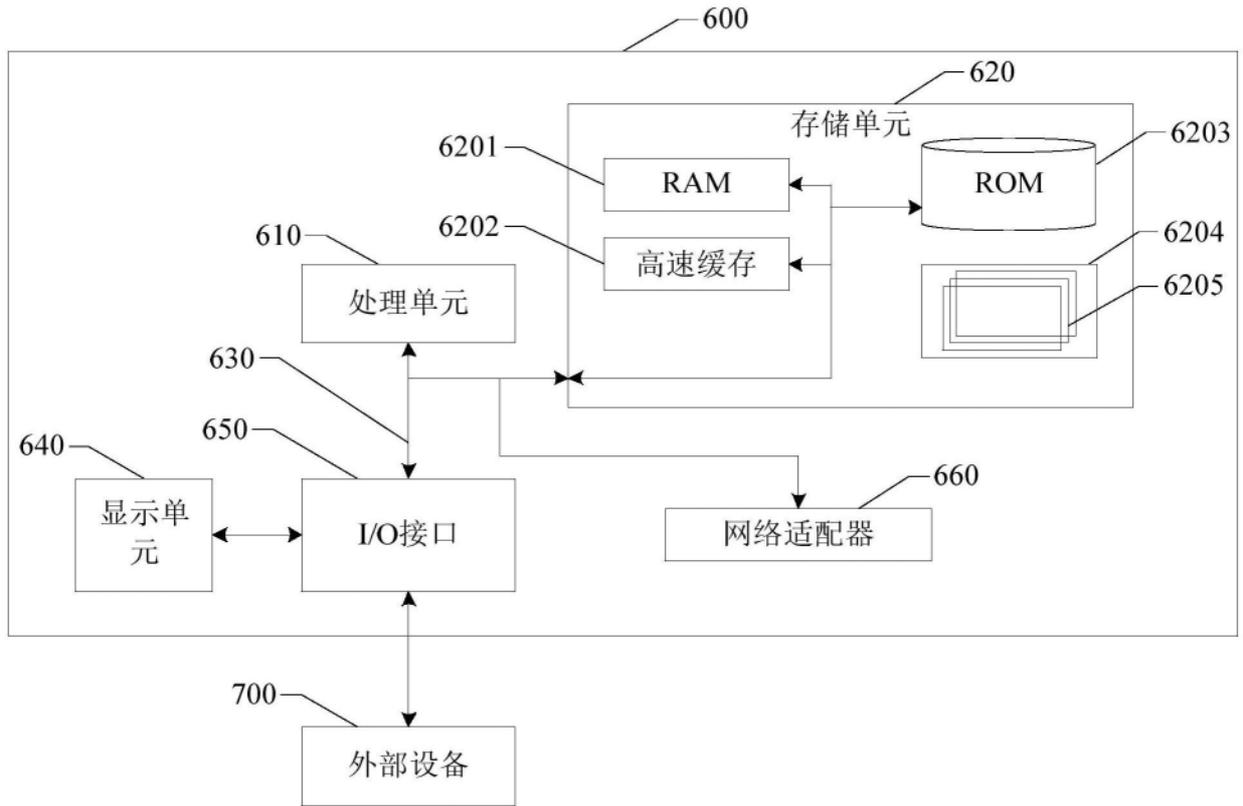


图9

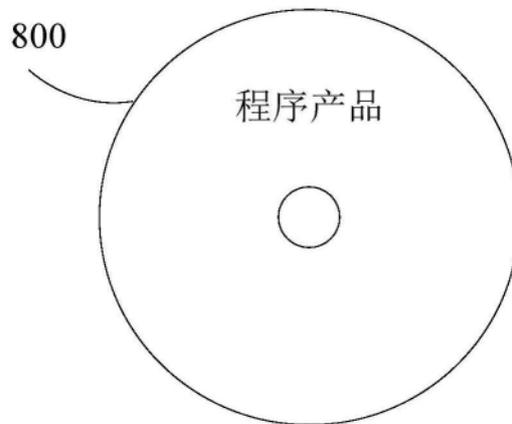


图10