



(12) 发明专利

(10) 授权公告号 CN 114467078 B

(45) 授权公告日 2024. 10. 29

(21) 申请号 202080069035.4

(22) 申请日 2020.09.18

(65) 同一申请的已公布的文献号
申请公布号 CN 114467078 A

(43) 申请公布日 2022.05.10

(30) 优先权数据
62/909,224 2019.10.01 US
16/777,854 2020.01.30 US

(85) PCT国际申请进入国家阶段日
2022.03.30

(86) PCT国际申请的申请数据
PCT/US2020/051375 2020.09.18

(87) PCT国际申请的公布数据
W02021/067047 EN 2021.04.08

(73) 专利权人 微软技术许可有限责任公司
地址 美国华盛顿州

(72) 发明人 C·克莱因 R·彭莱
L·埃德尔迈耶 P·哈默奎斯特
C·P·H·汉森 D·C·纳贝尔

(74) 专利代理机构 北京市金杜律师事务所
11256
专利代理师 豐迅

(51) Int.Cl.
G06F 9/451 (2018.01)
G06F 1/16 (2006.01)
G06F 3/0481 (2006.01)

(56) 对比文件
CN 102197362 A, 2011.09.21
US 2004056900 A1, 2004.03.25
审查员 谢萍

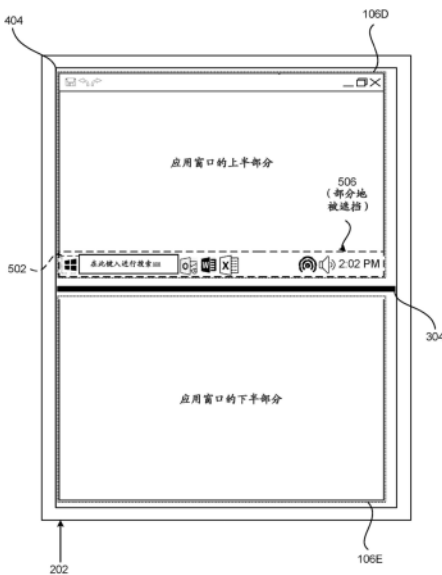
权利要求书3页 说明书16页 附图14页

(54) 发明名称

基于推断的内容遮挡和用户意图的用户界面适配

(57) 摘要

公开了一种可折叠计算设备,其被配置为基于内容遮挡和用户意图的预测来确定用户界面(UI)组件的绘制模式。可折叠计算设备被配置为基于UI组件是否以及在何处遮挡底层应用窗口或其他现有UI来增加、维持或降低UI组件的显眼度。可折叠设备还可以基于与刚刚启动的应用相关联的应用窗口的预期位置来确定现有UI组件的绘制模式。此外,可折叠设备可以预测用户将对应用窗口做什么,包括导致任何未来的遮挡,并相应地调整UI组件的绘制模式。



1. 一种计算机实现的方法,所述方法由计算设备执行,包括:
计算与要被显示的用户界面UI组件相关联的分数;
计算与在所述UI组件被显示的事件中将被所述UI组件遮挡的现有UI相关联的分数;
在与要被显示的所述UI组件相关联的所述分数和与在所述UI组件被显示的所述事件中将被所述UI组件遮挡的所述现有UI相关联的所述分数之间执行比较;
至少部分地基于所述比较,从多个不同的绘制模式中选择针对所述UI组件的绘制模式;以及
以所选择的所述绘制模式显示所述UI组件。
2. 根据权利要求1所述的计算机实现的方法,其中当与要被显示的所述UI组件相关联的所述分数小于与在所述UI组件被显示的所述事件中将被所述UI组件遮挡的所述现有UI相关联的所述分数时,所述绘制模式被设置为第一绘制模式。
3. 根据权利要求1所述的计算机实现的方法,其中当与在所述UI组件被显示的所述事件中将被所述UI组件遮挡的所述现有UI相关联的所述分数小于与要被显示的所述UI组件相关联的所述分数,并且与所述现有UI相关联的所述分数和与要被显示的所述UI组件相关联的所述分数之间的差异不超过预定的量时,所述绘制模式被设置为第二绘制模式。
4. 根据权利要求1所述的计算机实现的方法,其中当与在所述UI组件被显示的所述事件中时将被所述UI组件遮挡的所述现有UI相关联的所述分数比与要被显示的所述UI组件相关联的所述分数小至少预定的量时,所述绘制模式被设置为第三绘制模式。
5. 根据权利要求2所述的计算机实现的方法,其中在所述第一绘制模式中显示所述UI组件包括至少部分透明地或半透明地显示所述UI组件,在所述UI组件在定义的时间段后没有接收用户输入的指示的事件中自动隐藏所述UI组件,以最小化模式演示所述UI组件,或显示将由所述UI组件以正常绘制模式显示的UI子组件的子集。
6. 根据权利要求3所述的计算机实现的方法,其中以所述第二绘制模式显示所述UI组件包括不透明地显示所述UI组件。
7. 根据权利要求4所述的计算机实现的方法,其中以所述第三绘制模式显示所述UI组件包括主动地显示所述UI组件。
8. 根据权利要求7所述的计算机实现的方法,其中主动地显示所述UI组件是基于确定第一应用窗口是包括与所述第一应用窗口同时使用的第二应用窗口的应用工作流的一部分。
9. 根据权利要求8所述的计算机实现的方法,其中所述计算设备包括可折叠计算设备,其中所述应用工作流包括历史上已被显示在所述可折叠计算设备的第一显示区域上的所述第一应用窗口,而所述第二应用窗口历史上已被显示在所述可折叠计算设备的第二显示区域上,并且其中所述UI组件响应于所述第一应用窗口被显示在所述第一显示区域中而被主动地显示在所述第二显示区域中。
10. 根据权利要求9所述的计算机实现的方法,其中所述UI组件可以被激活以在所述第二显示区域中启动所述第二应用窗口。
11. 根据权利要求7所述的计算机实现的方法,其中所述计算设备周期性地评估在多个显示区域中的多个位置处的多个潜在UI组件以标识UI组件以主动地显示在所述多个显示区域中的一个显示区域中。

12. 一种可折叠计算设备, 包括:

一个或多个处理器; 和

至少一个非瞬态计算机可读存储介质, 在其上存储有计算机可执行指令, 所述计算机可执行指令在由所述一个或多个处理器执行时, 使所述可折叠计算设备:

计算与要由所述可折叠计算设备显示的用户界面UI组件相关联的分数;

计算与在所述UI组件被显示的事件中将被所述UI组件遮挡的现有UI相关联的分数;

在与所述UI组件关联的所述分数和与在所述UI组件被显示的所述事件中将被所述UI组件遮挡的所述现有UI关联的所述分数之间执行比较;

至少部分地基于所述比较, 从多个绘制模式中选择针对所述UI组件的绘制模式; 以及使用所选择的所述绘制模式显示所述UI组件。

13. 根据权利要求12所述的可折叠计算设备, 其中:

与所述现有UI相关联的所述分数是至少部分地基于对包括在所述现有UI中的内容或功能中的至少一个的分析而被计算的; 以及

所述分数指示以下中的至少一项: (i) 基于所述内容或所述功能中的至少一者的所述现有UI的预测的效用的量或 (ii) 所述内容或所述功能中的至少一者的重要性。

14. 根据权利要求12所述的可折叠计算设备, 其中与所述UI组件相关联的所述分数是至少部分地基于用户与所述UI组件的其他实例的交互的历史频率而被计算的。

15. 根据权利要求12所述的可折叠计算设备, 其中所述现有UI被划分为部分, 其中针对所述现有UI的每个部分生成分数, 其中所述UI组件被划分为部分, 其中针对所述UI组件的每个部分生成分数, 并且其中所述UI组件的每个部分的所述绘制模式至少部分地基于所述UI组件的该部分的分数和与所述UI组件的该部分重叠的所述现有UI的一个或多个部分的分数而被调整。

16. 根据权利要求15所述的可折叠计算设备, 其中所述现有UI的各部分是基于历史上在所述现有UI上执行的用户交互事件的集群而被标识的。

17. 根据权利要求16所述的可折叠计算设备, 其中所述现有UI由应用窗口生成, 并且其中所述现有UI是基于所述应用窗口的用户界面元素而被划分为多个部分的。

18. 一种非瞬态计算机可读存储介质, 在其上存储有计算机可执行指令, 所述计算机可执行指令在由计算设备执行时, 使所述计算设备:

接收与应用窗口相关联的应用被调用的指示;

确定所述应用窗口的预期位置;

计算与用户界面UI组件相关联的分数, 所述用户界面组件基于所述预期位置与所述应用窗口的至少一部分重叠;

计算与将与所述UI组件重叠的所述应用窗口的所述一部分相关联的分数;

在与所述UI组件相关联的所述分数和与所述应用窗口的所述部分相关联的所述分数之间执行比较;

至少部分地基于所述比较, 从多个绘制模式中选择针对所述UI组件的绘制模式; 以及使用所选择的所述绘制模式显示所述UI组件。

19. 根据权利要求18所述的非瞬态计算机可读存储介质, 其中所述UI组件包括由所述计算设备的操作系统生成的任务栏或选项菜单。

20. 根据权利要求1所述的计算机实现的方法, 其中个体分数至少指示以下中的一项:
(i) 基于UI内容或UI功能中的至少一者的预测的效用的量或 (ii) 所述UI内容或所述UI功能中的至少一者的重要性。

基于推断的内容遮挡和用户意图的用户界面适配

背景技术

[0001] 可折叠计算设备包括具有通过铰链连接的两个屏幕的设备或具有可弯曲屏幕的设备。这些类型的设备可以提供优于传统计算设备(诸如膝上型计算机)的优势。然而,通常,这些设备实现最初为具有传统形状因子的计算设备所设计的用户界面(“UI”)范例。结果,由可折叠计算设备所提供的UI可能很笨重并且容易出错,这会导致不正确的、低效的或无意的用户输入以及不必要的计算资源消耗。

[0002] 本文提出的公开内容正是针对这些和其他技术挑战提出的。

发明内容

[0003] 本文公开了用于基于内容遮挡和用户意图的预测来确定UI组件绘制模式的技术。还公开了用于基于现有应用布局和预期用户工作流程来定位UI组件的技术。所公开的技术通过使得具有多个显示区域的可折叠计算设备能够突出或淡化(de-emphasize)遮挡现有UI内容的UI组件来解决上述技术问题。以这种方式,用户的注意力会被吸引到重要的信息上,而不太重要的信息会被淡化。所公开的技术通过适时地移动UI组件以增强预测的用户工作流程来进一步解决上述技术问题。

[0004] 通过实现所公开的技术,UI可以由更易于利用并且导致更少用户输入错误的可折叠设备来提供。此外,通过避免与UI的低效导航和无意或不正确的用户输入相关联的处理,可以减少可折叠计算设备对计算资源的利用。此处未具体提及的其他技术益处也可以通过实施所公开的主题来实现。

[0005] 基于内容遮挡选择针对UI组件的绘制模式

[0006] 在一个实施例中,可折叠计算设备被配置为基于UI组件是否遮挡现有UI以及在何处遮挡现有UI来增加、维持或降低UI组件的显眼度(prominence)。整个本文档中使用的UI组件的示例包括任务栏、开始菜单和其他操作系统外壳组件,但是类似地设想了来自任何类型的应用的任何类型的UI组件。现有UI是在UI组件将与之重叠的位置中被绘制的任何UI。

[0007] 可折叠设备可以基于与UI组件相关联的分数和与将被UI组件遮挡的现有UI相关联的分数之间的比较来设置UI组件的显眼度。分数可以量化显示UI组件和现有UI对用户的好处。

[0008] 可折叠设备可以通过使用特定绘制模式显示UI组件来设置UI组件的显眼度。绘制模式确定如何显示UI组件,例如选择UI组件的哪些子组件进行显示,什么样的不透明度级别被应用到UI组件,UI组件是否在一段不活动时段后自动隐藏自己,等等。

[0009] 当与UI组件相关联的分数被认为小于与现有UI相关联的分数时,可折叠设备可以以“害羞(shy)”绘制模式显示UI组件。例如,如果具有平均分数的UI组件与现有UI的常用且因此得分高的部分重叠,则该UI组件可能会使用害羞绘制模式来显示。

[0010] 在一些配置中,可以半透明地显示使用害羞绘制模式显示的UI组件的全部或部分。例如,可以将使用害羞绘制模式显示的UI组件的非交互式或背景部分设为半透明,从而

允许通过半透明度感知现有UI。作为另一个示例,可以使使用害羞绘制模式显示并且遮挡现有UI的高得分部分的UI组件的部分比遮挡现有UI的较低得分部分的UI组件的部分更半透明。

[0011] 在一些配置中,可以透明地绘制使用害羞绘制模式显示的UI组件的一个或多个子组件。例如,可以将UI组件的子组件替换为其自身的轮廓,从而创建一个窗口,通过该窗口可以示出现有UI。害羞UI组件还可以省略或减少子组件的占用空间、将子组件嵌套在菜单或选项卡中、自动隐藏或以其他方式变得较小、较少断言或更短暂。

[0012] 例如,如果与UI组件相关联的分数大于与现有UI相关联的分数,则可折叠设备可以使用“正常”模式显示UI组件。使用“正常”模式显示的UI组件可能会像在正常情况下一样显示,占用尽可能多的空间并像通常一样突出出现。

[0013] 如果与UI组件相关联的分数超过与现有UI相关联的分数至少已定义的量,则可折叠设备可以使用“自豪(proud)”模式来显示UI组件。例如,提醒用户注意安全问题的UI组件,特别高得分的UI组件,即使在最近接收到用户交互的现有UI上也可以使用自豪模式显示。

[0014] 在一些配置中,使用“自豪”模式显示的UI组件可以默认被扩展或在用户请求它之前被主动显示。例如,可折叠设备可以分析多个可能显示区域中的多个可能位置中的多个可能UI组件以确定UI组件是否应该被主动显示在特定屏幕位置。

[0015] 在一些配置中,除了将UI组件的绘制模式基于与UI组件重叠的现有UI的存在和分数之外,可折叠设备可以将UI组件的绘制模式基于与刚刚被调用的应用相关联的应用窗口的预期位置。此外,在一些配置中,可折叠设备可以预测用户将如何与一个与正在启动的应用相关联的应用窗口进行交互,包括导致任何未来的遮挡,并主动地调整被用以显示预期与应用窗口重叠的UI组件的绘制模式。

[0016] 适时地移动UI组件以增强预测的用户 workflow

[0017] 在一些配置中,可折叠设备改变UI组件的位置以便减少遮挡。例如,如果用户从第一显示区域中的任务栏调用日历,使得日历通常将被显示在第一显示区域中,如果第一显示区域正在使用中或第二显示区域为空时可折叠设备可以将日历重新定位到第二显示区域。

[0018] 在一些配置中,如果UI组件的调用被确定为是将利用不同的显示区域的工作流的一部分,则可折叠设备可以将UI组件的位置移动到该不同的显示区域。例如,如果当用户打开电子邮件时电子邮件应用在第一显示区域中打开,并且如果第二显示区域为空,则可折叠设备可以确定打开电子邮件是将利用两个显示区域的并排任务流的一部分。在诸如此类的情形中,可折叠设备可以在第二显示区域中显示电子邮件。

[0019] 还应当了解,上述主题可以被实现为计算机控制的装置、计算机实现的方法、计算设备,或者被实现为诸如计算机可读介质的制品。通过阅读以下详细描述并且查看相关附图,这些和各种其他特征将是显而易见的。

[0020] 本发明内容被提供来以简化的形式介绍所公开技术的一些方面的简要描述,所公开技术在详细描述中被进一步描述。本发明内容并非旨在标识所要求保护的主题的关键特征或必要特征,也不旨在使用本发明内容来限制所要求保护的主题的范围。此外,要求保护的主题不限于解决在本公开的任何部分中指出的任何或所有缺点的实现。

附图说明

[0021] 图1A和图1B是图示了以一种特定配置实现所公开技术的铰接的计算设备的配置和操作方面的设备姿态图；

[0022] 图2A-图2C是图示了以一种特定配置实现所公开技术的可弯曲计算设备的配置和操作方面的设备姿态图；

[0023] 图3A和图3B是根据本文公开的一个实施例图示了对可弯曲设备的多个屏幕模式的各种适配的设备用户界面图；

[0024] 图4是根据一个实施例的图示了跨越可折叠设备的两个区域的应用的窗口和使用遮挡应用的窗口的底部部分的“正常”绘制模式所显示的任务栏的设备用户界面图；

[0025] 图5是根据一个实施例的图示了跨越可折叠设备的两个区域的应用的窗口和使用遮挡应用的窗口的中间部分的“害羞”绘制模式所显示的任务栏的设备用户界面图；

[0026] 图6是根据一个实施例的图示了跨越可折叠设备的两个区域的应用的窗口和使用遮挡应用的窗口的中间部分的“害羞”绘制模式所显示的任务栏的设备用户界面图；

[0027] 图7是根据一个实施例的图示了应用的窗口和任务栏的操作的方面的设备用户界面图,该应用的窗口跨越可折叠设备的一个区域并且该任务栏使用“自豪”绘制模式被显示并且在空白显示区域中未经提示地出现；

[0028] 图8是根据一个实施例的图示了在可折叠设备的一个区域中显示的应用的窗口和为了避免遮挡应用的窗口而在空白区域中显示的UI组件的操作的方面的设备用户界面图；

[0029] 图9是根据一个实施例的图示了在可折叠设备的一个区域中显示的应用的窗口和为了避免遮挡应用的窗口而在空白区域中显示的UI组件的操作的方面的设备用户界面图；

[0030] 图10是示出根据一个实施例的例程的流程图,该例程图示了用于基于内容遮挡和用户意图的预测来确定UI组件绘制模式的可折叠设备的操作的方面；

[0031] 图11是示出了针对可以实现本文提出的技术方面的计算设备的说明性计算机硬件和软件架构的计算机架构图。

具体实施方式

[0032] 以下详细描述针对用于基于内容遮挡和用户意图的预测选择针对UI组件的绘制模式的技术。如上面简要讨论的,所公开技术的实现可以使得能够提供更容易利用并且导致更少用户输入错误的UI。因此,与以前的解决方案相比,可以通过避免与UI的低效导航和无意或不正确的用户输入相关联的处理来减少计算资源的利用。本文未具体提及的其他技术益处也可以通过实施所公开的主题来实现。

[0033] 本领域技术人员将认识到,本文公开的主题可以利用各种类型的计算系统和模块来实现,其中至少一些在下面被详细描述。本领域技术人员还将了解,本文描述的主题可以利用其他计算机系统配置来实践,该其他计算机系统配置包括手持设备、多处理器系统、基于微处理器或可编程的消费电子产品、嵌入在设备中的计算或处理系统(诸如可穿戴设备、汽车、家庭自动化等)、嵌入在设备中的计算或处理系统(诸如可穿戴计算设备、汽车、家庭自动化等)等。

[0034] 在以下详细描述中,对附图进行参考,附图形成详细描述的一部分并且通过说明特定配置或示例的方式被示出。现在参考附图,其中相似的数字在几幅附图中表示相似的

元素,将描述用于基于内容遮挡和用户意图的预测来确定UI组件绘制模式的各种技术的各方面。

[0035] 在讨论所公开技术的特定方面之前,将提供对可折叠计算设备(其在本文中可被称为“可折叠设备”)的简要介绍。如上面简要讨论的,可折叠设备包括具有通过铰链或其他等效机构连接在一起的两个物理显示屏的多个屏幕形状因子设备(其在本文中可被称为“铰接设备”)。通过借助铰链操纵显示屏相对于彼此的取向,这样的设备可以被配置成多种姿势,其中一些在下面关于图1A和图1B更详细地描述。

[0036] 可折叠设备还指的是具有可弯曲显示屏的计算设备(在本文中可被称为“可弯曲设备”),诸如利用柔性屏幕技术的计算设备。当这样的设备不弯曲时,它呈现单个显示表面。当被弯曲时,这些设备呈现中间有折痕的单个显示表面。可弯曲设备也可以通过改变弯曲量而被配置成多种姿势,其中一些也将在下面参考图1A和图1B更详细地描述。

[0037] 可折叠计算设备的显示屏可以是触敏的,从而使得这些设备能够识别触摸或触控笔输入、按压、轻扫和其他类型的手势,其中一些在下面被描述。当然,这些设备也可以在被保持在各种取向时被使用,其中一些在下面关于图1A和图1B进行描述。

[0038] 现在参考图1A和图1B,将描述关于针对可折叠显示设备的几种姿势的细节。图1A和图1B中所示的示例图示了铰接的设备102的一些可能姿势。然而,应当了解,所图示的姿势通常也适用于可弯曲设备。

[0039] 如图1A中所示,铰接的设备102可以通过改变连接两个显示设备104A和104B(在本文中可被称为“显示器”)的铰链104的角度而被配置成多种姿势。例如,在图1A中,铰接的设备102被示为显示屏幕104A和104B在纵向取向上彼此平行的平坦姿势、在纵向取向上仅单个显示器104B可见的完全打开姿势、以及在纵向取向上铰链104被弯曲的部分打开姿势。铰接的设备102也可以被配置成两个显示屏都不可见的闭合姿势(未示出)。

[0040] 在图1B中,铰接的设备102被示出:处于平坦姿势,其中显示屏104A和104B处于横向取向;处于帐篷和支撑姿势,其中铰链104的角度使得铰接的设备102自身能够站立并且呈现单个显示器104A;和处于部分打开姿势,其中一个显示器104A是平坦的,而另一个显示器104B至少部分直立,从而形成类似于传统膝上型计算机的配置。在这点上,应当理解,图1A和图1B中所图示的姿势是说明性的并且其他姿势也是可能的。

[0041] 还如图1A和图1B中所示,显示设备104A提供涵盖显示器104A的全部或一部分的第一显示区域106A。类似地,显示设备104B提供涵盖显示器104B的全部或一部分的第二显示区域106B。第一显示区域106A和第二显示区域106B可以呈现图形用户界面(“GUI”)元素、文本、图像、视频、通知、虚拟按钮、虚拟键盘、消息收发数据、互联网内容、设备状态、时间、日期、日历数据、偏好、地图信息、位置信息以及能够以视觉形式呈现的任何其他信息。下面将描述由铰接的设备102提供的显示区域106A和106B的各种用途。

[0042] 现在参考图2A和图2B,将提供关于可弯曲设备202的配置和操作的细节。如上面所提及,可弯曲设备202也可以被配置成与以上关于图1A和图1B描述的姿势相同或相似的姿势。例如,在图2B中所示的示例姿势中,可弯曲设备202已被放置在未弯曲位置,从而呈现整个显示器104C。可弯曲设备202也可以被配置成显示区域都不可见的闭合姿势。

[0043] 在图2B和图2C中所示的示例姿势中,可弯曲设备202已被部分弯曲,类似于图1A和图1B中所示的部分打开姿势。特别地,在图2B中所示的示例中,当屏幕104C处于横向取向

时,可弯曲设备202已被弯曲。在图2C所示的示例中,当屏幕104C处于纵向取向时,设备202已被弯曲。

[0044] 当可弯曲设备202弯曲时,在显示器104C中形成折痕或“折叠”204。如本文中所使用的术语“折叠”可以指的是可折叠设备被折叠的区间(即,铰链的设备102上的铰链104的区间或可弯曲设备202的显示器弯曲的区间)。

[0045] 与铰接的设备102的情况一样,可弯曲设备202也可以提供一个或多个显示区域。然而,在可弯曲设备202的情况下,可用显示区域的数量可以基于设备的姿势而变化。例如,当可弯曲设备202处于如图2A中所示的未弯曲状态时,提供单个显示区域106C。当可弯曲设备202处于弯曲姿势时,诸如图2B中所示,在显示器104C上可以提供两个显示区域106D和106E。下面提供了关于由可折叠设备提供的显示区域106(在整个附图中使用虚线来绘制)的配置和使用的各种细节。

[0046] 现在参考图3A和图3B,将提供关于可弯曲设备202的多个屏幕模式的各种适配的细节。应当了解,为了便于说明,图3A和图3B(以及图4-图9)中所示的可弯曲设备202以二维的方式来示出。然而,这并不旨在表明可弯曲设备202是平放的。

[0047] 在讨论图3A和图3B以及其他图之前,还应当了解,某些相对术语(例如,高度、宽度、顶部、底部、左、右)可以在本文中被利用来描述本文描述的显示器和显示区域的配置。在这点上,还应当理解,这些术语在本文中为了便于讨论而被利用,而不是限制显示区域或UI元素的配置。可以利用其他术语来描述显示器、显示区域、UI元素以及它们彼此之间的空间关系。还应该理解,虽然通常是分开描述的,但是在上面简要描述并且在下面进一步详细描述的各种实施例可以彼此结合利用。

[0048] 如上面简要讨论的,可以在如图3A中所示的纵向取向和如图3B中所示的横向取向之间操纵可折叠设备。同样如上面所讨论的,这些设备可以通过修改铰链的角度而被弯曲到不同程度。在铰接的设备102的情况下,铰链104通常从设备的前部和后部可见。在可弯曲设备202的情况下,从设备的前部看不到铰链,因为可弯曲显示器104C涵盖整个设备。此外,在这两种类型的设备中,铰链都可以配置有传感器,该传感器能够测量铰链的弯曲角度(即“铰链角度”)并且将数据提供给指定铰链角度的设备的处理器。

[0049] 在一个实施例中,可弯曲设备202被配置为基于设备202的铰链角度在针对单个和多个显示区域106优化的交互模式之间动态转变。例如而非限制,当设备202未弯曲时,可弯曲设备202可能表现得好像它具有单个显示屏并且提供单个显示区域106C。在该示例中,显示区域106C延伸跨过整个显示器,包括设备202的在弯曲时将出现折痕204的区域。这种交互模式在本文中可以被称为“单显示区域模式”。

[0050] 当设备202弯曲时,可弯曲设备202还可以表现为双屏设备并且提供多个显示区域106D和106E。在所图示的示例中,显示区域106D涵盖设备202的整个左侧,并且显示区域106E涵盖设备202的整个右侧。这种交互模式在本文中可以被称为“多显示区域模式”。如果设备202处于未弯曲姿势并转变为弯曲姿势,则设备202将从单显示区域模式转变为多显示区域模式。类似地,如果设备202处于弯曲姿势并且转变为未弯曲姿势,则设备202将从多显示区域模式转变为单显示区域模式。

[0051] 在一个实施例中,操作系统在可弯曲设备202的显示屏底部上呈现任务栏302。任务栏302可以提供各种类型的功能性,包括但不限于启动应用、显示通知、显示UI控件以用

于配置设备操作的各方面(例如更改音量级别)、搜索、查看可用应用、显示时间、启动当前可用窗口的视图等。

[0052] 在图3A中所示的实施例中,当处于单显示区域模式中时,可弯曲设备202在显示区域106C的整个底部边缘上显示任务栏302。当设备202从单显示区域模式转变为多显示区域模式时,设备202修改任务栏302的显示以使得它仅涵盖区域106D和106E之一的底部边缘。在图3B中所示的示例中,任务栏302跨越显示区域106D的底部边缘,然而,在其他实施例中,任务栏302可以以类似的方式被呈现在显示区域106E中。在多显示区域模式中,任务栏302常常被呈现在显示区域106D或106E中,但是任务栏302也可以同时被呈现在显示区域106D和106E上。

[0053] 当设备202在多显示区域模式下操作时,从任务栏302启动的应用所呈现的UI将默认出现在与任务栏302相同的显示区域中。例如,如果任务栏302被显示在显示区域106D中并被利用以启动应用,则由启动的应用所提供的UI将被示出在显示区域106D中。类似地,如果任务栏302被显示在显示区域106E中并被利用来启动应用,则由启动的应用所提供的UI将被显示在显示区域106E中。

[0054] 还如图3B中所示,当设备202在多显示区域模式下操作时,可以在折痕204的大致位置处呈现人造硬件接缝304。在一个实施例中,人造硬件接缝304是在设备202的显示屏上显示的显示区域106D和106E之间的竖条。

[0055] 基于内容遮挡选择针对UI组件的绘制模式

[0056] 现在转向图4,将提供关于用于基于UI组件的分数与在UI组件被显示时将被UI组件遮挡的任何现有UI的分数的比较来确定UI组件的UI绘制模式的技术细节。“现有UI”是指应用、对话框或其他UI内容的预先存在的窗口。现有UI可能会或可能不会被UI组件遮挡。例如,如果UI组件的一部分被设为透明,则一些现有UI可能不会被遮挡。如果UI组件被显示在不同的位置,则现有的UI也可能不会被遮挡。“被遮挡的UI”是指实际被UI组件遮挡的现有UI。

[0057] 在一种配置中,可折叠设备102计算与UI组件相关联的分数和与任何现有UI相关联的分数。在一些配置中,分数表示UI组件将提供给用户的效用或其他益处的预测量。例如,可折叠设备102可以将更高的分数指派给向用户提供信息或使用户能够执行功能的UI组件。信息的紧迫性和功能的重要性也可能影响分数。

[0058] 可以基于UI组件的历史使用数据来计算分数。例如,可折叠设备102可以将更高的分数指派给与之交互多于平均UI组件的UI组件。类似地,与平均UI组件相比,可折叠设备102可以将较低分数指派给已被解除、最小化或移动到显示器上的外围位置的UI组件。

[0059] 可折叠设备102然后可以基于UI组件的分数是否超过现有UI的分数以及在何种程度上超过现有UI的分数来改变UI组件的外观。可折叠设备可以通过使用特定绘制模式显示UI组件来改变UI组件的外观。绘制模式确定如何显示UI组件,例如选择UI组件的哪些子组件进行显示,什么级别的不透明度被应用到UI组件,UI组件是否在一段不活动时段后自动隐藏,等等。

[0060] 当分数之间的差异大并且为正时,即,当与UI组件相关联的分数比与现有UI相关联的分数大至少已定义的量时,可折叠设备102可以通过使用自豪绘制模式显示UI组件来给予UI组件高显眼度。

[0061] 当差异为负时,即,UI组件的分数小于被遮挡的UI的分数,那么可折叠设备102可以通过使用害羞绘制模式显示UI组件来给予UI组件低显眼度。当分数之间的差异没有达到这些阈值中的任何一个时——即,与要被显示的UI组件相关联的分数大于与现有UI相关联的分数,但是小于已定义的量,则可折叠设备102可以通过使用正常绘制模式显示UI组件来给予UI组件中等显眼度。

[0062] 虽然在整个本文档中使用了“正常”、“害羞”和“自豪”绘制模式,但是这些名称和描绘只是示例——影响UI组件如何绘制和响应于用户输入的其他类型的绘制模式也被类似地设想。

[0063] 在一个特定示例中,可折叠设备102可以响应于确定与任务栏302相关联的分数大于与应用窗口404的现有UI 406相关联的分数但是小于已定义的量而使用正常绘制模式来显示任务栏302。出于多种原因,任务栏302的分数可能超过应用窗口404的现有UI 406的分数。例如,现有UI 406可以显示非必要信息或根本不显示信息,从而导致较低的分数。

[0064] 此外,任务栏302可以提供功能,例如在运行的应用之间进行切换,该功能比其他UI组件更频繁被使用,从而导致分数增加。任务栏302还可以显示内在的高价值信息,例如安全信息、登录信息、电池寿命信息等,进一步导致分数增加。在一些实施例中,当使用正常绘制模式显示任务栏302时,任务栏302显示一组完整的标准图标和其他子组件。附加地或备选地,当使用正常绘制模式显示任务栏302时,可以将任务栏302设置为完全不透明。

[0065] 在一些配置中,确定任务栏302与比遮挡的UI更高的分数相关联可以包括标识将被遮挡的应用窗口404的一个或多个部分并将这些部分的分数与任务栏302的分数进行比较。每个应用窗口可以与预定的部分列表相关联,例如边缘部分、拐角部分等,并且每个预定部分具有对应的分数列表。部分列表可以对所有应用通用,或者特定于特定应用或应用类型。类似地,与每个部分相关联的分数列表可以对所有应用通用,或者特定于特定应用或应用类型。预定的部分的列表和对应的分数可以由应用窗口404的开发者、遮挡的UI组件(诸如任务栏302)的开发者或者用户手动选择。

[0066] 应用窗口404的各部分也可以基于历史用户界面交互(诸如鼠标光标倾向于在应用窗口内的什么位置、倾向于执行鼠标点击的位置等的历史)而被自动识别。对应允许文本编辑的应用,各部分可以被识别为插入点(也被称为“插入符号”)趋向于所处的区域,或者当接收到键盘输入时插入点趋向于所处的区域。例如,可折叠设备102可以标识发生用户交互的位置集群,并且将应用窗口的一部分定义为包围该集群的几何形状。集群可以是至少具有已定义的密度的一组用户交互位置,例如每小时每平方英寸至少250次交互。

[0067] 在另一个实施例中,可以将一部分自动识别为应用窗口内的一个或多个窗口、对话框、控件或其他用户界面元素。例如,如果可折叠设备102确定在应用窗口处底部处的状态栏内接收到大量鼠标输入,则整个状态栏UI元素可以被认为是应用窗口的一部分。

[0068] 在一些配置中,应用窗口的各部分或与那些部分相关联的值可以基于应用窗口404的位置、大小和/或形状而改变。例如,当应用窗口404的高度大于已定义的量时,或者如果应用窗口404被最大化,则应用窗口404可以沿着底部具有低值部分。然而,如果应用窗口小于已定义的量,例如当所有应用窗口404显示高值内容时,同一应用窗口可能不具有沿着底部的低值部分。在一些配置中,可以分析历史用户交互数据以确定区域变得高值或低值的已定义高度。

[0069] 不管应用窗口的各部分是如何被确定的,从用户与应用窗口404的交互所收集的历史数据都可以被编程地分析以确定哪些部分针对特定用户具有低分、平均分和高分。例如,可以分析类似于这些上述讨论的关于标识部分的用户界面交互的历史,以确定当前用户与这些部分交互的频率。然后可以使用交互频率来确定给定部分的分数。可以基于用户最近进行的交互或基于用户的整个用户交互历史来计算应用窗口的各部分的分数。

[0070] 在一些实施例中,可以基于对各部分内的内容的分析来为应用窗口的各部分分配分数。例如,在整个文档或跨文档重复的文本可能被认为具有低分数,因为重复的文本可能是样板文件。来自应用窗口不同部分的文本也可以由机器学习分类器分析,以识别该部分的分数。在一些配置中,基于内容的分析可以与用户交互历史分析相结合,以将分数分配给应用窗口的各部分。

[0071] 图4图示了使用“正常”绘制模式显示的UI组件,任务栏302。在一些配置中,“正常”绘制模式是应用默认配置的绘制模式,包括不透明度级别、组成UI子组件的数量和类型等。虽然针对不同的UI组件,什么是“正常”的细节可能有所不同,但是许多使用“正常”绘制模式显示的UI组件将几乎或完全不透明。与将图标移动到在访问图标之前必须首先激活的菜单相反,“正常”绘制模式常常也将利用可直接访问的图标来公开功能。“正常”绘制模式常常也将保持固定位置,即在已定义的不活动时间段后不会自动折叠或以其他方式隐藏。

[0072] 虽然任务栏302被用作图4中的UI组件的示例,但是其他UI组件被类似地设想,包括任务启动菜单(例如“开始菜单”)、弹出对话框、安全确认屏幕、固定或“始终在顶部”应用、运行对话框、性能监视器、任务管理器等等。类似地,虽然应用窗口404跨区域106D和106E显示时,当任务栏302遮挡显示在单个区域中的应用窗口时,也可以使用“正常”绘制模式。

[0073] 图5是图示了使用遮挡应用窗口404的中间部分506的“害羞”绘制模式显示的任务栏502的各方面的设备用户界面图。当可折叠设备102确定UI组件将遮挡现有UI中的中心的、重要的或频繁使用的信息时,害羞绘制模式可以被选择。害羞绘制模式可导致一些或全部UI组件透明或半透明。使用害羞绘制模式显示的UI组件也可以呈现最小化的用户界面,例如将图标放置在菜单中、使用较小版本的图标等的用户界面。使用害羞绘制模式显示的UI组件也可以是瞬态的,例如具有自动隐藏自身的用户界面。

[0074] 可折叠设备102可以基于确定与任务栏502相关联的分数小于与现有UI相关联的分数(例如,与应用404的部分506相关联的分数)而使用害羞绘制模式来显示任务栏502。可折叠设备102可以基于与以上结合图4描述的分析类似的分析来做出该确定,例如通过标识应用窗口404中将被UI组件遮挡的各部分的列表,并将每个部分的分数与任务栏502的分数进行比较。如果一个或多个被遮挡部分具有比任务栏502的分数更高的分数,那么可以使用害羞绘制模式来显示任务栏502。

[0075] 在一些配置中,UI组件的一部分可以使用正常绘制模式来显示,而同一UI组件的另一部分可以使用害羞绘制模式来显示。例如,如果任务栏502的部分具有比现有UI的部分更高的分数,则可以使用正常绘制模式来显示任务栏502的那些部分。然而,如果任务栏502的其他部分具有比现有UI的部分更低的分数,则可以使用害羞绘制模式来显示任务栏502的这些其他部分。

[0076] 任务栏502只是可以使用害羞绘制模式显示的UI组件的一个示例——其他UI组件

和其他类型的UI组件,例如窗口、菜单、图标、对话框等可以类似地使用害羞绘制模式来显示。

[0077] 在一些配置中,当确定与UI组件相关联的分数相对于与现有UI相关联的分数时,遮挡本身的负面影响可能会降低与UI组件相关联的分数。以这种方式,与比与现有UI相关联的分数更高的分数相关联的UI组件仍可以使用害羞模式来显示。例如,如果UI组件和现有UI的各部分按1-100的标度评分,1表示很小到没有值,100表示最高值,那么如果分数为55的UI组件遮挡了与分数为53相关联的现有UI,则分数为55的UI组件通常将被设置为正常绘制模式。然而,如果可折叠设备102确定遮挡本身将UI组件的分数降低了5,有效分数为50,那么UI组件将使用害羞绘制模式来显示——尽管名义上具有高于现有内容的更大分数。

[0078] 害羞绘制模式可以导致像任务栏502这样的UI组件是半透明的,如图5中的虚线所描绘的。通过半透明地显示任务栏502,可折叠设备102允许现有UI 506也是可见的,同时维持任务栏502的正常数量和类型的子组件。如所图示,任务栏502包括与正常绘制的任务栏302相同数量的图标和其他子组件。

[0079] 在一些配置中,可以基于与现有UI相关联的分数大于与UI组件相关联的分数的程度来设置半透明级别。半透明度可以被设置为可归因于现有UI的半透明图像的百分比。例如,70%的半透明度百分比意味着UI组件和现有UI之间的混合有70%来自现有UI。70%的半透明度百分比可能被认为是高的,在现有UI具有明显高于UI组件的价值的情况下被保留。

[0080] 例如,如果UI组件具有的分数为20而现有UI具有的分数为65,则对于45的净分差,可折叠设备102可以将UI组件的半透明度设置为70%。较小的分差,例如分数为55的UI组件和分数为65的现有UI,可以对应于50%的半透明度,因为现有UI和UI组件具有相当的分数。

[0081] 使用害羞绘制模式显示类似任务栏502的UI组件也可以导致任务栏502是部分透明的,即任务栏502的子组件可以被绘制为轮廓,从而允许现有UI出现在轮廓内。例如,搜索框508可以允许现有UI除了搜索框的轮廓和其中包含的任何文本之外完全可见。

[0082] 现在转向图6,将提供关于跨越可折叠设备102的区域106的应用窗口404和使用遮挡应用窗口404的中间部分606的“害羞”绘制模式所显示的任务栏602的细节。图6中描绘的害羞绘制模式将任务栏602置于精简模式,省略非必要的子组件,将子组件的尺寸最小化,将子组件放置在菜单和选项卡中,并且以其他方式采取措施来减小任务栏602的整体尺寸。可以不透明地绘制任务栏602的精简模式,以使得应用窗口404中的一些保持被遮挡。

[0083] 在一些配置中,也可以将任务栏602制成半透明的。用于使用害羞绘制模式显示UI组件的其他技术包括自动隐藏任务栏602、将任务栏602缩小为可以被激活以显示一些或全部任务栏602的单个图标等。例如,任务栏602可以在已定义的未使用时段后被叠成单个图标。可折叠设备102可以检测对单个图标的触摸,并且作为响应,恢复一些或全部任务栏602。

[0084] 现在转向图7,将提供关于跨越可折叠设备102的区域106D的应用窗口704和使用“自豪”绘制模式显示的并在空白显示区域106E中未经提示出现的任务栏702的细节。在一些实施例中,当使用“自豪”绘制模式显示UI组件(诸如任务栏702)时,与使用正常模式或自动隐藏模式相比,UI组件使用扩展模式显示UI组件。扩展模式可以填充区域106D的未使用

部分,以使得自豪UI组件占用它所需要的尽可能多的空间而不会造成遮挡。如图7中所描绘的,如果任务栏702要出现在空白区域106E中,则可折叠设备102确定将不存在遮挡。

[0085] 可折叠设备102可以确定与任务栏702相关联的分数大于与区域106E的空白空间相关联的分数。例如,由于应用窗口默认位于被用来启动它们的UI组件的显示区域中,因此与区域106E中显示任务栏702相关联的分数可以部分地基于使用户便于启动在区域106E中具有用于与应用窗口704并排操作的窗口的第二应用。该分数可以大于区域106E中的空白空间的分数——即,该分数可以大于任务栏702的不打扰用户的主动提供的出现的分数。可折叠设备102可以基于应用窗口704和第二应用窗口历史上以并排配置使用的频率来确定用户会重视启动具有与应用窗口704并排操作的窗口的第二应用。

[0086] 在一些配置中,当使用自豪绘制模式自发地显示任务栏702时,任务栏702可以显示、突出显示或以其他方式引起对历史上与应用窗口704同时执行的其他应用的注意,例如历史上已经以并排模式与应用窗口704一起执行的其他应用。例如,任务栏702可以包括与应用窗口704相关联的应用的“快速启动”图标。在一些配置中,在自豪绘制模式中提供的“快速启动”图标可以不是由任务栏在正常或害羞绘制模式中提供的图标。

[0087] 此外,在自豪绘制模式中提供的“快速启动”图标可以基于应用窗口704、绘制应用窗口704的应用的类型、用户最近在应用窗口704中采取的动作等而变化。当应用通常与应用窗口704并排显示时,这使得可折叠设备102的用户能够方便地在区域106E中启动应用。在一些配置中,可折叠设备102周期性地分析潜在的UI组件,如任务栏702,以确定是否有任何潜在的UI组件具有足够高的分数以保证在没有用户输入的情况下主动显示自己。

[0088] 当预测用户想要将另一个应用启动到区域106E中时在空白空间中显示任务栏702是将UI组件设置为自豪绘制模式的一个原因。类似地考虑其他UI组件,诸如应用启动菜单(例如,“开始”菜单),是作为用于确定与使用自豪绘制模式显示UI组件相关联的分数何时大于不是使用自豪绘制模式显示UI组件的其他标准。进一步考虑了使用其他绘制模式(例如害羞或正常)显示的UI组件可以被显示在非默认位置,诸如在默认显示区域上的不同位置或在不同显示区域上的一个位置。

[0089] 在一些配置中,除了考虑由UI组件在现有UI上创建的实际遮挡之外,可折叠设备102可以基于任务上下文(即,已经运行的应用的数量、类型和位置)来预测最近启动的应用窗口将在哪里显示以及用户将如何与最近启动的应用窗口交互。

[0090] 例如,用于查看视频内容的应用窗口可以具有用于浏览视频的一个任务上下文和用于观看全屏视频的不同的任务上下文。文字处理应用窗口可以具有用于编辑文档的第一任务上下文、用于查看文档的第二任务上下文、用于审阅文档的第三任务上下文等。该任务上下文信息可以被用于在特定的绘制模式中显示UI组件。例如,如果当前任务上下文和历史使用数据指示最近启动的Web浏览器可以被用来查看全屏视频,则与最近启动的应用窗口重叠或预期与最近启动的应用窗口重叠的任何UI组件都可以被设置为害羞绘制模式。

[0091] 现在转向图8,将提供关于在可折叠设备102的区域106D中运行的应用窗口704和在空白区域106E中显示以避免遮挡应用窗口704的UI组件806的细节。在该示例中,任务栏302包括时钟显示802。响应于指针804的触摸,时钟显示802可以使UI组件806、日历被绘制。指针804可以是手指、触控笔、笔或其他能够与区域106D的触摸传感器交互的设备。通常,时钟显示802将在时钟显示802所在的区域106D中显示UI组件806。

[0092] 然而,所公开实施例的各方面使UI组件806被显示在区域106E中。在该示例中,区域106E是空白的,并且因此与在区域106E中显示UI组件806相关联的分数被确定为大于与在区域106D中绘制组件806相关联的分数,在该区域中它将遮挡应用窗口704的一部分。因此,可折叠设备102使用自豪绘制模式显示UI组件806,从而使UI组件806被主动显示在区域106E中。在一些实施例中,如果可折叠设备102在UI组件806被显示的区域106E的部分中显示其他内容,从而UI组件806遮挡区域106E中的其他内容,则可折叠设备102可以将UI组件806重新定位到区域106D中的默认位置。

[0093] 现在转向图9,将提供关于在可折叠设备的区域106D中运行的应用窗口704和在空白区域106E中显示以避免遮挡应用窗口704的UI组件的细节。在一些配置中,应用启动菜单906由指针804触摸图标902来调用。通常,应用启动菜单906将出现在区域106D中的图标902上方。然而,可折叠设备102可以确定区域106E是空白的,该应用启动菜单906可以遮挡应用窗口704,并且因此在区域106E中显示应用启动菜单906比在区域106D中有更多值。

[0094] 确定在区域106E中显示应用启动菜单906的另一个因素是,通过经由应用启动菜单906启动的应用所绘制的任何应用窗口908将位于区域106E中,从而创建与应用窗口704并排的工作流。这使得可折叠设备102能够在区域106E中显示应用启动菜单906,以便进入并排显示应用窗口704和不同应用窗口的应用工作流。例如,可折叠设备102可以确定应用窗口704常常是与文字处理应用并排的应用工作流的一部分。虽然描述了并排工作流,但是这只是一个示例。类似地设想了其他类型的工作流,包括父子、平铺、标签或其中两个或更多应用可以在显示时相关的任何其他方式。工作流还可以包括时间或顺序组件,例如将工作流中的应用呈现给用户的已定义顺序。

[0095] 应用窗口704常常是具有文字处理应用的并排工作流的一部分的确定可以基于以与文字处理应用的并排配置来使用应用窗口704的历史来做出。在一些配置中,该确定可以基于以与任何文字处理应用的并排配置来使用应用窗口704的历史。该确定可以基于当前用户的应用使用历史、相关用户的应用历史或者通常基于用户的应用历史。用户可以基于共同的地理位置、共同的工作类型、一组共享的常用应用、年龄、性别、设备类型或可能与推断由一个用户使用的工作流将对其他用户有用相关的任何其他属性相关。

[0096] 响应于确定应用窗口704通常是并排工作流的一部分,可折叠设备102可以在区域106E中主动地显示应用启动菜单906。以这种方式,如果文字处理应用被启动,则应用窗口704和文字处理应用将成为所确定的并排工作流的一部分。为了进一步促进并排工作流,可折叠设备102还可以在应用启动菜单906内显示启动文字处理应用的图标910、超链接或其他方式。在一些配置中,图标910可以被放置得比在应用窗口704不是具有文字处理应用的工作流的一部分的情况下更显眼。尽管图9描绘了应用启动菜单906,但是相同的概念可以被应用于其他类型的UI,包括任务栏、运行菜单、桌面图标或可以启动应用的任何其他方式。

[0097] 在一些配置中,可折叠设备102可以使从区域106D中的图标912启动的应用被主动显示在非默认区域中,例如106E。可折叠设备102可以这样做以便利用应用窗口704的将应用输入到应用工作流中。在一些配置中,当应用窗口704与应用工作流中的应用一起使用的频率满足或超过已定义的阈值时,与可以被用来启动应用的UI的相反,可折叠设备102可以在非默认区域中主动显示应用本身。

[0098] 图10是示出了例程1000的流程图,其图示了用于基于内容遮挡和用户意图的预测来确定UI组件的绘制模式的可折叠设备102的操作的各方面。应当理解,在本文中关于图10和其他图描述的逻辑操作可以被实现为(1)在计算设备上运行的计算机实现的动作或程序模块的序列和/或(2)在计算设备内的互连机器逻辑电路或电路模块。

[0099] 本文公开的技术的特定实现是取决于计算设备的性能和其他要求的选择问题。因此,在本文中描述的逻辑操作被不同地称为状态、操作、结构设备、动作或模块。这些状态、操作、结构设备、动作和模块可以以硬件、软件、固件、专用数字逻辑以及它们的任何组合来实现。应当理解,可以执行比图中所示以及本文中描述的更多或更少的操作。这些操作也可以以不同于本文描述的顺序来执行。

[0100] 例程1000开始于操作1002,其中可折叠设备102确定与显示UI组件402相关联的分数,如上面详细描述的。例程1000然后进行到操作1004,在操作1004中,可折叠设备102确定与被UI组件402在视觉上遮挡的现有UI 406相关联的分数,如上面详细描述的。

[0101] 例程1000然后进行到操作1006,在操作1006,可折叠设备102确定在显示UI组件402时使用哪种绘制模式,如上面详细描述的。如果与UI组件402相关联的分数小于与被遮挡内容406相关联的分数,则例程1000进行到操作1008,在操作1008,可折叠设备102使用害羞绘制模式来显示UI组件402,如以上详细描述的。

[0102] 如果与UI组件402相关联的分数大于或等于与现有UI 406相关联的分数,那么例程1000进行到操作1010,在操作1010,可折叠设备102使用正常绘制模式来显示UI组件402,如上面详细描述的。如果UI组件402的分数比现有UI 406高出已定义的余量,那么可折叠设备可以使用自豪绘制模式来显示UI组件402。

[0103] 在一些配置中,如果可折叠设备102确定UI组件402通常会被显示在区域106D中但可以被显示在区域106E中而不遮挡其他内容,那么例程1000进行到操作1012,在操作1012,可折叠设备102使用自豪绘制模式来显示UI组件402,如上面详细描述的。例程1000从操作1008和1010进行到操作1018,它在操作1018结束。

[0104] 例程1000从操作1012进行到操作1014,在操作1014,可折叠设备102确定UI组件402是否适合区域106E的空白部分。如果UI组件402可以适合区域106E的空白部分,则例程1000进行到操作1016,在操作1016中,可折叠设备102将UI组件402移动到区域106E的空白部分,如上面详细描述的。如果UI组件402不能适合区域106E的空白部分,则例程1000进行到操作1018,它在操作1018结束。

[0105] 图11是示出了可以实现本文提出的各种技术的用于计算设备的说明性计算机硬件和软件架构的计算机架构图。特别地,图11中所图示的架构可以被利用来实现本文所述的铰接的设备102和可弯曲设备202。所图示的架构还可以被利用来实现其他类型的计算系统。

[0106] 图11中图示的计算机1100包括中央处理单元1102(“CPU”)、包括随机存取存储器1106(“RAM”)和只读存储器(“ROM”)1108的系统存储器1104、以及将存储器1104耦合到CPU 1102的系统总线1110。包含基本例程的基本输入/输出系统(“BIOS”或“固件”)可以被存储在ROM1108中,基本例程例如在启动期间有助于在计算机1100内的元件之间传送信息。计算机1100还包括大容量存储设备1112,该大容量存储设备1112用于存储操作系统1122、应用程序1130和其他类型的程序。上述功能由这些程序中的一个或多个以各种配置来实现。大

容量存储设备1112还可以被配置为存储其他类型的程序和数据。

[0107] 大容量存储设备1112通过连接到总线1110的大容量存储控制器(未示出)连接到CPU 1102。大容量存储设备1112及其关联的计算机可读介质为计算机1100提供非易失性存储。尽管本文所包含的计算机可读介质的描述指的是大容量存储设备,诸如硬盘、CD-ROM驱动器、DVD-ROM驱动器或USB存储密钥,但是本领域技术人员应当理解,计算机可读介质可以是计算机1100可以访问的任何可用的计算机存储介质或通信介质。

[0108] 通信介质包括计算机可读指令、数据结构、程序模块或已调制数据信号中的其他数据(诸如载波或其他传输机制),并且包括任何递送介质。术语“调制数据信号”意指以某种方式改变或设置其一个或多个特性以便在信号中对信息进行编码的信号。作为示例而非限制,通信介质包括诸如有线网络或直接有线连接之类的有线介质以及诸如声学、射频、红外线和无线介质之类的无线介质。以上任何内容的组合也应被包括在计算机可读介质的范围内。

[0109] 作为示例而非限制,计算机存储介质可以包括以用于存储诸如计算机可读指令、数据结构、程序模块或其他数据之类的信息的任何方法或技术来实现的易失性和非易失性、可移动和不可移动介质。例如,计算机存储介质包括但不限于RAM、ROM、EPROM、EEPROM、闪存或其他固态存储器技术、CD-ROM、数字多功能磁盘(“DVD”)、HD-DVD、BLU-RAY,或其他光学存储、磁盒、磁带、磁盘存储或其他磁存储设备或可以被用来存储所需信息并且可由计算机1100访问的任何其他介质。对于权利要求,短语“计算机存储介质”及其变体不包括波或信号本身或通信介质。

[0110] 根据各种配置,计算机1100可以使用通过诸如网络1120的网络到远程计算机的逻辑连接的网络环境中操作。计算机1100可以通过连接到总线1110的网络接口单元1116来连接到网络1120。应当理解,网络接口单元1116也可以被利用以连接到其他类型的网络和远程计算机系统。计算机1100还可以包括输入/输出控制器1118,用于接收和处理来自许多其他设备的输入,包括键盘、鼠标、触摸输入、数字笔或诸如相机和生物传感器之类的物理传感器。

[0111] 计算机1100还可以配置有合适的视频输出设备,该视频输出设备可以向一个或多个显示屏提供输出,诸如上述那些。一个或多个显示器可以是被配置为检测触摸的存在和位置的触敏显示器。这种显示器可以是电阻式触摸屏、电容式触摸屏、表面声波触摸屏、红外触摸屏、光学成像触摸屏、色散信号触摸屏、声脉冲识别触摸屏或者可以利用任何其他触摸屏技术。在一些配置中,触摸屏作为透明层而被并入到显示器顶部上,以使得用户能够使用一个或多个触摸来与显示器上呈现的对象或其他信息进行交互。

[0112] 触敏显示器可以被配置为检测离散的触摸、单触摸手势和/或多触摸手势。为方便起见,这些在本文中被统称为“手势”。现在将描述几个手势。应该理解,这些手势是说明性的,并且不旨在限制所附权利要求的范围。

[0113] 在一些配置中,计算机1100支持用户轻击显示器一次的轻击手势。还可以支持用户轻击显示器两次的双击手势。可以出于各种原因使用双击手势,包括但不限于分级放大或缩小。在一些配置中,计算机1100支持轻击并保持手势,其中用户轻击并维持接触至少预定义时间。可以出于各种原因使用轻击并保持手势,包括但不限于打开特定于上下文的菜单。

[0114] 在一些配置中,计算机1100支持平移手势(pan gesture),其中用户将手指放置在显示器上并在移动其手指的同时维持与显示器的接触。可以出于各种原因使用平移手势,包括但不限于以受控速率在屏幕、图像或菜单中移动。多个手指平移手势也可被设想。

[0115] 在一些配置中,计算机1100支持轻拂手势,其中用户在用户希望屏幕移动的方向上轻扫手指。可以出于各种原因使用轻拂手势,包括但不限于水平或垂直滚动菜单或页面。在一些配置中,计算机1100支持捏合和拉伸手势,其中用户用两个手指(例如,拇指和食指)做出捏合运动或将两个手指移开。可以出于各种原因使用捏合和拉伸手势,包括但不限于逐渐放大或缩小网站、地图或图片。

[0116] 尽管已经参考使用一个或多个手指来执行手势从而呈现了上述手势,但是可以使用诸如数字笔之类的其他附件来与计算设备1100交互。如此,上述手势应该被理解为说明性的,而不应被解释为以任何方式进行限制。

[0117] 应当理解,本文所述的软件组件在被加载到CPU 1102中并被执行时,可以将CPU 1102和整个计算机1100从通用计算设备变换为定制为促进本文提出的功能性的专用计算设备。CPU 1102可以由任意数量的晶体管或其他分立电路元件构成,它们可以单独地或共同地显露任意数量的状态。更具体地,CPU 1102可以操作为有限状态机,以响应包含在本文公开的软件模块内的可执行指令。这些计算机可执行指令可以通过指定CPU 1102如何在状态之间转变来变换CPU 1102,从而变换构成CPU 1102的晶体管或其他离散硬件元件。

[0118] 对本文提出的软件模块进行编码还可以变换本文提出的计算机可读介质的物理结构。在本说明的不同实现中,物理结构的具体变换取决于各种因素。这样的因素的示例包括但不限于用于实现计算机可读介质的技术、计算机可读介质是被表征为主存储装置还是辅助存储装置等。例如,如果计算机可读介质被实现为基于半导体的存储器,则本文公开的软件可以通过变换半导体存储器的物理状态而被编码在计算机可读介质上。例如,该软件可以变换构成半导体存储器的晶体管、电容器或其他分立电路元件的状态。该软件还可以变换这些组件的物理状态,以便在其上存储数据。

[0119] 作为另一示例,本文公开的计算机可读介质可以使用磁性或光学技术来实现。在这样的实现中,当在其中编码软件时,本文提出的软件可以变换磁性或光学介质的物理状态。这些变换可以包括改变给定磁介质内的特定位置的磁特性。这些变换还可以包括改变给定光学介质内的特定位置的物理特征或特性,以改变那些位置的光学特性。在不背离本说明书的范围和精神的情况下,物理介质的其他变换是可能的,提供前述示例只是为了便于讨论。

[0120] 鉴于上述情况,应当理解,许多类型的物理变换发生在计算机1100中以便存储和执行本文提出的软件组件。还应当理解,针对计算机1100的图11中所示的架构或类似架构可以被利用来实现其他类型的计算设备,包括手持计算机、视频游戏设备、嵌入式计算机系统、移动设备诸如智能手机、平板电脑和AR/VR设备以及本领域技术人员已知的其他类型的计算设备。还设想了计算机1100可以不包括图11中所示的所有组件,可以包括未在图11中明确示出的其他组件,或者可以利用与图11中所示的架构完全不同的架构。

[0121] 应当了解,为了便于讨论,图11中所示的计算架构已被简化。还应当理解,所图示的计算架构可以包括和利用本文未具体描述的更多的计算组件、设备、软件程序、联网设备和其他组件。

[0122] 本文提出的公开内容还包括在以下条款中阐述的主题：

[0123] 条款1.一种由计算设备102执行的计算机实现的方法,包括:计算与要被显示的用户界面(UI)组件402相关联的分数;计算与在UI组件402被显示时将被UI组件402遮挡的现有UI 406相关联的分数;在与要被显示的UI组件402相关联的分数和与在UI组件402被显示时将被UI组件402遮挡的现有UI 406相关联的分数之间执行比较;至少部分地基于比较选择针对UI组件402的绘制模式;以及以选定的绘制模式显示UI组件402。

[0124] 条款2.根据条款1所述的计算机实现的方法,其中当如果要被显示的UI组件相关联的分数小于与如果UI组件被显示时将被UI组件遮挡的现有UI相关联的分数时,绘制模式被设置为第一绘制模式。

[0125] 条款3.根据条款1和2所述的计算机实现的方法,其中当与如果UI组件被显示时将被UI组件遮挡的现有UI相关联的分数小于与要被显示的UI组件相关联的分数并且与现有UI相关联的分数和与要被显示的UI组件相关联的分数之间的差异不超过预定的量时,绘制模式被设置为第二绘制模式。

[0126] 条款4.根据条款1-3所述的计算机实现的方法,其中当与如果UI组件被显示时将被UI组件遮挡的现有UI相关联的分数小于与要被显示的UI组件相关联的分数至少预定的量时,绘制模式被设置为第三绘制模式。

[0127] 条款5.根据条款1-4所述的计算机实现的方法,其中以第一绘制模式显示UI组件包括:至少部分透明或半透明地显示UI组件,如果UI组件在已定义的时间段后没有接收到用户输入的指示则自动隐藏UI组件,以最小化模式显示UI组件,或显示将由UI组件以正常绘制模式显示的UI子组件的子集。

[0128] 条款6.根据条款1-5所述的计算机实现的方法,其中以第二绘制模式中显示UI组件包括不透明地显示UI组件。

[0129] 条款7.根据条款1-6所述的计算机实现的方法,其中以第三绘制模式显示UI组件包括主动显示UI组件。

[0130] 条款8.根据条款1-7所述的计算机实现的方法,其中主动显示UI组件是基于确定第一应用窗口是包括与第一应用窗口同时使用的第二应用窗口的应用工作流的一部分。

[0131] 条款9.根据条款1-8所述的计算机实现的方法,其中计算设备包括可折叠计算设备,其中应用工作流包括历史上已被显示在可折叠计算设备的第一显示区域上的所述第一应用窗口,而第二应用窗口历史上已被显示在可折叠计算设备的第二显示区域上,并且其中UI组件响应于第一应用窗口被显示在第一显示区域中,而被主动显示在第二显示区域中。

[0132] 条款10.根据条款1-9所述的计算实现的方法,其中UI组件可以被激活以在第二显示区域中启动第二应用窗口。

[0133] 条款11.根据条款1-10所述的计算机实现的方法,其中计算设备周期性地评估在多个显示区域中的多个位置处的多个潜在UI组件以标识UI组件从而主动显示在多个显示区域中的一个显示区域中。

[0134] 条款12.一种可折叠计算设备301,包括:一个或多个处理器1102;和在其上存储有计算机可执行指令的至少一个非瞬态计算机可读存储介质1112,该计算机可执行指令在由一个或多个处理器1102执行时,使可折叠计算设备102:计算与要由可折叠计算设备102显

示的用户界面 (UI) 组件402相关联的分数;计算与如果UI组件402被显示时将被UI组件402遮挡的现有UI 406相关联的分数;在与UI组件402相关联的分数和与在UI组件402被显示时将被UI组件402遮挡的现有UI 406相关联的分数之间执行比较;至少部分地基于该比较选择针对UI组件402的绘制模式;并且使用选定的绘制模式412显示UI组件402。

[0135] 条款13.根据条款12所述的可折叠计算设备,其中与现有UI相关联的分数至少部分地基于对包括在现有UI中的一段内容的内容分析。

[0136] 条款14.根据条款12和13所述的可折叠计算设备,其中与UI组件相关联的分数至少部分地基于用户与现有UI的其他实例的交互的历史频率。

[0137] 条款15.根据条款12-14所述的可折叠计算设备,其中现有UI被划分为部分,其中针对现有UI的每个部分生成分数,其中UI组件被划分为部分,其中针对UI组件的每个部分生成分数,并且其中UI组件的每个部分的绘制模式至少部分地基于UI组件的该部分的分数以及与UI组件的该部分重叠的现有UI的一个或多个部分的分数而被调整。

[0138] 条款16.根据条款12-15所述的可折叠计算设备,其中现有UI的各部分是基于历史上在现有UI上执行的用户交互事件的集群而被标识的。

[0139] 条款17.根据条款12-16所述的可折叠计算设备,其中现有UI由应用窗口生成,并且其中现有UI基于应用窗口的用户界面元素而被划分为各部分。

[0140] 条款18.根据条款12-17所述的可折叠计算设备,其中遮挡现有UI降低与UI组件相关联的分数。

[0141] 条款19.一种在其上存储有计算机可执行指令的非瞬态计算机可读存储介质,该计算机可执行指令在由可折叠计算设备102执行时,使可折叠计算设备102:接收与应用窗口404相关联的应用被调用的指示;确定应用窗口404的预期位置;计算与和应用窗口404的预期位置的至少一部分重叠的用户界面 (UI) 组件402相关联的分数;计算与在应用窗口404的一部分被显示时将与UI组件402重叠的应用窗口404的该部分相关联的分数;在与UI组件402相关联的分数和与应用窗口404的该部分相关联的分数之间执行比较;至少部分地基于该比较为UI组件402选择绘制模式;并使用选定的绘制模式显示UI组件402。

[0142] 条款20.根据条款19所述的非瞬态性计算机可读存储介质,其中与应用窗口的该部分相关联的分数至少部分地基于用户与应用窗口的该部分的交互的历史频率。

[0143] 基于上述内容,应当理解,在本文中已经公开了用于基于内容遮挡和用户意图的预测来选择UI组件绘制模式的技术。尽管本文提出的主题已经以特定于计算机结构特征、方法和变革行为、特定计算机器和计算机可读介质的语言进行了描述,但是应理解,所附权利要求中阐述的主题不一定局限于本文描述的特定特征、行为或介质。相反,特定特征、动作和介质作为实现要求保护的主题的示例形式而被公开。

[0144] 上述主题仅通过说明的方式来提供并且不应被解释为限制性的。在不遵循图示和描述的示例配置和应用的情况下,并且在不背离在所附权利要求中阐述的本公开的范围的情况下,可以对本文描述的主题进行各种修改和改变。

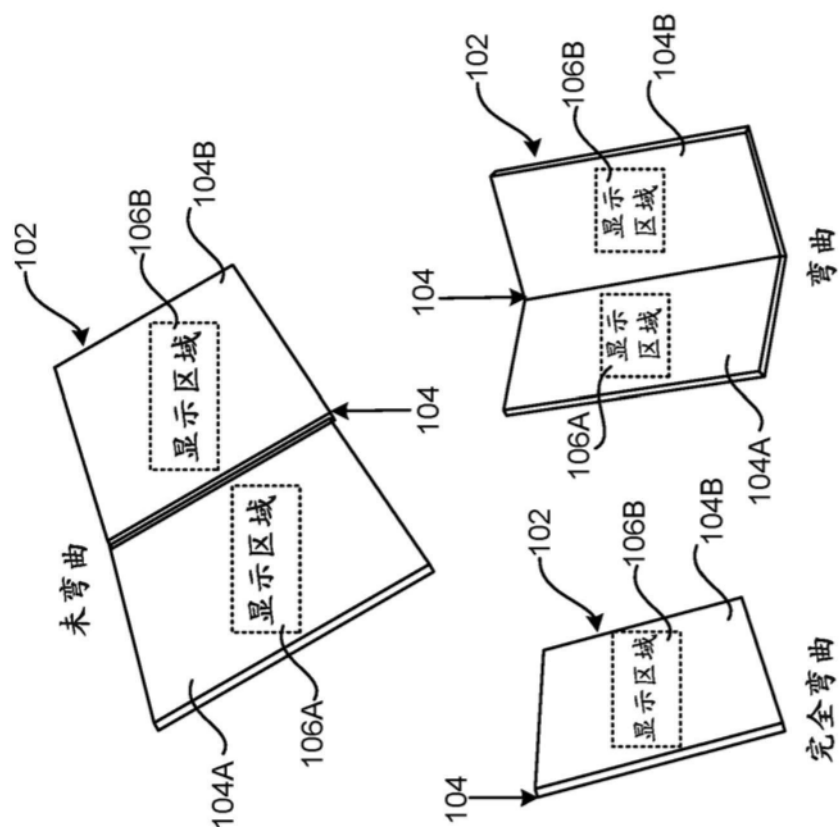


图1A

纵向姿势

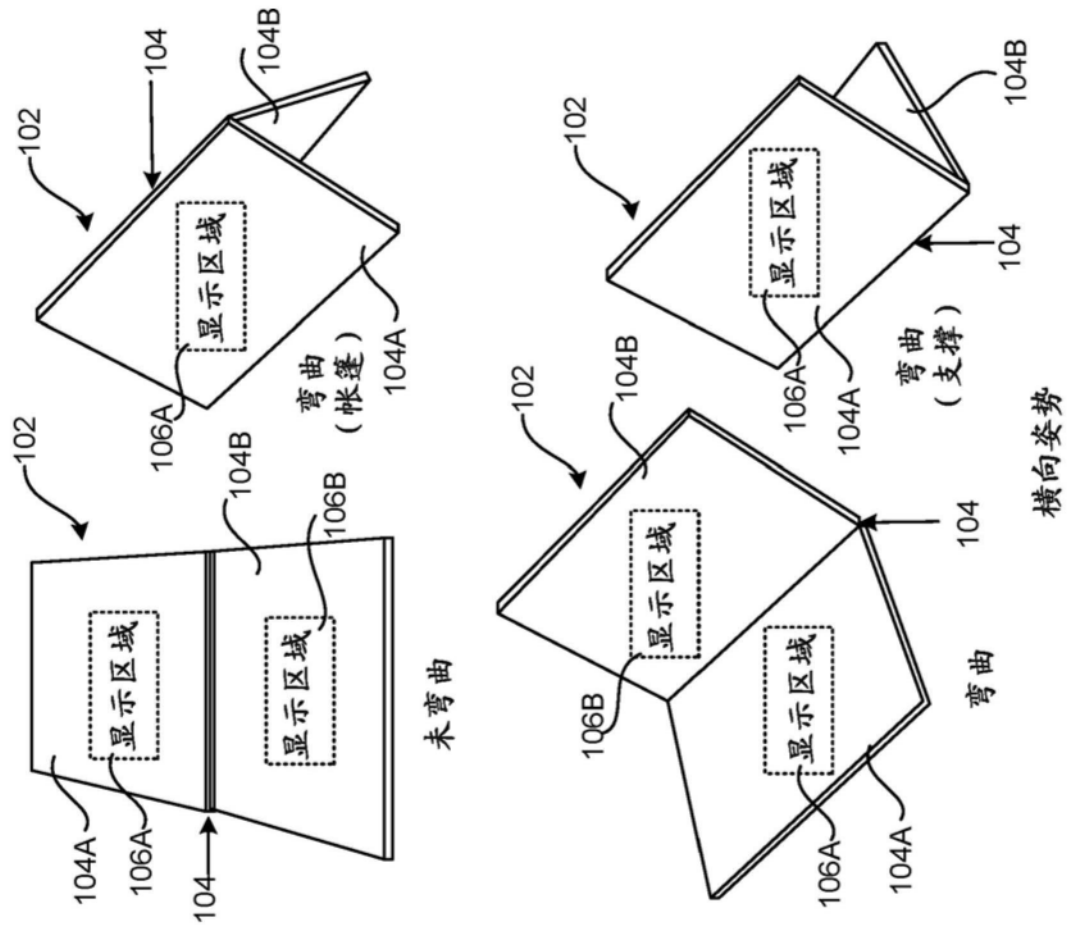


图1B

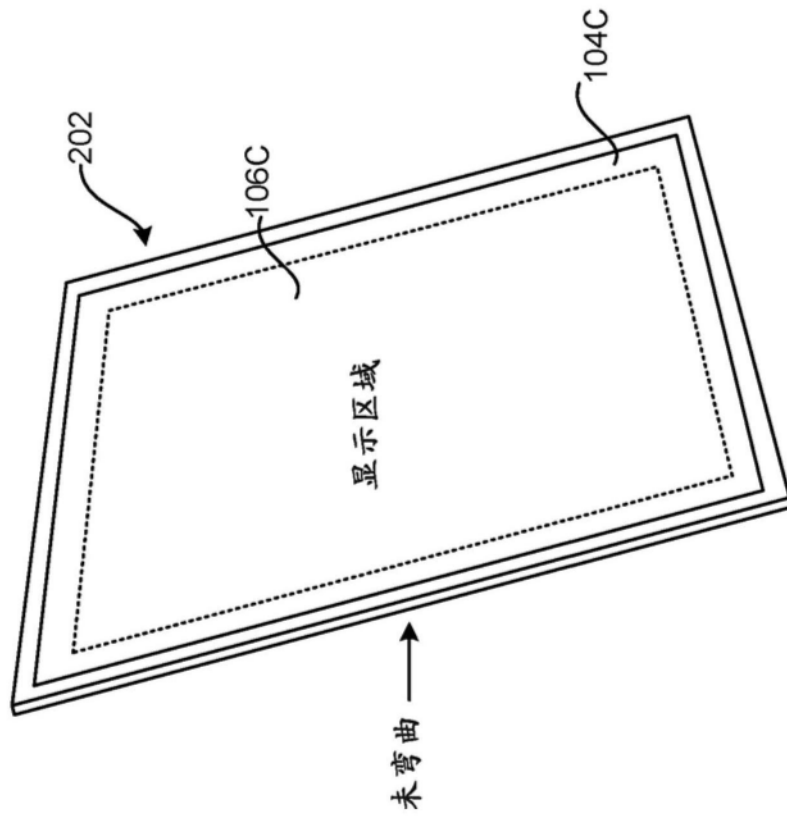


图2A

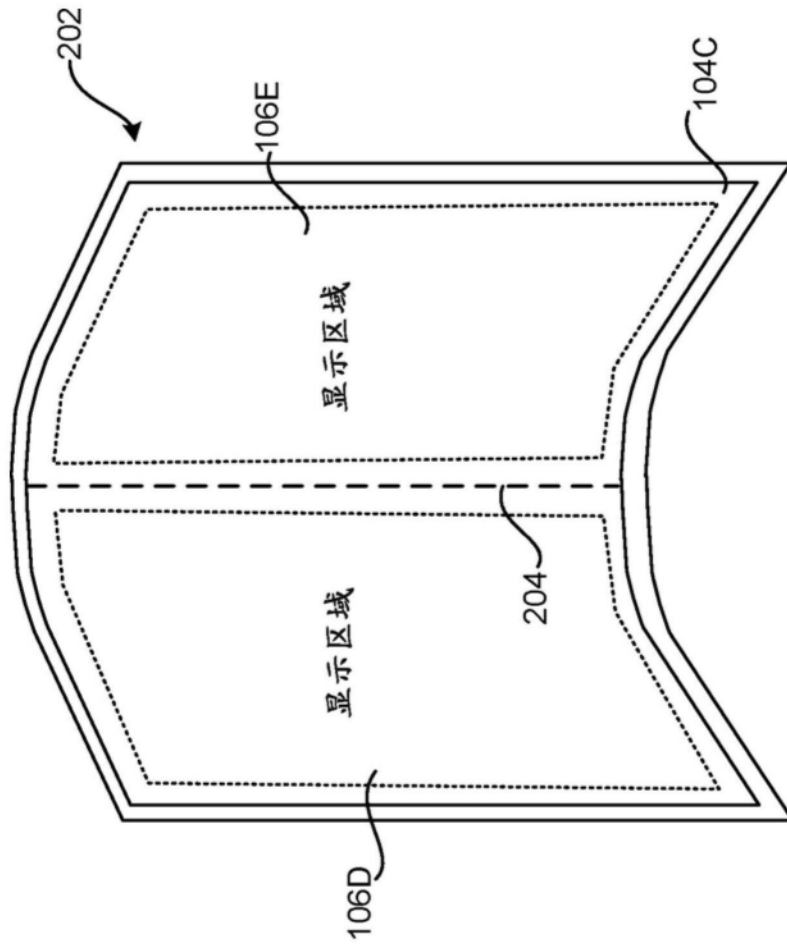


图2B

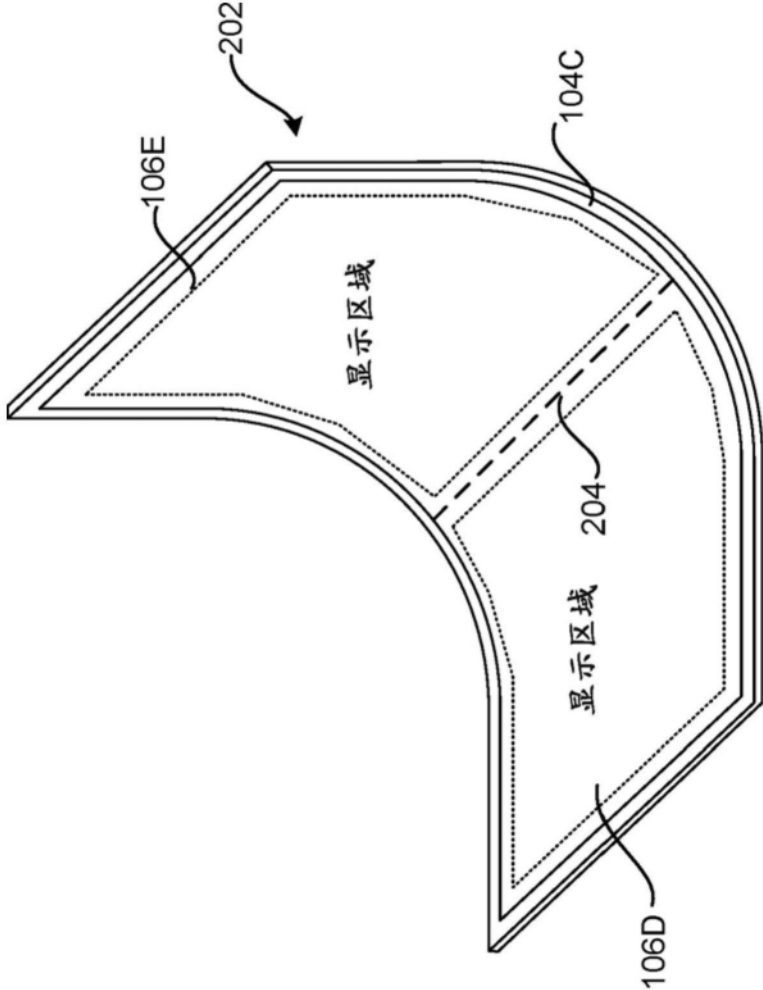


图2C

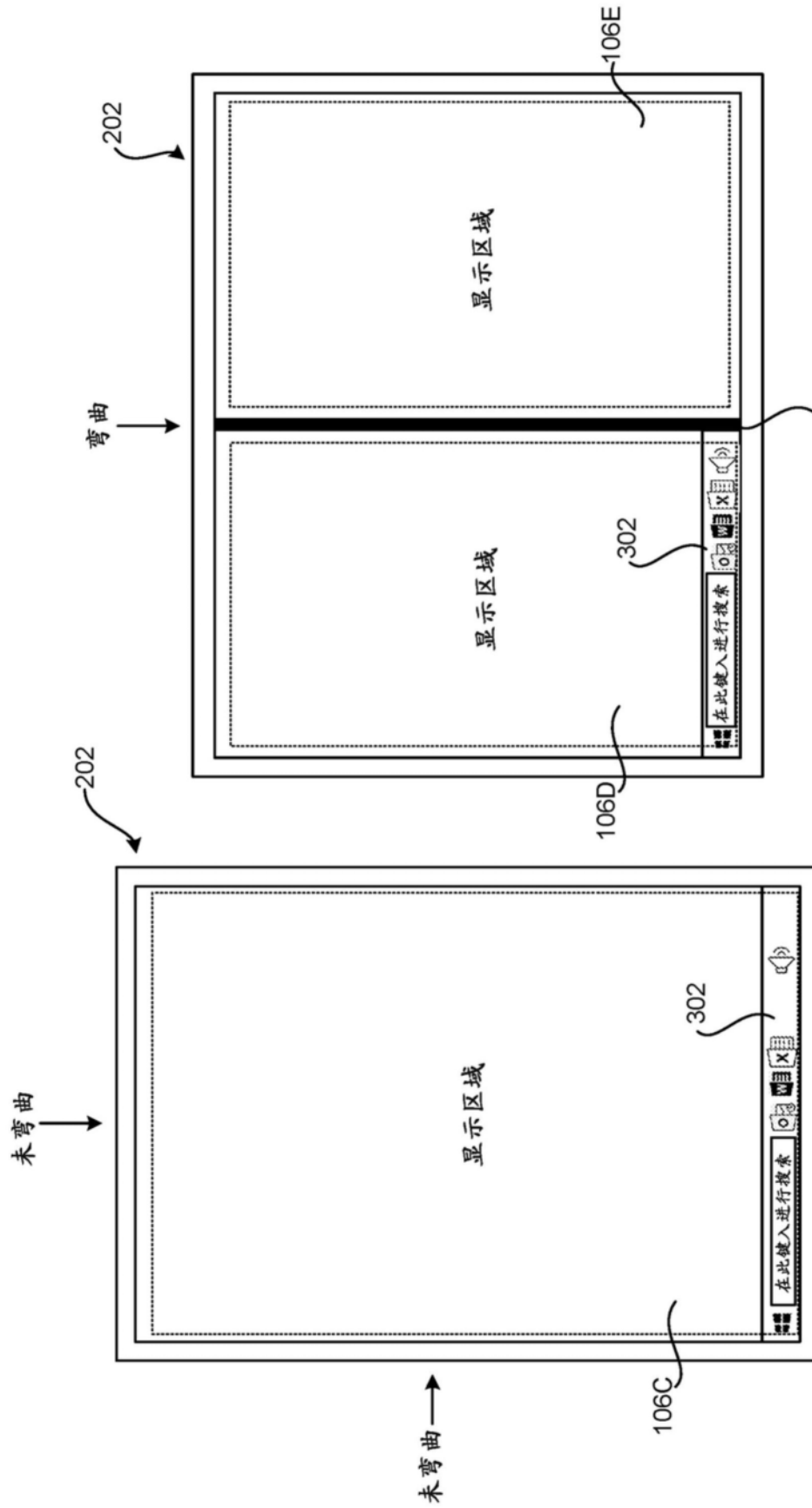


图 3B

图 3A

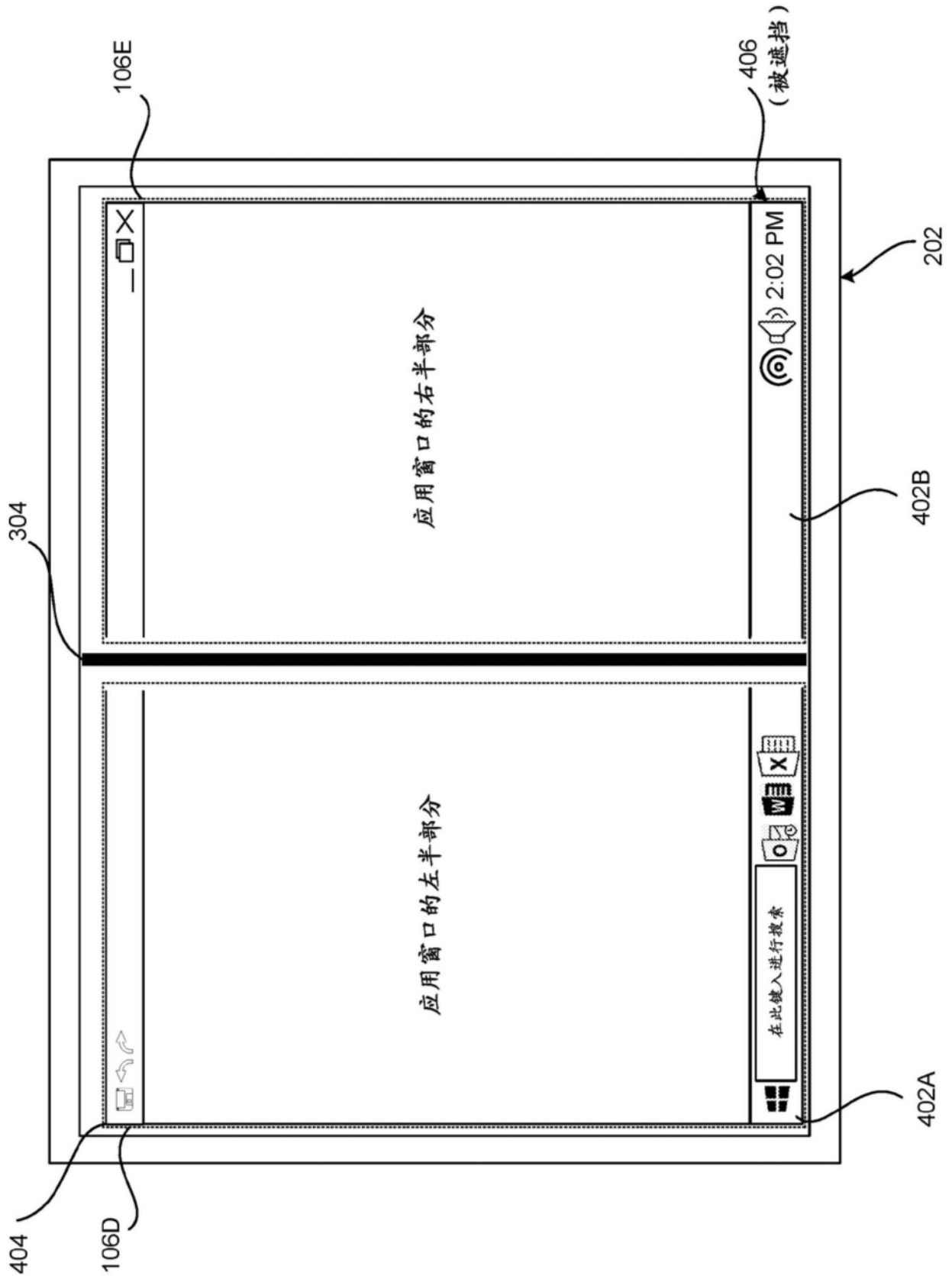


图4

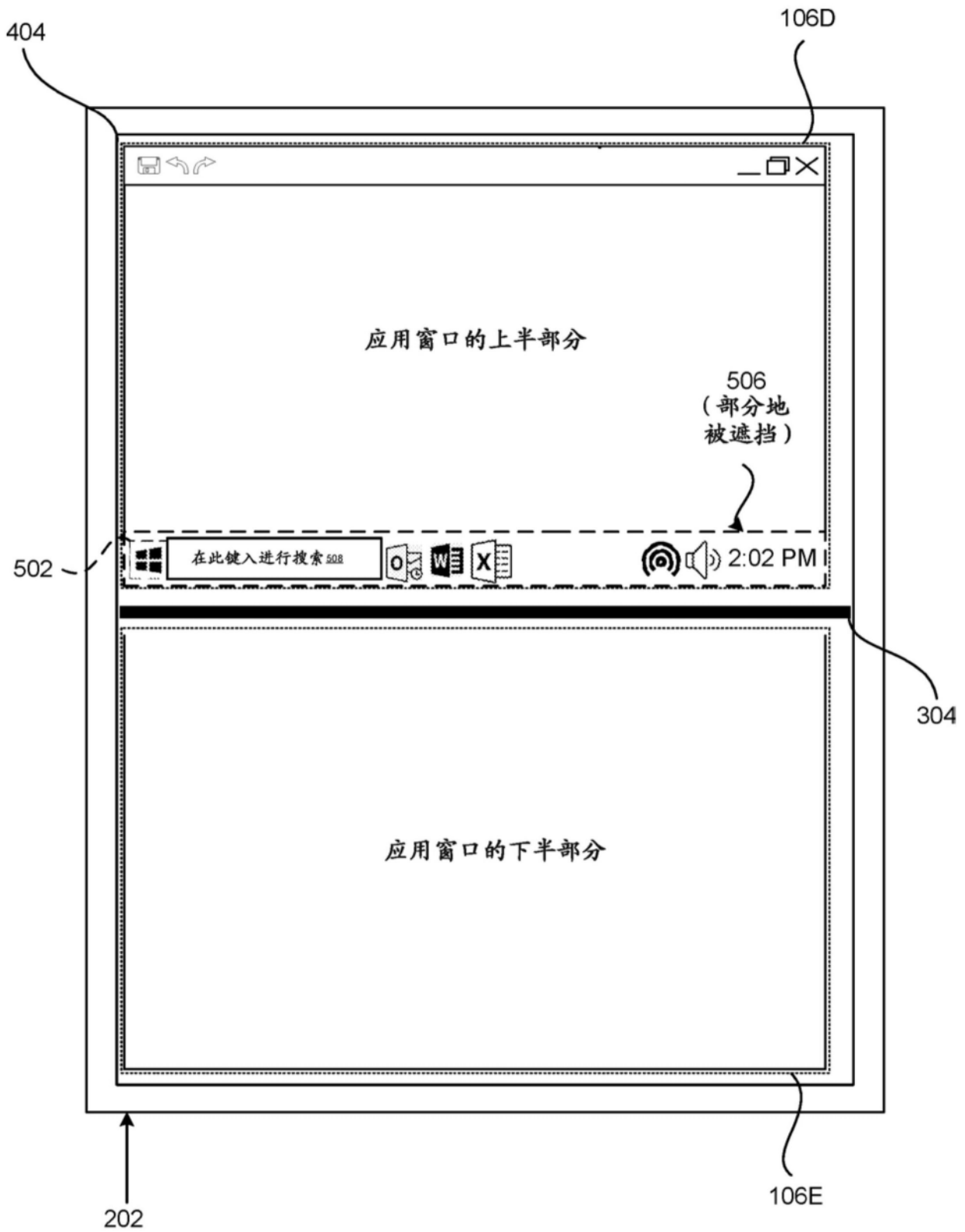


图5

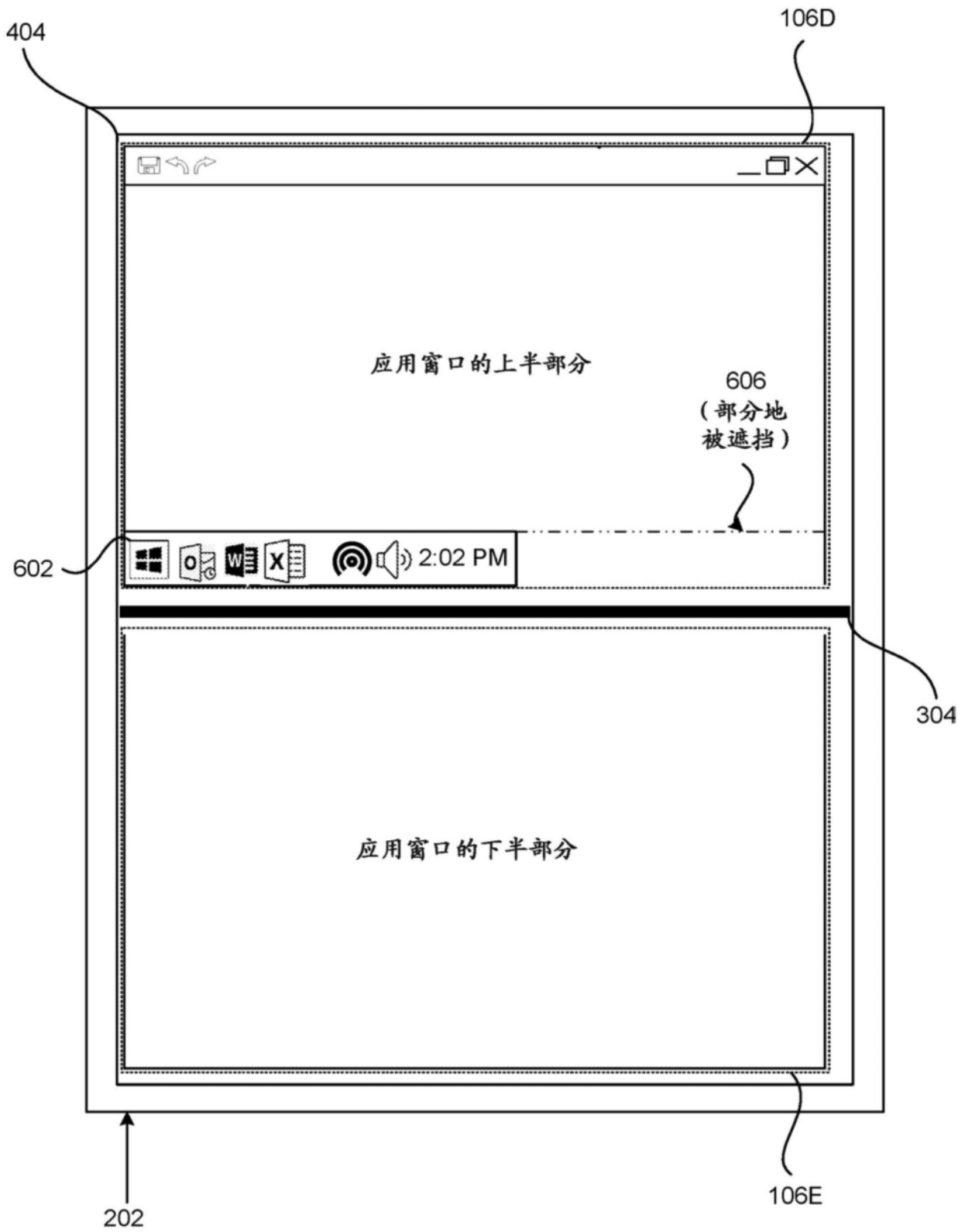


图6

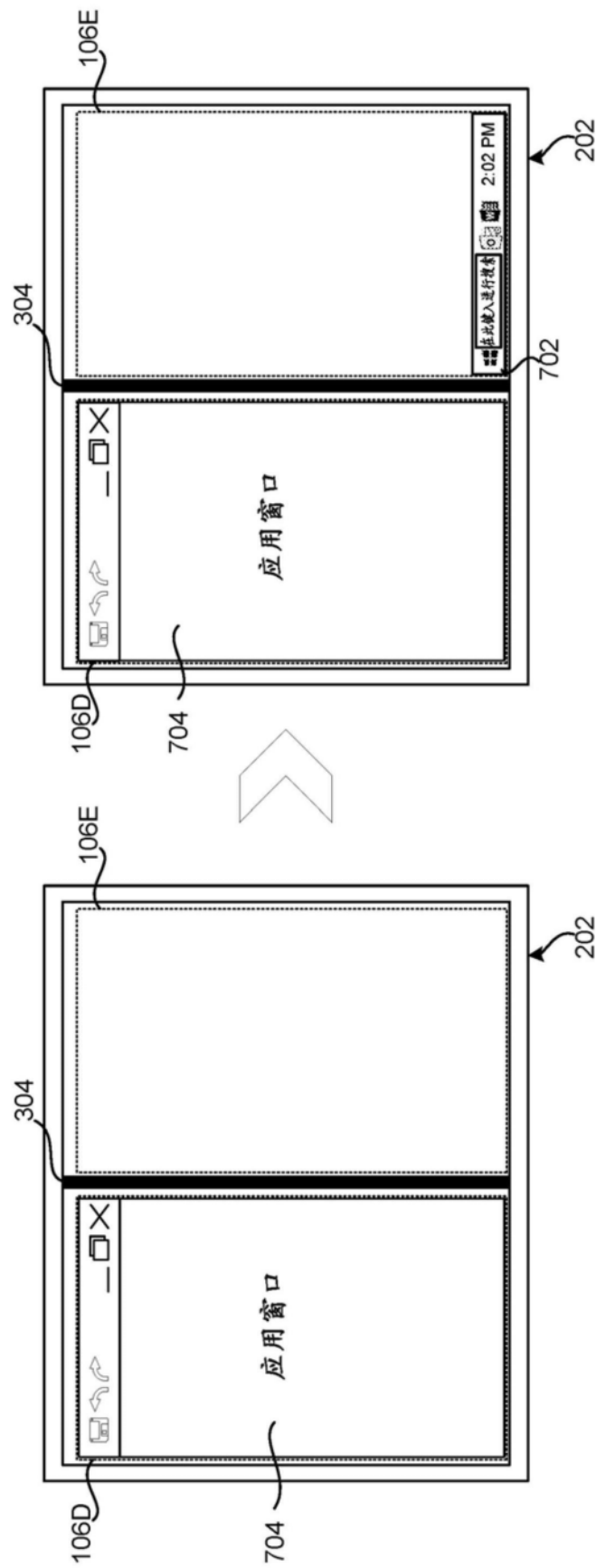


图7

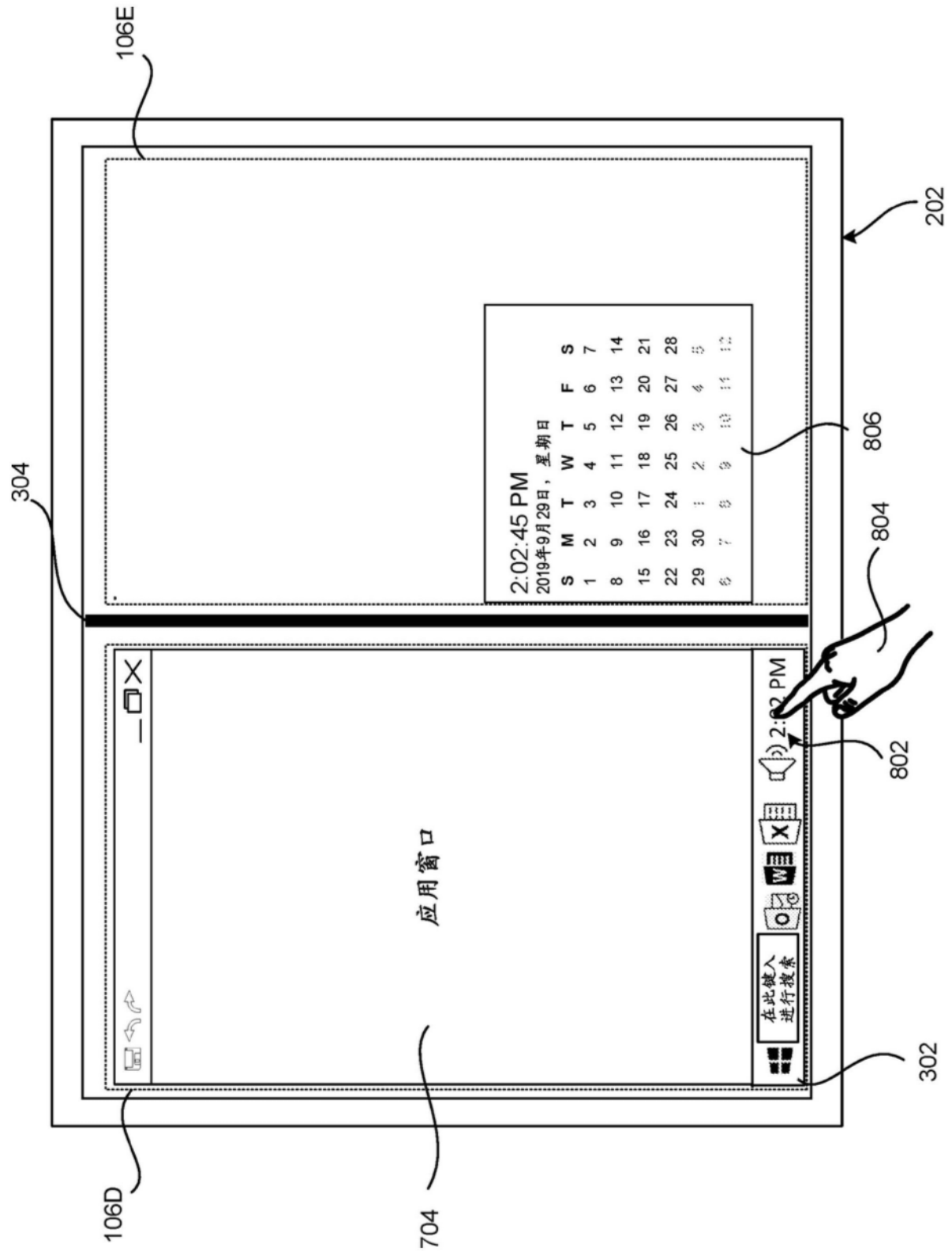


图8

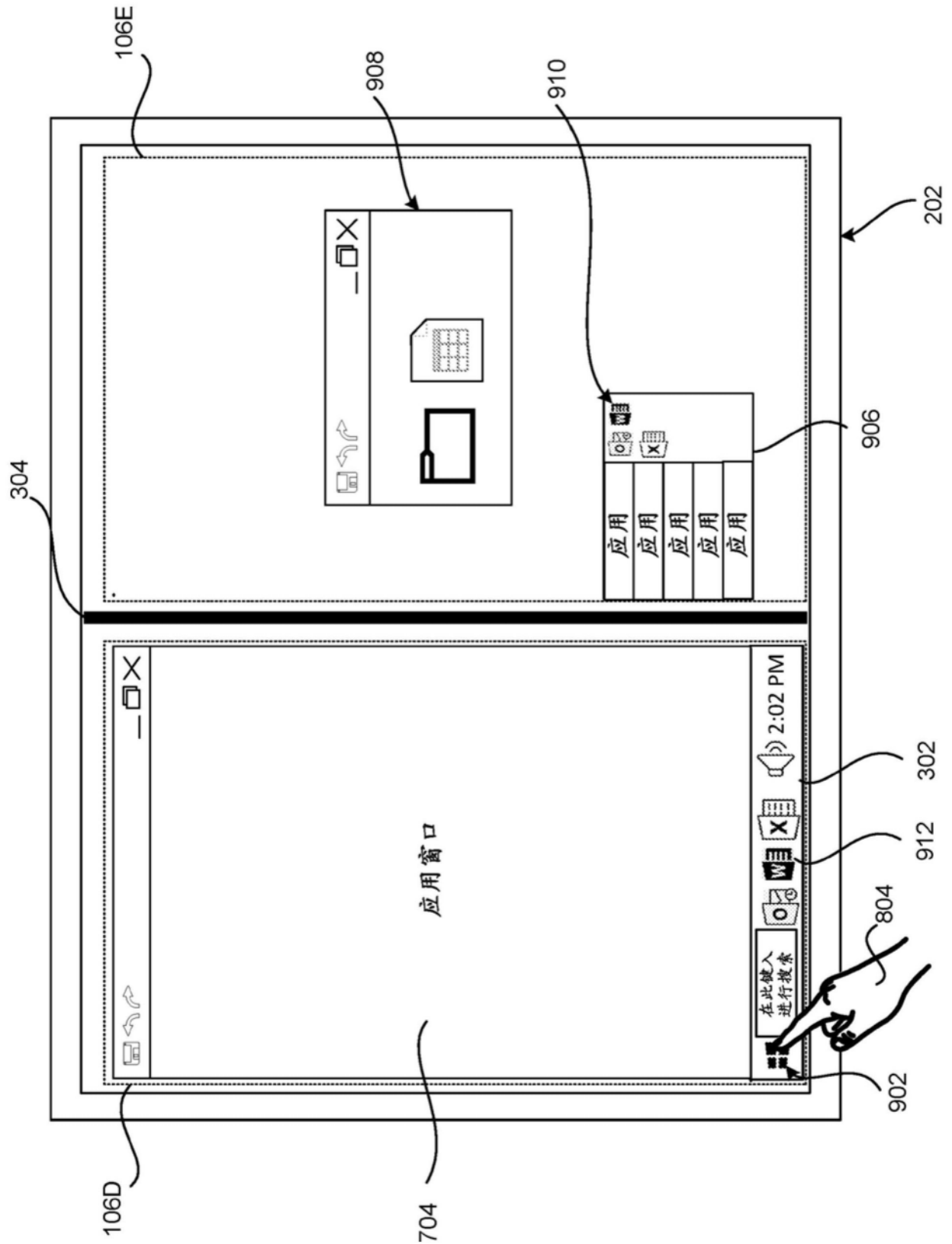


图9

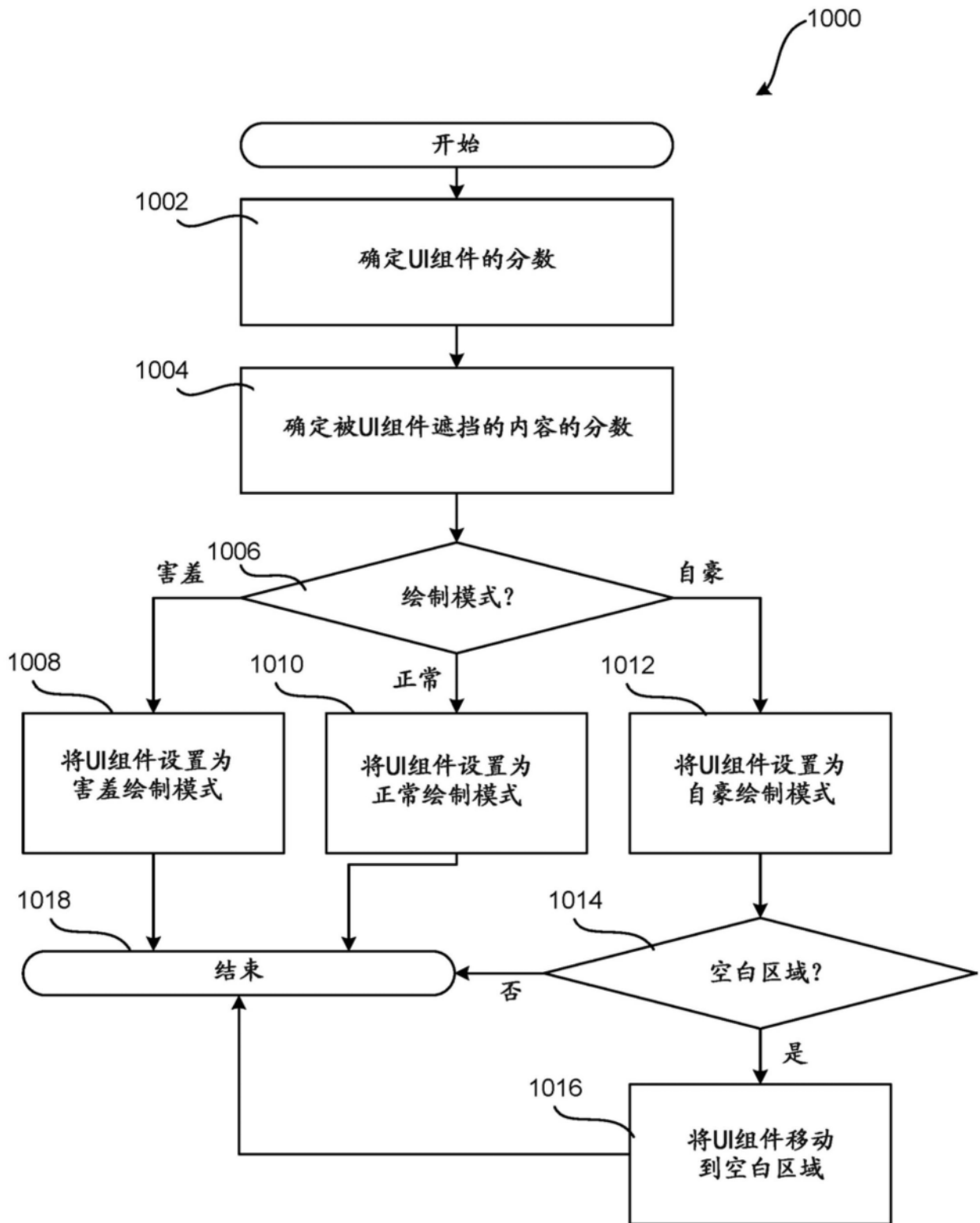


图10

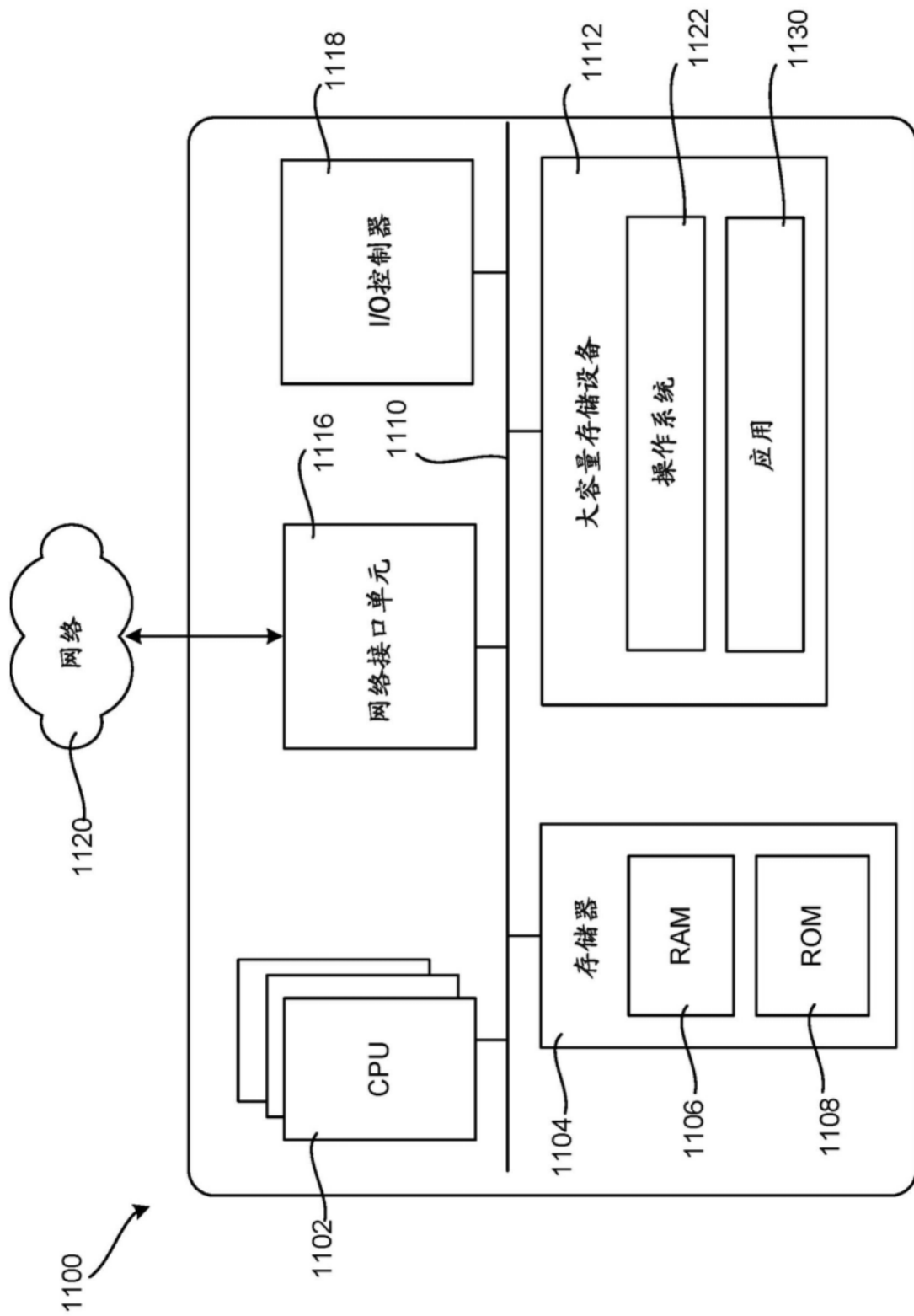


图11