



(12) 发明专利

(10) 授权公告号 CN 102782747 B

(45) 授权公告日 2016.02.10

(21) 申请号 201080057011.3

(22) 申请日 2010.12.10

(30) 优先权数据

12/638,941 2009.12.15 US

(85) PCT国际申请进入国家阶段日

2012. 06. 15

(86) PCT国际申请的申请数据

PCT/CA2010/001966 2010. 12. 10

(87) PCT国际申请的公布数据

W02011/072379 EN 2011. 06. 23

(73) 专利权人 ATI 科技无限责任公司

地址 加拿大安大略省

(72) 发明人 伊南·蒋 沙赫里亚·佩泽什基

钱明伟

(74) 专利代理机构 上海胜康律师事务所 31263

代理人 李献忠

(51) Int. Cl.

G09G 5/00(2006.01)

G06F 3/14(2006.01)

G09G 5/393(2006.01)

G09G 5/397(2006.01)

(56) 对比文件

US 7477205 B1, 2009.01.13.

CN 1886779 A, 2006. 12. 27.

审查员 高倩倩

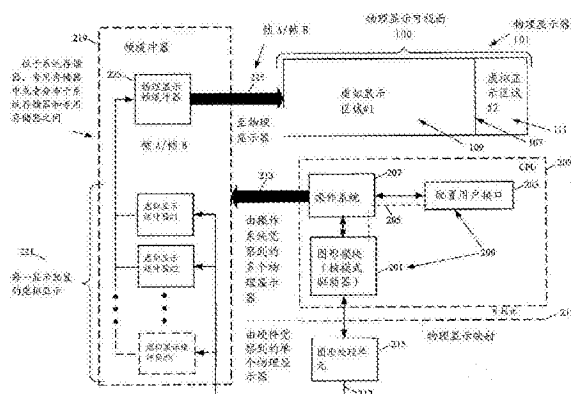
权利要求书4页 说明书10页 附图15页

(54) 发明名称

用于将显示面分割成多个虚拟显示区域的装置和方法

(57) 摘要

将单一显示器的可视区域分割成至少两个虚拟可视区域并且通过操作系统将至少两个虚拟可视区域仿真为至少两个仿真物理显示以使操作系统表现得好像与至少两个实际的独立物理显示器交互一样的方法和装置。所述方法为操作系统提供响应于来自操作系统的查询而生成的每个仿真物理显示的显示标识数据(诸如“EDID”)。所述方法和装置还接收中断通知(其中中断对应于单一物理显示器),并且将与至少两个仿真物理显示对应的至少两组中断报告信息报告给操作系统,好像接收到两个中断一样。操作系统因此“假装”好像两个物理显示器正在运行中一样动作。



1. 一种用于将显示面分割成多个虚拟显示区域的方法,所述方法包括:
将单一显示器的可视区域分割为被限定的至少两个虚拟可视区域;以及
通过将至少两个仿真物理显示中的每一个有关的信息提供给操作系统将所述至少两个虚拟可视区域仿真为所述至少两个仿真物理显示,使得所述操作系统表现得好像与两个实际的物理显示器交互一样。
2. 如权利要求 1 中所述的方法,其中,将所述至少两个虚拟可视区域仿真为至少两个仿真物理显示包括:
将响应于来自所述操作系统对显示信息的查询而生成的所述至少两个仿真物理显示中的每个的显示标识数据提供给所述操作系统。
3. 如权利要求 1 中所述的方法,其中,将所述至少两个虚拟可视区域仿真为至少两个仿真物理显示包括:
接收来自图形处理单元的中断通知,其中所述中断对应于所述单一显示器;以及
将与所述至少两个仿真物理显示对应的至少两组中断报告信息报告给所述操作系统,好像接收到两个中断一样。
4. 如权利要求 1 中所述的方法,包括:
在所述至少两个虚拟可视区域的一个虚拟可视区域内以全屏模式显示第一应用程序窗口,其中处于所述全屏模式的所述应用程序窗口仅在所述一个虚拟可视区域的周边内扩展。
5. 如权利要求 4 所述的方法,包括:
在所述至少两个虚拟可视区域的另一虚拟可视区域内显示第二应用程序窗口,而在所述一个虚拟可视区域内以所述全屏模式显示所述第一应用程序窗口;以及
在所述第二应用程序窗口成为关注焦点的同时继续以所述全屏模式显示所述第一应用程序窗口。
6. 如权利要求 1 所述的方法,包括:
合成显示数据帧,其中每个数据帧包括来自所述至少两个虚拟可视区域中的两个的数据;以及
将所述显示数据帧发送给所述单一显示器。
7. 如权利要求 6 所述的方法,包括:
确定在所述至少两个虚拟可视区域的第一虚拟可视区域内显示的显示数据已经变化;以及
将来自所述至少两个虚拟可视区域中的第二虚拟可视区域的显示数据复制到第一显示数据帧中。
8. 如权利要求 1 所述的方法,包括:
接收与将所述单一显示器的可视区域分割为所述至少两个虚拟可视区域的显示配置对应的选择输入;以及
其中,将单一显示器的可视区域分割为至少两个虚拟可视区域是对所述选择输入的响应。
9. 如权利要求 1 所述的方法,其中,将所述至少两个虚拟可视区域仿真为至少两个仿真物理显示包括:

接收来自所述操作系统的缓冲数据已准备好在所述至少两个仿真物理显示中的一个上显示的通知；

确定所述至少两个仿真物理显示中的第一仿真物理显示上的显示数据已经变化；

将所述至少两个仿真物理显示中的第二仿真物理显示的显示数据复制到第一数据帧中,其中所述第二仿真物理显示的显示数据不变,并且其中所述第二仿真物理显示的所述显示数据不存在于所述第一数据帧中,并且对于所述第一和第二仿真物理显示在所述第一数据帧中将标记设定为完成;以及

对于所述第一仿真物理显示在第二数据帧中将标记设定为未完成。

10. 如权利要求 1 所述的方法,其中,将所述至少两个虚拟可视区域仿真为至少两个仿真物理显示包括:

接收来自所述操作系统的缓冲数据已准备好在所述至少两个仿真物理显示中的一个上显示的通知;

确定所述至少两个仿真物理显示中的第一仿真物理显示上的显示数据未变化并且不存在于第二数据帧中;

复制所述第一仿真物理显示以及第二仿真物理显示这两者的显示数据,其中所述第二仿真物理显示的显示数据已变化,将所述第一和第二仿真物理显示的所述显示数据复制到所述第二数据帧中,并且对于所述第一和第二仿真物理显示在所述第二数据帧中将标记设定为完成;以及

对于所述第二仿真物理显示,在第一数据帧中将标记设定为未完成。

11. 如权利要求 1 所述的方法,其中,将所述至少两个虚拟可视区域仿真为至少两个仿真物理显示包括:

接收来自所述操作系统的缓冲数据已准备好在所述至少两个仿真物理显示中的一个上显示的通知;

确定所述至少两个仿真物理显示中的第一仿真物理显示的显示数据已经变化,并且来自第二仿真物理显示的第二显示数据未变化并且存在于第一数据帧中;

将所述第一仿真物理显示的显示数据复制到所述第一数据帧中,并且对于所述第一仿真物理显示在所述第一数据帧中将标记设定为完成;以及

对于所述第一仿真物理显示在第二数据帧中将标记设定为未完成。

12. 如权利要求 1 所述的方法,其中,将所述至少两个虚拟可视区域仿真为至少两个仿真物理显示包括:

接收与光标位置对应的中断;

确定所述至少两个仿真物理显示中的所述光标所在的仿真物理显示;以及

将相对于所述光标所在的所述仿真物理显示的所述光标的相对位置转换为与所述单一显示器的总显示面对应的物理位置。

13. 一种用于将显示面分割成多个虚拟显示区域的装置,所述装置包括:

用于将单一显示器的可视区域分割为被限定的至少两个虚拟可视区域的装置;以及

用于通过将与至少两个仿真物理显示中的每一个有关的信息提供给操作系统将所述至少两个虚拟可视区域仿真为所述至少两个仿真物理显示,以使所述操作系统表现得好像与两个实际的物理显示器交互一样的装置。

14. 如权利要求 13 所述的装置, 其中, 用于将所述至少两个虚拟可视区域仿真为至少两个仿真物理显示的装置包括:

用于将响应于来自所述操作系统对显示信息的查询而生成的所述至少两个仿真物理显示中的每个的显示标识数据提供给所述操作系统的装置。

15. 如权利要求 13 所述的装置, 其中, 用于将所述至少两个虚拟可视区域仿真为至少两个仿真物理显示的装置包括:

用于接收来自图形处理单元的中断通知的装置, 其中所述中断对应于所述单一显示器; 以及

用于将与所述至少两个仿真物理显示对应的至少两组中断报告信息报告给所述操作系统, 好像接收到两个中断一样的装置。

16. 如权利要求 13 所述的装置, 进一步包括:

用于在所述至少两个虚拟可视区域的一个虚拟可视区域内以全屏模式显示第一应用程序窗口的装置, 其中处于所述全屏模式的所述应用程序窗口仅在所述一个虚拟可视区域的周边内扩展。

17. 如权利要求 16 所述的装置, 包括:

用于在所述至少两个虚拟可视区域的另一虚拟可视区域内显示第二应用程序窗口, 而在所述一个虚拟可视区域内以所述全屏模式显示所述第一应用程序窗口的装置; 以及

用于在所述第二应用程序窗口成为关注焦点的同时继续以所述全屏模式显示所述第一应用程序窗口的装置。

18. 如权利要求 13 所述的装置, 包括:

用于合成显示数据帧的装置, 其中每个数据帧包括来自所述至少两个虚拟可视区域中的两个的数据; 以及

用于将所述显示数据帧发送给所述单一显示器的装置。

19. 如权利要求 18 所述的装置, 包括:

用于确定在所述至少两个虚拟可视区域的第一虚拟可视区域内显示的显示数据已经变化的装置; 以及

用于将来自所述至少两个虚拟可视区域中的第二虚拟可视区域的显示数据复制到第一显示数据帧中的装置。

20. 如权利要求 13 所述的装置, 包括:

用于接收与将所述单一显示器的可视区域分割为所述至少两个虚拟可视区域的显示配置对应的选择输入的装置; 以及

其中, 将单一显示器的可视区域分割为至少两个虚拟可视区域是对所述选择输入的响应。

21. 如权利要求 13 所述的装置, 其中, 用于将所述至少两个虚拟可视区域仿真为至少两个仿真物理显示的装置包括:

用于接收来自所述操作系统的缓冲数据已准备好在所述至少两个仿真物理显示中的一个上显示的通知的装置;

用于确定所述至少两个仿真物理显示中的第一仿真物理显示上的显示数据已经变化的装置;

用于将所述至少两个仿真物理显示中的第二仿真物理显示的显示数据复制到第一数据帧中的装置,其中所述第二仿真物理显示的显示数据不变,并且其中所述第二仿真物理显示的所述显示数据不存在于所述第一数据帧中,并且对于所述第二仿真物理显示在所述第一和第二数据帧中将标记设定为完成;以及

用于对于所述第一仿真物理显示在第二数据帧中将标记设定为未完成的装置。

22. 如权利要求 13 所述的装置,其中,用于将所述至少两个虚拟可视区域仿真为至少两个仿真物理显示的装置包括:

用于接收来自所述操作系统的缓冲数据已准备好在所述至少两个仿真物理显示中的一个上显示的通知的装置;

用于确定所述至少两个仿真物理显示中的第一仿真物理显示上的显示数据未变化并且不存在于第二数据帧中的装置;

用于复制所述第一仿真物理显示以及第二仿真物理显示这两者的显示数据的装置,其中所述第二仿真物理显示的显示数据已变化,还包括用于将所述第一和第二仿真物理显示的所述显示数据复制到所述第二数据帧中,并且对于所述第一和第二仿真物理显示在所述第二数据帧中将标记设定为完成的装置;以及

用于对于所述第二仿真物理显示在第一数据帧中将标记设定为未完成的装置。

23. 如权利要求 13 所述的装置,其中,用于将所述至少两个虚拟可视区域仿真为至少两个仿真物理显示的装置包括:

用于接收来自所述操作系统的缓冲数据已准备好在所述至少两个仿真物理显示中的一个上显示的通知的装置;

用于确定所述至少两个仿真物理显示中的第一仿真物理显示上的显示数据已经变化,并且来自第二仿真物理显示的第二显示数据未变化并且存在于第一数据帧中的装置;

用于将所述第一仿真物理显示的显示数据复制到所述第一数据帧中,并且对于所述第一仿真物理显示在所述第一数据帧中将标记设定为完成的装置;以及

用于对于所述第一仿真物理显示在第二数据帧中将标记设定为未完成的装置。

24. 如权利要求 13 所述的装置,其中,用于将所述至少两个虚拟可视区域仿真为至少两个仿真物理显示的装置包括:

用于接收与光标位置对应的中断的装置;

用于确定所述至少两个仿真物理显示中的所述光标所在的仿真物理显示的装置;以及

用于将相对于所述光标所在的所述仿真物理显示的所述光标的相对位置转换为与所述单一显示器的总显示面对应的物理位置的装置。

用于将显示面分割成多个虚拟显示区域的装置和方法

技术领域

[0001] 本公开涉及显示并且涉及将单一物理显示器的显示面分割成多个虚拟显示。

背景技术

[0002] 为了监控既定的应用程序,具有将显示面拆分成多个部分的能力是方便的。例如,用户可能希望使应用程序在显示的一部分中运行,并且使因特网聊天应用程序在另一分区中运行。通过这种方式,用户可以在与一个或多个游戏伙伴或其它朋友运行聊天对话的同时参与在线游戏应用程序。在另一实施例中,用户可能希望在继续监控诸如股票报价、新闻等其它项的同时在一个分区中运行主要的应用程序。

[0003] 存在允许用户创建分区的一些实用程序。不幸的是,应用程序设置优先于分区并且使得这些现有的实用程序不能适应一些使用情况的方案。例如,当应用程序在“全屏模式”运行时,即在正常情况下应用程序窗口遍及实际显示面的整个区域运行时,应用程序将遮住任何设定的分区,从而不再能看到任何其它的应用程序。换句话说,诸如全屏模式的应用程序设置优先于由现有的分区实用程序创建的任何显示分区设置。

[0004] 因此,对于这样的装置和方法存在需求:所述装置和方法能够在不受应用程序干涉的情况下对显示面进行分区,使得在全屏模式下运行的应用程序将保留在设定分区内并且不妨碍在其它分区中显示的其它应用程序。

附图说明

[0005] 图 1 为根据实施方案的装置的框图。

[0006] 图 2 为根据实施方案的具有两个虚拟显示分区的显示面的图。

[0007] 图 3 为根据实施方案的具有两个虚拟显示分区的显示面的图,其中一个应用程序在不妨碍另一虚拟显示分区的情况下在全屏模式下运行。

[0008] 图 4 为根据实施方案的具有两个虚拟显示分区的显示面的图,其中使一个应用程序窗口移动而横过虚拟显示边界。

[0009] 图 5 为根据实施方案的具有两个虚拟显示分区的显示面的图,其中使光标移动而横过虚拟显示边界。

[0010] 图 6 为根据实施方案的具有两个虚拟显示分区的显示面的图,其中第二显示提供了虚拟显示分区中的一个的代用件。

[0011] 图 7 为根据实施方案以呈现出图像撕裂的方式图示出帧成分的流程图。

[0012] 图 8 为根据实施方案图示出虚拟显示的硬件中断处理和硬件中断模拟的流程图。

[0013] 图 9 为根据实施方案图示出硬件终端处理和虚拟显示更新的流程图。

[0014] 图 10 为根据实施方案图示出在光标在各个虚拟显示分区之间移动时光标的处理的流程图。

[0015] 图 11 为图示出各个实施方案的高级操作的流程图。

[0016] 图 12 为图示出各个实施方案的高级操作的流程图。

[0017] 图 13 为图示出当应用程序在全屏模式下运行时各个实施方案的高级操作的流程图。

[0018] 图 14 为图示出根据各个实施方案的具有多个虚拟分区的拆分显示的启动的流程图。

[0019] 图 15 为图示出根据各个实施方案的拆分显示的关闭的流程图。

[0020] 图 16 为根据实施方案的拆分显示配置应用程序窗口的实施例。

[0021] 图 17 为根据实施方案可设置在拆分显示配置应用程序窗口中的示例性的显示配置的图。

具体实施方式

[0022] 本公开提供了一种方法,包括:将单一显示器的可视区域分割成至少两个虚拟可视区域;以及通过操作系统将至少两个虚拟可视区域仿真为至少两个仿真物理显示,使得操作系统表现得好像与两个实际的物理显示器交互一样。所述方法实现了以各种方式对至少两个仿真物理显示的仿真。例如,所述方法为操作系统提供响应于来自操作系统对于显示信息的查询而生成的每个仿真物理显示的显示标识数据(诸如 EDID,“扩展显示标识数据”)。

[0023] 在另一实施例中,该方法通过如下操作对至少两个仿真物理显示进行仿真:接收来自图形处理单元的中断通知(其中中断对应于单一物理显示器);以及将与至少两个仿真物理显示相对应的至少两组中断报告信息报告给操作系统,好像接收到两个中断一样。操作系统因此“假装”动作,好像是两个物理显示器正在操作一样。

[0024] 该方法进一步提供:在至少两个虚拟可视区域中的被仿真为物理显示的一个虚拟可视区域内在全屏模式下显示第一应用程序窗口,其中全屏模式下的应用程序仅在一个虚拟可视区域的周边内扩展。该方法可进一步包括:在至少两个虚拟可视区域中的被仿真为另一物理显示的另一虚拟可视区域内显示第二应用程序窗口,而在一个虚拟可视区域内在全屏模式下显示第一应用程序;以及继续在全屏模式下显示第一应用程序而第二应用程序窗口被访问,例如集中关注第二应用程序窗口。

[0025] 该方法包括用于编写显示数据帧的双缓冲方法,其中每个数据帧包括来自至少两个虚拟可视区域中的两个虚拟可视区域的数据。该方法包括:将显示数据帧发送到单一显示器。该方法还可以包括:确定在至少两个虚拟可视区域的第一虚拟可视区域内显示的显示数据已经改变;以及将来自至少两个虚拟可视区域的第二虚拟可视区域的显示数据复制到第一显示数据帧中。

[0026] 该方法还可以包括:接收与将单一显示器的可视区域分割成至少两个虚拟可视区域的显示配置相对应的选择输入。在该实施方案中,该方法响应于所述选择输入而将单一显示器的可视区域分割成至少两个虚拟可视区域。

[0027] 在一些实施方案中,该方法还可以通过如下操作将至少两个虚拟可视区域仿真为至少两个仿真物理显示:接收来自操作系统的缓冲数据已准备好在至少两个仿真物理显示中的一个上显示的通知;确定至少两个仿真物理显示中的第一仿真物理显示上的显示数据已经变化;将显示数据未变化但是不在至少两个仿真物理显示中的第一数据帧中的第二仿真物理显示的显示数据复制到第一数据帧中,并且对于第一和第二仿真物理显示在第一数

据帧中将标记设定为完成 ; 以及对于第一仿真物理显示在第二数据帧中将标记设定为未完成。仿真的方法还可以包括 : 接收来自操作系统的缓冲数据已准备好在至少两个仿真物理显示中的一个上显示的通知 ; 确定至少两个仿真物理显示中的第二仿真物理显示上的显示数据已经变化 ; 将至少两个仿真物理显示中的显示数据未改变但是不在第二数据帧中的第一仿真物理显示以及第二仿真物理显示这两者的显示数据复制到第二数据帧中, 并且对于第一和第二仿真物理显示在第二数据帧中将标记设定为完成 ; 以及对于第二仿真物理显示在第一数据帧中将标记设定为未完成。

[0028] 该方法还可以包括光标控制 : 例如, 接收与光标位置对应的通知 ; 确定至少两个仿真物理显示中的光标所在的仿真物理显示 ; 以及将相对于光标所在的仿真物理显示的光标相对位置转换为与单一显示器的总显示面对应的物理位置。

[0029] 所公开的实施方案还提供了一种具有至少一个可编程处理器以及与可编程处理器可操作联接的存储器的装置。所述存储器收容由至少一个处理器执行的可执行指令。当至少一个可编程处理器执行可执行指令时, 可编程处理器可操作以将单一显示器的可视区域分割成至少两个虚拟可视区域并且通过操作系统将至少两个虚拟可视区域仿真为至少两个仿真物理显示, 使得操作系统表现为好像与两个实际的物理显示器交互 (interface) 一样。

[0030] 装置的至少一个可编程处理器在执行可执行指令时还可操作以通过将响应于来自操作系统对显示信息的查验而生成的至少两个仿真物理显示中的每个的显示标识数据提供给操作系统而将至少两个虚拟可视区域仿真为至少两个仿真物理显示。至少一个可编程处理器还可操作以通过接收来自装置的图形处理单元的中断通知并且将与至少两个仿真物理显示对应的至少两组中断报告信息报告给操作系统而仿真至少两个仿真物理显示, 其中所述中断对应于单一显示器, 好像是接收到两个中断一样。

[0031] 所述装置还可包括显示器, 所述显示器可操作以在至少两个虚拟可视区域的一个虚拟可视区域内以全屏模式显示第一应用程序窗口, 其中全屏模式下的应用程序窗口仅在一个虚拟可视区域的周边内扩展。所述显示器可进一步在至少两个虚拟可视区域的另一虚拟可视区域内显示第二应用程序窗口, 而在一个虚拟可视区域内以全屏模式显示第一应用程序。所述显示器随后继续以全屏模式显示第一应用程序窗口, 而第二应用程序窗口被访问, 例如集中关注第二应用程序窗口。

[0032] 所公开装置的至少一个可编程处理器也可编写显示数据帧, 其中每个数据帧包括来自至少两个虚拟可视区域中的两个的数据。所述显示数据帧被发送到单一显示器。至少一个可编程处理也可以 : 确定在至少两个虚拟可视区域的第一虚拟可视区域内显示的显示数据已经变化 ; 以及将来自至少两个虚拟可视区域的第二虚拟可视区域的显示数据复制到第一显示数据帧中。

[0033] 在一些实施方案中, 还为拆分显示的配置以及选择如何布置虚拟显示区域提供了用户接口。因此, 装置的至少一个可编程处理器还可操作以接收将单一显示器的可视区域分割成至少两个虚拟可视区域的与显示配置对应的选择输入。单一显示器的可视区域随后响应于选择输入而被分割成至少两个虚拟可视区域。

[0034] 在一些实施方案中, 装置的至少一个可编程处理器可操作以通过如下操作将至少两个虚拟可视区域仿真为至少两个仿真物理显示 : 接收来自操作系统的缓冲数据已准备好

至少两个仿真物理显示中的一个上的显示的通知 ; 确定至少两个仿真物理显示中的第一仿真物理显示上的显示数据已经变化 ; 将至少两个仿真物理显示中的显示数据未改变的第二仿真物理显示的显示数据复制到第一数据帧中 , 并且对于第二仿真物理显示在第一数据帧中设定标记为完成 ; 以及对于第二仿真物理显示在第二数据帧中设定标记为未完成。

[0035] 可编程处理器还可操作以通过如下操作来仿真物理显示 : 接收来自操作系统的缓冲数据已准备好至少两个仿真物理显示中的一个上的显示的通知 ; 确定至少两个仿真物理显示中的第一仿真物理显示上的显示数据已经变化 ; 复制至少两个仿真物理显示中的显示数据未变化但是不在第一数据帧中的第一仿真物理显示和第二仿真物理显示的显示数据复制到第一数据帧中 , 并且对于第一和第二仿真物理显示在第一数据帧中将标记设定为完成 ; 以及对于第一仿真物理显示在第二数据帧中将标记设定为未完成。可编程处理器还可操作以通过如下操作来仿真物理显示 : 接收来自操作系统的缓冲数据已准备好至少两个仿真物理显示中的一个上的显示的通知 ; 确定至少两个仿真物理显示中的第二仿真物理显示上的显示数据已经变化 ; 将至少两个仿真物理显示中的显示数据已改变的第二仿真物理显示以及显示数据未改变但是不在第二数据帧中的第一仿真物理显示的显示数据复制到第二数据帧中 , 并且对于第一和第二仿真物理显示在第二数据帧中设定标记为完成 ; 以及对于第二仿真物理显示在第一数据帧中设定标记为未完成。

[0036] 至少一个可编程处理器还可操作以实现光标控制 , 包括 : 接收与光标位置对应的通知 ; 确定至少两个仿真物理显示中的光标所在的仿真物理显示 ; 以及将相对于光标所在的仿真物理显示的光标相对位置转换为与单一显示器的总显示面对应的物理位置。

[0037] 本公开进一步提供了计算机可读存储器 , 所述计算机可读存储器包含由至少一个处理器执行的可执行指令 , 当执行可执行指令时使得至少一个处理器执行依照上文所列的实施方案的操作和方法。例如 , 当执行可执行指令时 , 可执行指令可使得至少一个处理器将单一显示器的可视区域分割成至少两个虚拟可视区域 , 并且通过操作系统将至少两个虚拟可视区域仿真为至少两个仿真物理显示 , 使得操作系统表现得好像与两个实际物理显示器交互一样。

[0038] 计算机可读存储器可以为可用于将可执行指令或程序代码装载到诸如下面要进一步详细说明了其它处理器件或电子器件的任何适当的非易失性存储器 , 诸如但不限于诸如 EEPROM、快擦写 ROM (拇指驱动)、光盘 (CD)、数字视频盘 (DVD) 等可编程芯片。

[0039] 本文所使用的术语“模块”可以包括在一个或多个可编程处理器、ASIC、DSP、硬布线逻辑或它们的组合上执行的软件和 / 或固件 , 并且可存在于电子器件 / 装置内和 / 或显示器件内。

[0040] 现在转到附图 , 在附图中相同的标记表示相同的部件 , 图 1 图示出根据实施方案的装置或电子器件。所述装置可以为例如膝上型计算机、台式计算机、诸如但不限于移动电话、电子书阅读器、PDA 等手持式电子器件 , 或利用显示的任何这样的电子器件 , 而无论显示是处于在电子器件的外部或集成到电子器件内。图 1 的装置包括中央处理单元 CPU 209 , 以及与 CPU 209 可操作联接的图形处理单元 GPU215。CPU 209 包括操作系统 207 以及与操作系统 207 可操作联接的图形模块 201。在一些实施方案中 , 图形模块 201 可以实现为核模式驱动器。操作系统 207 还与也可在 CPU 209 上执行的配置用户接口 203 可操作地联接。配置用户接口 203 连同图形模块 201 一起形成拆分显示系统 200。在各个实施方案中 , CPU

209 和 GPU 215 可以各种方式来实现。例如, CPU 和 GPU 可包括一个或多个处理磁心并且可以以物理方式位于单独的集成电路上, 甚至可位于单独的印制电路板上。然而, 在一些实施方案中, CPU 209 和 GPU 215 可以为集成到单个集成电路管芯上的集成电路。CPU 和 GPU 可共享系统存储器, 系统存储器还可物理上位于 CPU 和 GPU 的外部或位于与 CPU 和 GPU 相同的管芯或印制电路板上。

[0041] 系统存储器可以包括帧缓冲器 219。然而, 在一些实施方案中, GPU 215 可具有专用存储器并且可包含帧缓冲器 219。在另外的实施方案中, 帧缓冲器 219 可分布在 GPU 215 专用存储器和由 CPU 209 与 GPU 215 共享的系统存储器之间。因此, 根据实施方案, 帧缓冲器 219 可存储在任何适当的存储器中, 无论是系统存储器还是诸如 GPU 215 专用存储器之类的专用存储器。帧缓冲器 219 包括两个主要部分, 虚拟显示缓冲组 221 和物理显示帧缓冲器 223。操作系统 207 将逻辑接口 213 提供给帧缓冲器 219, 然而经由接口 217 在 GPU 215 和帧缓冲器 219 之间发生实际的物理数据传送。根据实施方案, GPU 215 将数据提供给虚拟显示缓冲器组 221, 其中虚拟显示缓冲器分别对应于诸如物理显示器 101 之类的物理显示器的虚拟显示区域。由虚拟缓冲器 221 提供给物理显示帧缓冲器 223 的数据为显示数据, 即, 对应于虚拟显示的显示数据。物理显示缓冲器 223 对显示数据进行缓冲并且将显示数据发送到物理显示器 101, 以便在虚拟显示区域 109 和 111 上进行显示。物理显示缓冲器 223 被分割为两个帧, 即帧 A 和帧 B, 其中帧 A 和 B 中的每个均包含来自两个虚拟显示的显示数据。也就是说, 帧 A 将包含虚拟显示区域 #1 和虚拟显示区域 #2 的显示数据。同样, 帧 B 将包含虚拟显示区域 #1 和虚拟显示区域 #2 的显示数据。

[0042] 物理显示器 101 为显示器件并且可以为例如电视机、计算机监控器或集成到诸如膝上型计算机、手持式计算器件、移动电话、PDA 等另一电子器件中的显示器。显示器件可以为例如 CRT、LCD 平板、LED 平板、等离子体屏等, 即任何适当的显示技术。物理显示器 101 与 GPU 215 可操作地联接, 并且还与 CPU 209 可操作地联接 (例如经由 GPU 215)。物理显示器 101 可操作以显示与应用程序和 / 或诸如但不限于 JPG/JPEG、GIF、MPEG 等各种文件类型的电子文件、和 / 或由诸如但不限于“. doc”、“. pdf”等文件“扩展名”确定的文件有关的信息和图像。物理显示器 101 可以包括能对与在其屏幕空间上显示的图像有关的图像品质设置进行调节或配置的存储器和 / 或逻辑。物理显示器 101 包括接收并且响应诸如但不限于“EDID”信息 (“扩展显示标识数据”) 之类的信息的查询的能力。

[0043] 如图 1 中接近顶部所示, 物理显示器 101 包括物理显示可视面 100。根据实施方案, 通过边界 107 将物理显示可视面 100 分割为虚拟显示区域 #1109 和虚拟显示区域 #2111。因此, 虚拟显示区域为物理显示器 101 的可视面区域的分割部分。虚拟显示区域还可被视为与“虚拟显示”即, 下面将进一步描述的仿真物理显示相对应。两个虚拟显示区域均具有对应的虚拟显示缓冲器, 即, 分别为虚拟显示缓冲器 #1 和虚拟显示缓冲器 #2。尽管在图 1 所示的示例性实施方案中, 物理显示可视面 100 仅被分割成两个虚拟显示区域, 但是根据实施方案物理显示可视面 100 可被分割成任何期望数量的虚拟显示区域。因此, 帧缓冲器 219 可在虚拟显示缓冲器组 221 内具有 1 至 N 个虚拟显示缓冲器。如图 1 所示, 在每个显示配置内确定虚拟显示缓冲器组 221。如上所述, 显示配置可以是用户利用配置用户接口 203 确定的。虚拟显示缓冲器组 221 将显示数据提供给物理显示帧缓冲器 223, 物理显示帧缓冲器 223 将虚拟显示缓冲器数据合并到显示数据的单帧中并且通过接口 225 将它们提供给物

理显示器 101。

[0044] 根据各个实施方案,图形模块 201 提供关于 CPU 209 的可视化以及关于 GPU 215 的物理显示映射,如逻辑可视化/物理显示映射边界 211 所示。在操作中,操作系统 207 经由逻辑接口 213 将虚拟显示区域 109 和 111 觉察 (perceive) 为与 CPU 可操作联接的多个物理显示器。然而,从 GPU 215 的观点看,仅存在单个物理显示器,即物理显示器 101,并且该物理显示器与整个系统可操作地联接。图形模块 201 在操作系统 207 和 GPU 215 之间作接口以处理中断并且在它们之间提供信息,以使得操作系统表现得好像其与多个物理显示器交互一样。图形模块 201 和 GPU 215 之间的交互使 GPU 能够正常地表现为好像仅连接单个物理显示器一样,然而,实际上是来自虚拟显示区域的信息转回与物理显示器 101 的物理显示可视面 100 相对应的信息以使该信息能够提供给 GPU 215 的情况。换句话说,逻辑边界 211 的物理显示映射部分为图形模块 201 和 GPU 215 之间的接口。

[0045] 下面的图从图 2 至图 6 用于理解各个实施方案的操作。例如,图 2 图示出了物理显示器 101 的物理显示可视面 100 如何分割成两个虚拟显示区域 109 和 111。虚拟显示区域由垂直边界 107 限定。如图 2 所示,各个应用程序窗口可以被显示并且在各个虚拟显示区域内操作。例如,显示出虚拟显示区域 109 被显示具有应用程序窗口 103 (比如应用程序窗口 1 至 3),而虚拟显示区域 111 被显示具有应用程序窗口 4 至 6。此外,图 2 示出的是鼠标光标 105,其由本文所述的方法和装置处理并且下面将进行详细说明。转到图 3,显示为应用程序窗口 2 处于全屏模式,在该实施例中全屏模式为“独占模式”。也就是说,应用程序窗口 2 扩展以占据虚拟显示区域 109 的整个显示区域或屏幕空间,并且不显示任何边界。然而,不同于现有的分割实用程序,本文公开的各个实施方案使应用程序窗口 #2 能够在虚拟显示区域 109 内处于全屏模式,而不妨碍或遮住其它的虚拟显示区域 111。因此,如图 3 所示,在应用程序窗口 #2 在第一虚拟显示区域 109 内处于全屏模式的同时,存在于虚拟显示区域 111 内的应用程序窗口 4 至 6 仍可被用户被看到并且仍可成为关注焦点且被操作。使窗口成为关注焦点包括例如这样的操作:在所述操作之后,计算器件或电子器件的用户通常通过将鼠标光标在屏幕空间上的区域内移动并且点击来进行选择,即“选择”应用程序窗口,从而查看在应用程序窗口内显示的信息或者继续进行一些其它与应用程序或应用程序窗口有关的操作(诸如但不限于调整应用程序窗口的尺寸或者移动应用程序窗口)。在另一实施例中,应用程序窗口还可被置为全屏模式,其中应用程序窗口的边界仍可被显示并且沿着虚拟显示区域的周边延伸。在此情况下,应用程序窗口也不妨碍相邻的虚拟显示区域或在相邻的虚拟显示区域内显示的任何应用程序。也就是说,“全屏模式”可以为例如如在例如应用程序屏幕填满显示器的可视区域并且不具有边界的游戏(或其它)应用程序中使用的“独占模式”,或应用程序窗口在显示的可视区域内完全扩充但是仍可显示出应用程序窗口周边周围的边界的全屏窗口应用程序。

[0046] 图 4 图示出各个实施方案的进一步的能力。在图 4 中,应用程序窗口 2 被显示为移动而部分横过两个虚拟显示区域 109 和 111 或者在两个虚拟显示区域 109 和 111 之间移动。从图 4 中可以看出,应用程序窗口 #2 移动到虚拟显示区域 111 中不会妨碍各个应用程序窗口在虚拟显示区域 111 内的操作。图 1 所示的装置在操作系统 207 和 GPU 215 之间进行必要的转换。也就是说,图形模块 201 采取适当的措施以使诸如应用程序窗口 #2 之类的对象越过边界 107,从而使 GPU 215 接收适当的物理显示映射并且使操作系统 207 接收适合

于与虚拟显示区域 109 和 111 相对应的两个仿真物理显示的适当的信息。

[0047] 图 5 图示出鼠标光标 105 在两个虚拟显示区域 109 和 111 之间移动的方案。在图 5 中, 显示出在光标 105 从一个虚拟显示区域移动至另一虚拟显示区域时光标 105 越过边界 107。各个实施方案同样处理这样的鼠标光标移动方案, 下面将进行进一步讨论。

[0048] 图 6 示出了另一使用情况方案。在图 6 中, 物理显示器 101 与第二物理显示器 601 结合使用。第二物理显示器 601 包括物理显示器 101 上显示出的桌面或虚拟显示区域 109 的代用件。如图 6 所示, 物理显示器 101 上的应用程序窗口 #2 可移动而越过虚拟显示区域边界 107。然而, 相对于物理显示器 601 上显示出的虚拟显示区域 109 的代用件, 物理显示器 101 上的虚拟显示区域边界 107 对应于物理显示器 601 的实际物理显示可视面的端, 以使应用程序窗口 #2 移动而离开可视显示区域, 即在监控器窗口之外。

[0049] 通过图 7 至图 15 中提供的流程图来最佳地理解各个实施方案的操作以及实施方案如何处理前面的图中所示的各种方案, 现在将进行讨论。

[0050] 为了实现各个实施方案的操作, 图形模块 201 执行各种显示管理任务, 包括目标管理、EDID 管理、连接管理、模式列表管理、显示属性管理以及逆拓扑管理, 所述逆拓扑管理为诸如 Vsync 中断之类的中断的管理。根据实施方案图形模块必须执行的其它任务与渲染管理有关。因此, 在众多其它事项中, 根据实施方案的图形模块必须执行显示渲染检测、可显示帧成分以及包括帧缓冲器、命令缓冲器及成分背景管理以及光标管理在内的资源管理。因此, 图形模块 201 保持从物理显示器 101 获得的一个或多个虚拟显示并且将与虚拟显示有关的信息报告给操作系统 207。虚拟显示或者与虚拟显示区域对应的仿真物理显示被赋予与实际的物理显示器相同的属性, 例如 EDID、连接性、视频输出类型和中断。图形模块 201 可将专用存储器块作为将在物理显示器 101 上显示的最终图像的交换链保存在系统存储器或专用存储器中。图形模块 201 迫使仿真物理显示与物理显示器 101 共享相同的同步信号 (Vsync)。

[0051] 例如, 在一些实施方案中, 图形模块 201 可与桌面窗口管理器 (DWM) 交互。例如, 当桌面窗口管理器为关闭时, 图形模块可在专用存储器块中构建帧以便在物理显示器 101 上显示。显示数据帧的构建可以基于通过例如 “present()” 或 “flip()” 对两个仿真物理显示上的表面的新渲染请求的检测。对于桌面窗口管理器打开的情况, 图形模块 201 可将同步信号 (Vsync) 报告给两个虚拟显示的操作系统。当翻转请求接收到 DWM 打开或者全屏模式时, 对于虚拟显示区域 (即, 仿真物理显示) 中的任一个, 图形模块 201 执行从仿真物理显示的帧到交换链的一个帧的合成并且在物理显示器 101 上显示出新构建的帧。

[0052] 因为图形模块 201 确保了虚拟显示区域的应用程序独立性并且保持该状态而无论来自可能在各个虚拟显示区域内运行的各个应用程序的任何配置变化如何, 所以各个实施方案对于应用程序而言是透明的。为了保持用户感受的连续性和稳定性, 如果配置在由用户制作的各个虚拟显示区域上变化, 则图形模块 201 根据需要来采取措施。例如, 通过图形模块 201 来处理桌面的再定位或显示器上分辨率的变化。

[0053] 转到图 7, 该图示出了依照实施方案的帧合成方法。帧合成方法实质上为对于虚拟显示区域的特定必要性而调整的双缓冲操作。例如, 在块 701 中, 操作系统具有准备显示在一个虚拟显示区域上的缓冲数据, 在块 703 中, 图形模块接收来自操作系统的情况如此的通知。双缓冲方案将物理显示帧缓冲器 223 划分为两个部分, 即帧 A 和帧 B。因此, 如果待

显示的下个可视帧为如块 705 中所示的帧 A,则在块 707 中确定是否准备好显示左屏面或右屏面的判定。从块 707 延伸的流程图的两侧是对称的,因为,如果一个虚拟显示区域存在新渲染帧,则相对的或其它的虚拟显示区域表面数据被复制并且相对于该特定虚拟显示区域帧被标记为完成。例如,如果在块 711 所示的帧 A 中,右屏面表面存在新渲染帧,则如块 727 所示仅复制左屏面表面数据。然后,相对于如块 729 中所示的帧 A 中的左屏面信息,缓冲器被标记为完成。因为如块 731 所示必须更新帧 B,所以相对于左屏面帧 B 未被标记。然后,方法继续进行到块 733 至 743,其中相对于操作系统 207 响应中断。如果在块 711 所示的帧 A 中,右屏面表面不存在新渲染帧,则如块 723 所示复制左屏面和右屏面表面数据。然后,相对于如块 725 中所示的帧 A 中的右屏面信息,缓冲器被标记为完成。类似地,如果在块 709 所示的帧 A 中,左屏面表面存在新渲染帧,则如块 717 所示仅复制右屏面表面数据。然后,相对于如块 719 中所示的帧 A 中的右屏面信息,缓冲器被标记为完成。帧 B 相对于右屏面不进行标记,因为它必须被更新,如块 721 所示。然后,该方法继续进行到块 733 至 743,其中相对于操作系统 207 响应中断。如果在块 709 所示的帧 A 中,左屏面表面不存在新渲染帧,则如块 713 所示复制左屏面和右屏面表面数据。然后,相对于如块 715 中所示的帧 A 中的左屏面信息,缓冲器被标记为完成。图 8 示出了依照实施方案的仿真物理显示的硬件中断处理和硬件中断模拟。例如,GPU 215 可发送硬件中断,如块 801 所示。在所述的一些实施方案中,该中断可以为 Vsync 中断。如块 803 所示,图形模块 201 得到与 GPU 的控制器 ID 相关联的目标 ID。图形模块 201 随后将得到如块 805 所示的与控制器 ID 相关联的源 ID,并且在块 807 中将判定特定源是否正在参与拆分显示模式。如果为否,则图形模块 201 将取回的目标 ID 以及块 817 中所示的表面地址报告给 OS,并且将在块 819 中将控制返回到 OS。

[0054] 如果块 807 中的源正在参与拆分显示模式,则图形模块 201 查验任何附加的源是否正在参与如块 809 中所示的拆分显示模式。如果为是,则图形模块将搜索正在参与如块 811 所示的拆分显示的下个源 ID 和目标 ID 的阵列(比如阵列 825)。举例来说,在如块 821 中所示的操作系统拆分模式设定或拆分模式变化请求之后,如块 823 所示保存正在参与拆分显示的源 ID 和目标 ID,之后,阵列 825 被填入。图形模块随后将找到的源 ID 作为取回如块 813 所示的相关表面地址的索引。表面地址可从例如可存储于系统存储器中的表 831 中获得。举例来说,在如块 827 中所示的操作系统已缓冲数据准备好在显示 1 或 2 上显示之后,如块 829 所示使用源 ID 作为索引以设置表面地址的阵列,之后,表 831 被填入。然后,图形模块将取回的目标 ID 和表面地址报告给操作系统,如块 815 所示。当正在参与拆分显示的所有源已由框 809 确定时,图形模块 201 将如框 817 和 819 所示的报告给操作系统。因此,图形模块 201 模拟仿真物理显示对操作系统的中断,以使操作系统将不能偶然得出硬件未正确运行的结论;并且不能偶然地将 GPU 215 置为复位模式。应当理解的是,在图 8 中,目标 ID 可被视为作为与物理显示器对应的 ID 的监控器 ID。因此,图形模块 201 必须针对虚拟显示区域来模拟目标 ID,使得操作系统将虚拟显示区域视为单独的物理显示器。源 ID 对应于诸如虚拟显示区域 109 和 111 等虚拟显示区域或虚拟桌面。

[0055] 图 9 示出了当桌面窗口管理器(DWM)未运行时合成的处理。图形模块 201 通过诸如 Vsync 中断回调之类的硬件中断用 GPU 215 来创建回调 901。在此情况下,当中断发生时,在 903 中图形模块 201 将查验两个虚拟显示的两个新内容状态。如果两个虚拟显示区

域 909 均发生变化,则图形模块 201 将两个虚拟显示区域复制到物理显示帧缓冲器 223 中的第一显示帧 A 中且如块 919 中所示为两个虚拟显示区域重设新内容标记。然后,该方法继续进行到块 921,其中复制命令围栏被提交。如果任一虚拟显示区域 909 发生变化,则图形模块 201 仅将特定虚拟显示区域复制到物理显示帧缓冲器 223 中的第一显示帧 A 中且为该特定虚拟显示区域重设新内容标记,如块 905 和 907、块 911 和 915 以及块 913 和 917 所示。然而,图形模块不需要将任何信息报告给操作系统 207。当如块 923 所示,操作系统已缓冲数据准备好在显示 1 或 2 上显示时,表面地址的阵列(如块 927 中所示的这个阵列)通过使用源 ID 作为索引而形成,如块 925 中所示。当如块 929 所示,操作系统请求图形模块更新一个或多个虚拟显示缓冲器时,查验表面地址的阵列(如块 927 中所示的这个阵列)中的目的地 MC 地址。该方法继续进行以设置新内容的阵列,例如块 933。

[0056] 图 10 示出了与例如图 5 所示的方案对应的光标管理的方法,其中光标 105 移动而越过虚拟显示区域边界 107。如块 1001 所示,光标位置被调度和设定。接着,如块 1003 所示,该方法判定所请求的源是否处于拆分显示中。如果是,则图形模块 201 判定光标是否位于左屏面或右屏面上,也就是,图形模块 201 判定出鼠标光标存在于哪个虚拟显示上,如块 1005 所示。鼠标光标的位置从虚拟显示区域上的相对位置转换为与整个物理显示器 101 的物理显示可视面 100 对应的物理显示位置,如块 1007 所示。该信息随后提供给如块 1009 所示的 GPU 215。

[0057] 图 11-13 总结了各个实施方案的操作。如图 11 所示,块 1101,单一物理显示器的可视区域可分割为至少两个虚拟可视区域。各个实施方案的图形模块 201 随后可以将虚拟可视区域仿真为相对于操作系统 207 的物理显示,如块 1103 中所示。图 12 示出了系统 200 可以包括配置用户接口 203。因此,在块 1201 中,用户接口 203 可被设置以用于虚拟显示的配置。在块 1203 中,图形模块 201 可经由逻辑接口 205 接收来自配置用户接口 203 的显示布置的用户选择输入。如块 1205 中所示,图形模块 201 将虚拟显示数据映射到图形模块 201 和 GPU 215 之间的硬件接口上的物理显示器 101 上。如块 1207 中所示,操作系统的物理显示数据会被映射到虚拟显示部分。图形模块 201 随后将处理如块 1209 所示的越过虚拟显示部的图形移动。图 13 示出了当应用程序显示为处于全屏模式时虚拟显示区域的操作。在块 1301 中,第一应用程序可在第一虚拟可视区域内以全屏模式显示。在第二虚拟可视区域中为第二应用程序设置入口,而第一应用程序在如块 1302 所示的第一虚拟可视区域内以全屏模式显示。

[0058] 图 14 和图 15 为示出配置用户接口 203 的进一步操作并且示出物理显示器 101 上的拆分显示模式的启动和关闭的流程图。在块 1401 中,配置用户接口 203 可将用户希望进入拆分显示模式的通知发送给图形模块 201。在块 1403 中,图形模块 201 将拆分显示启动序列的开始通知给配置用户接口 203。这可通过利用例如“暂停”事件来实现。在块 1405 中,图形模块 203 保存拆分显示数据并且通知操作系统 207,原始显示,即物理显示器 101 被分离并且现在两个新显示被关联。两个新显示实际上为物理显示器 101 中的图 1 所示的虚拟显示区域 109 和 111 的仿真物理显示。操作系统随后查询如块 1407 所示的显示信息,并且图形模块 201 生成两个虚假的 EDID 并且将两个虚假的 EDID 提供给操作系统 207。操作系统 207 随后查询监控器模式,图形模块 201 为操作系统 207 提供如块 1409 所示的监控器源模式。在块 1411 中,操作系统 207 查询模式列表,图形模块 201 为操作系统 207 提供支持

模式列表。在块 1413 中,操作系统 207 将该模式设定在两个显示上,即,与对应于两个虚拟显示区域 109 和 111 的两个仿真物理显示对应。在块 1415 中,图形模块 201 利用例如“继续执行”事件将拆分显示序列结束通知给配置用户接口 203。在块 1417 中,操作系统 207 开始在两个虚拟显示区域上进行渲染,图形模块 201 在单一物理显示器 101 上合成两个独立的桌面。图 15 为示出了拆分模式关闭的操作的另一流程图。在图形模块 201 在块 1501 中接收拆分显示模式关闭的通知之后,它在块 1503 中通知配置用户接口 203 开始拆分显示关闭序列。图形模块 201 通过将两个虚假显示即与两个虚假 EDID 对应的两个仿真物理显示分离并且现在如块 1505 所示原始物理显示器 101 联接通知给操作系统 207 来关闭拆分显示模式。图形模块 201 为操作系统 207 提供如块 1507 至 1515 所示所需的信息,以使操作系统可再次在如块 1517 所示的物理显示器 101 的物理显示可视面 100 上开始进行渲染。

[0059] 因此,依照各个实施方案,在一些实施方案中,图形模块 201 可实现为核模式驱动器,在操作系统 207 和图形处理单元 215(以及诸如芯组等相关图形硬件)之间提供接口,以使操作系统 207 觉察到多个物理显示器的操作,即,图形处理单元以及相关的图形硬件相对于其特定操作仅觉察到单一物理显示器。因此,物理显示器可被分割为如上所述的各个虚拟显示区域。图 16 示出了可用作配置用户接口 203 的部分的示例性应用程序窗口。应用程序窗口 1600 指示用户选择期望的多个显示布置(如块 1605 所示)且允许用户通过例如下拉菜单 1603 来挑选显示布局。配置用户接口 203 的应用程序窗口 1600 可提供物理显示可视面 1601 的可视图形并且示出诸如物理显示划分边界 1609 之类的选定虚拟显示区域划分。用户随后可利用光标 1607 来点击确认并且如图 14 中的块 1401 所示进入拆分显示模式,其中配置用户接口 203 将拆分显示模式的通知发送给图形模块 201。因此,图 14 的块 1401 所示的动作可以是在配置用户接口 203 的应用程序窗口 1600 内用户动作的结果。

[0060] 应用程序窗口 1600 是用于解释可能的用户接口的简化的示例性应用程序窗口。然而,应用程序窗口 1600 应当被理解为仅为示例性的,因此可在所能理解的各个实施方案内使用应用程序窗口 1600 的其它各种布局 and 配置。例如,下拉菜单 1603 可提供物理显示可视区域的配置的文本描述,然而,下拉菜单还可提供如图 17 所示的配置类型 1700 所示的配置类型的符号。如图 17 所示,各种配置类型 1700 可被提供而作为应用程序窗口 1600 的下拉菜单 1603 的部分。因此,用户可仅选择下拉菜单并且从配置类型 1700 的配置选项中选择出适当的配置。当然,配置类型 1700 不应被视为包含一切的组,而是应被视为仅示例了各个实施方案可用的一些可能的配置。也就是,尽管图 17 所示的示例性的配置类型 1700 示出了最多四个虚拟显示区域,可存在大于四个的更多虚拟显示区域。此外,图 17 所示的虚拟显示区域由垂直位置边界和水平位置边限定。然而,虚拟显示区域不需仅由水平或垂直边限定,而是可由各种几何形状和图案限定,这些形状和图案诸如但不限于三角形、矩形、圆形或其它形状的虚拟显示区域边界图案。

[0061] 因此,本文已经公开了将显示面分割为多个虚拟显示区域的操作的装置和方法。与本文所公开的实施方案等同的其它变型例对于本领域技术人员而言将是显而易见的,并且这些变型例仍与由随附的权利要求书限定的实施方案的范围相一致。

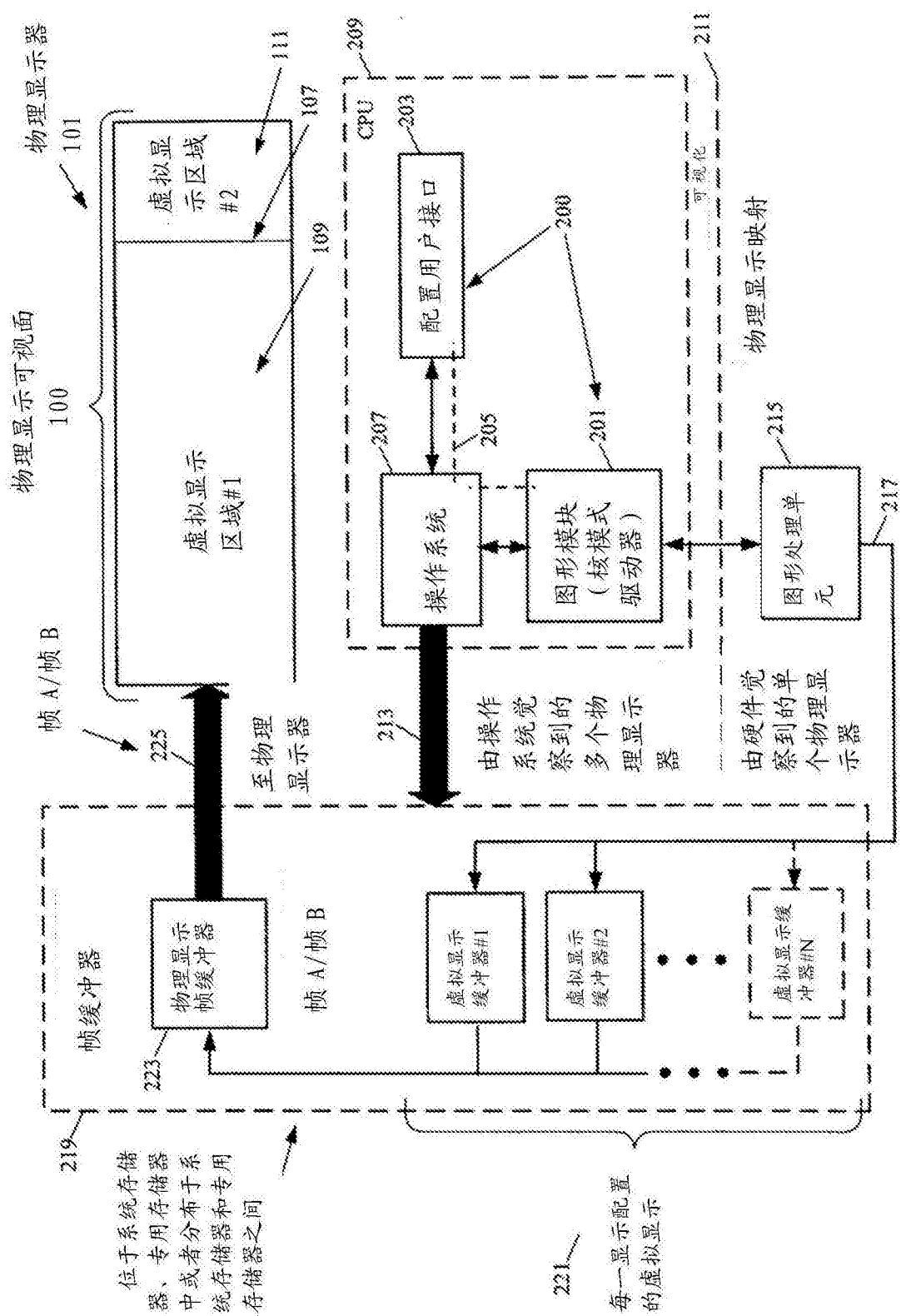


图 1

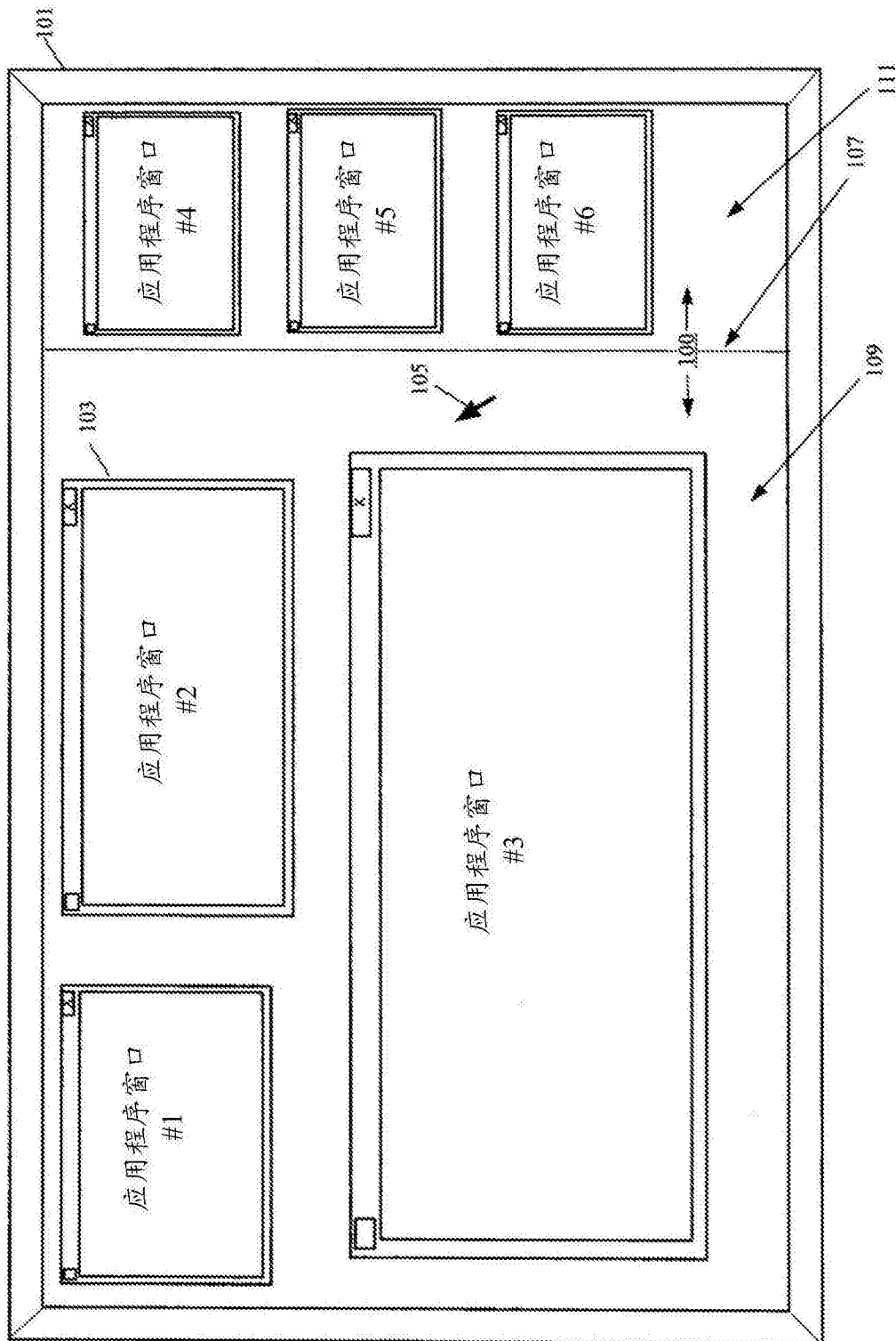


图 2

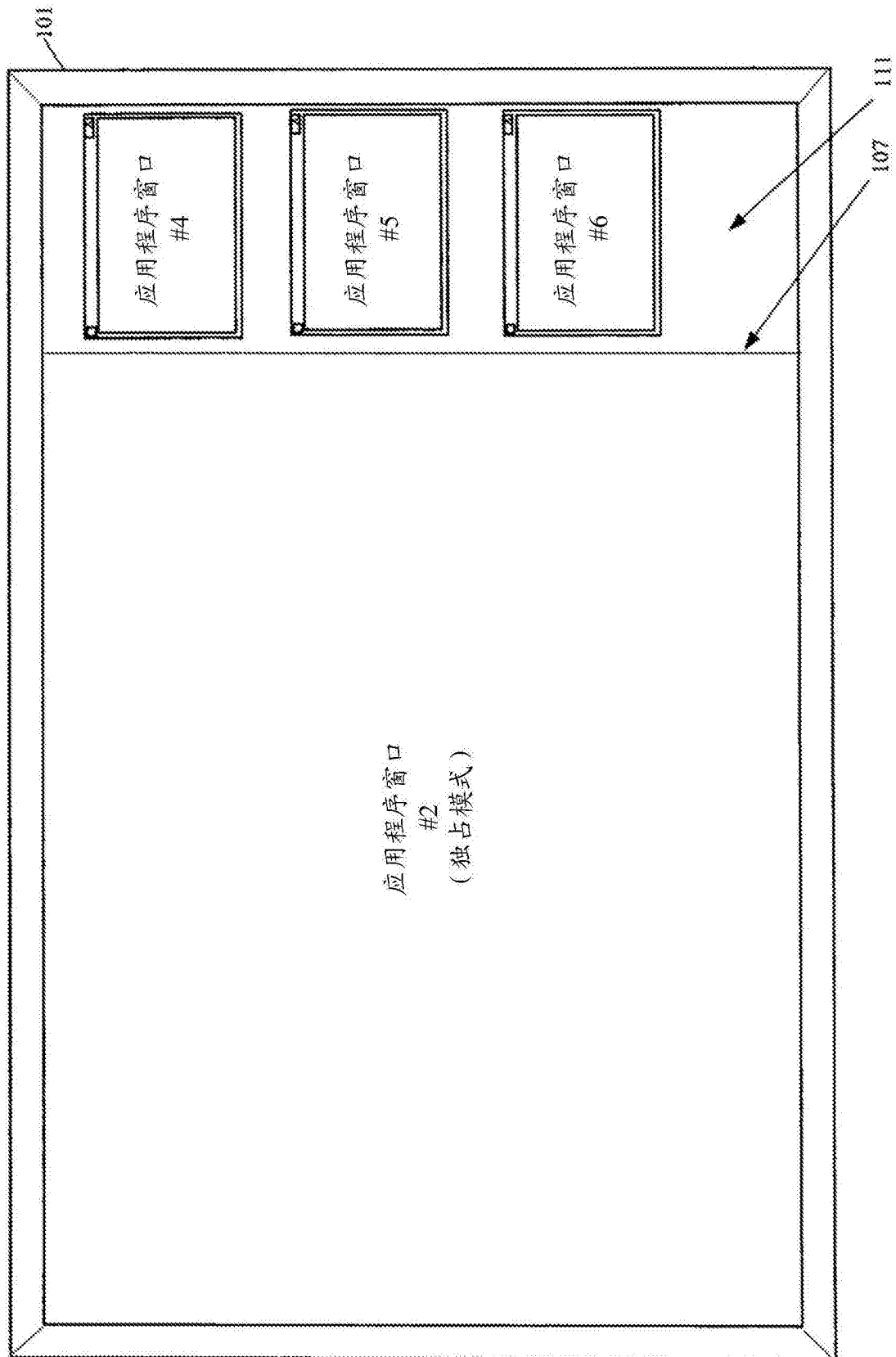


图 3

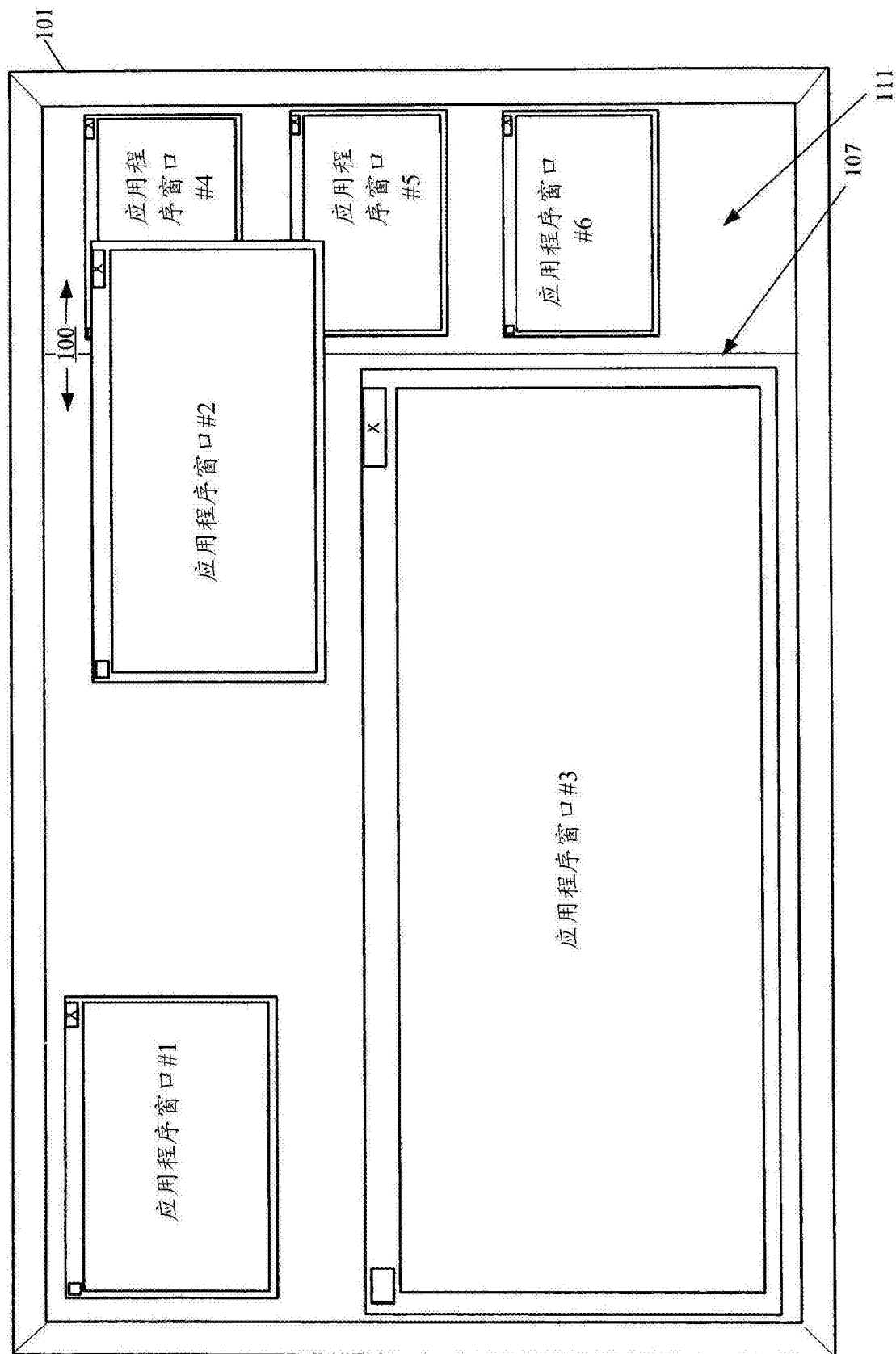


图 4

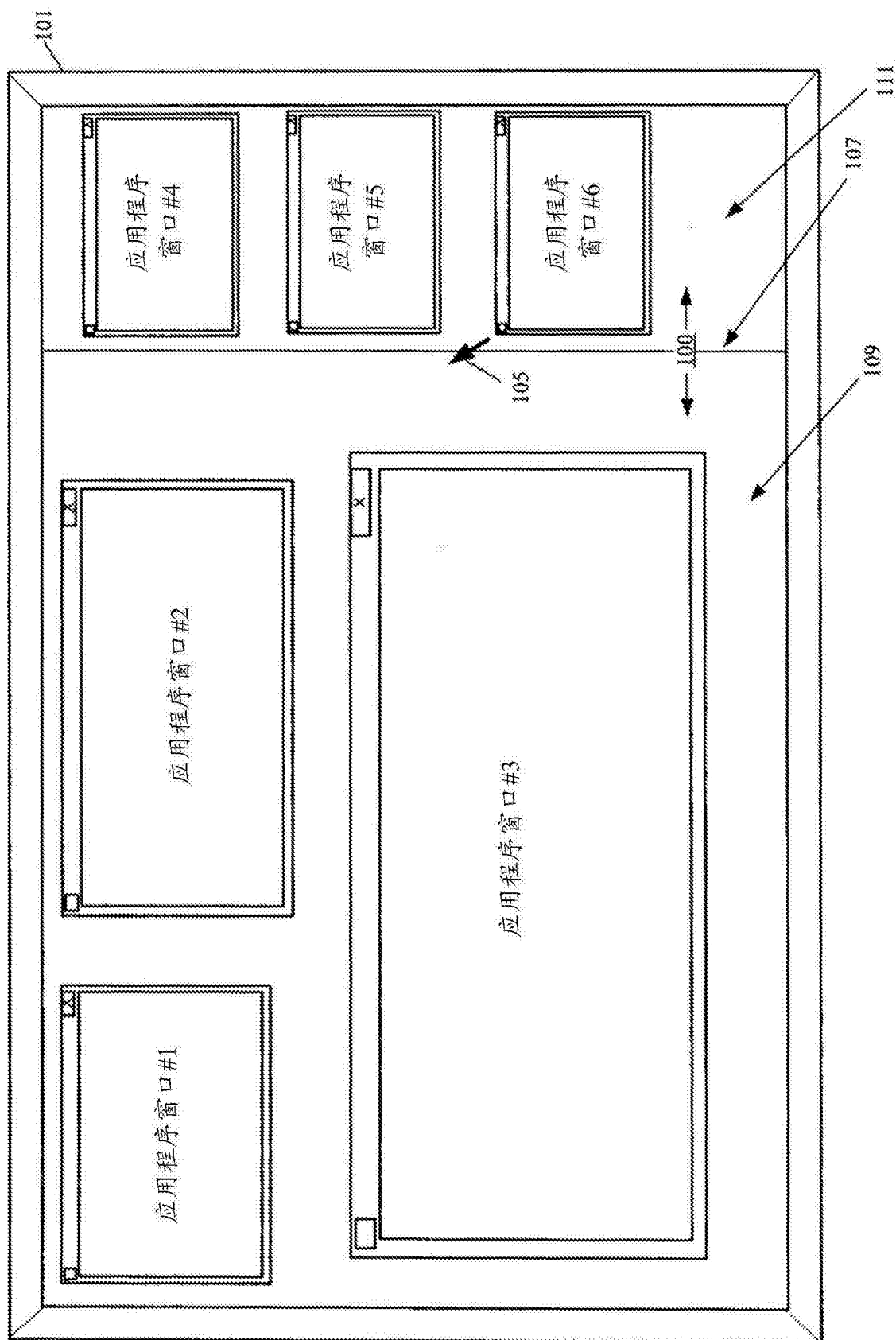


图 5

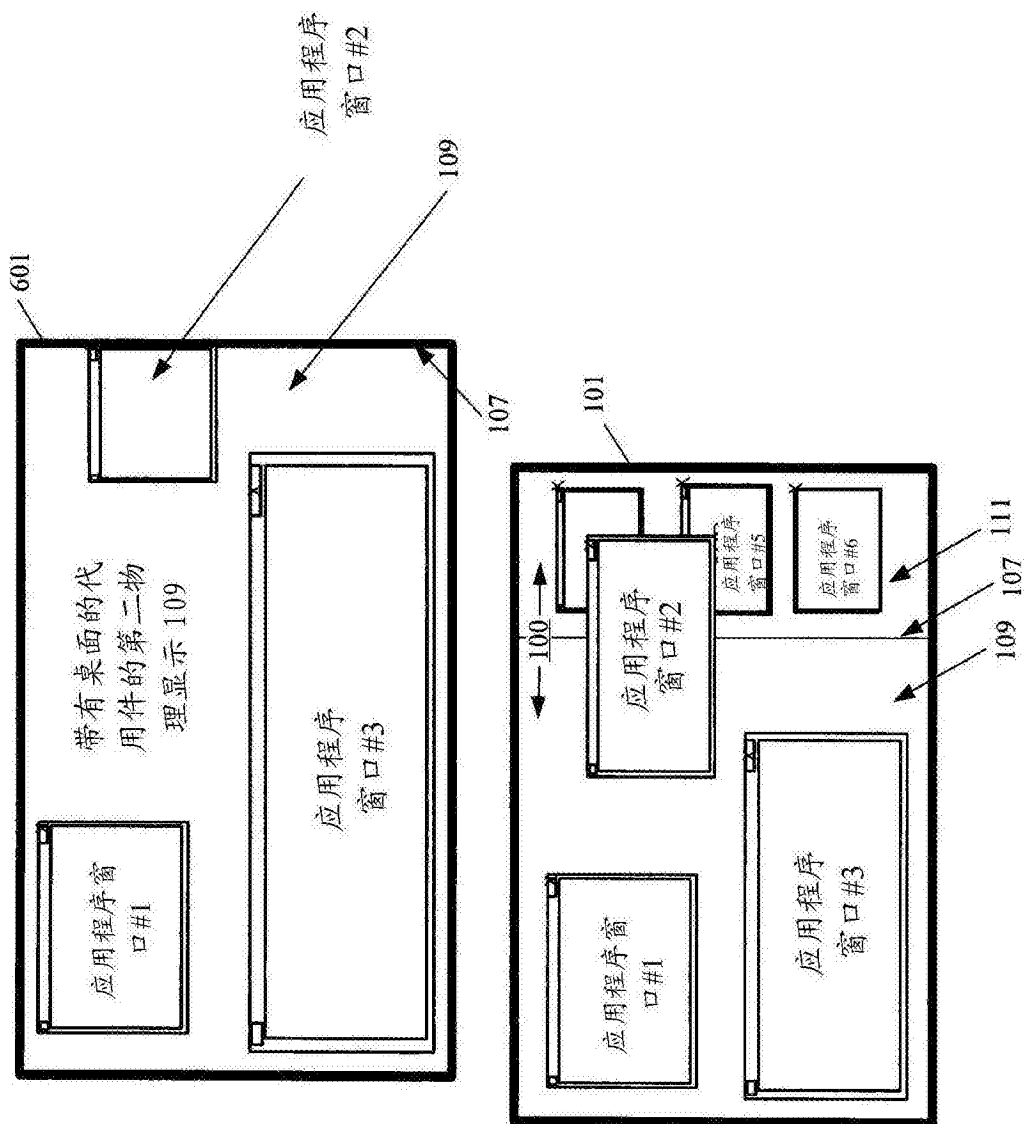


图 6

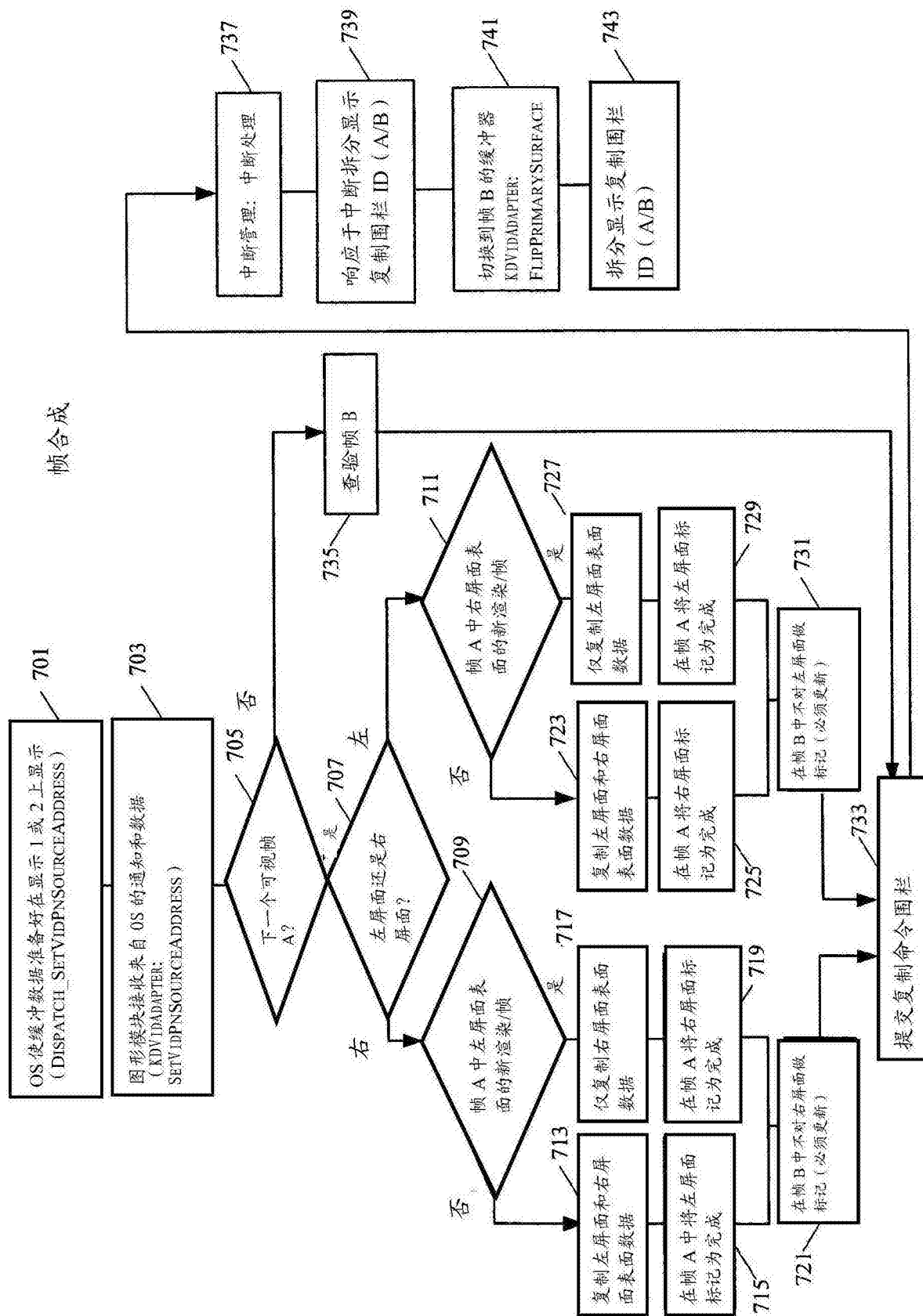


图 7

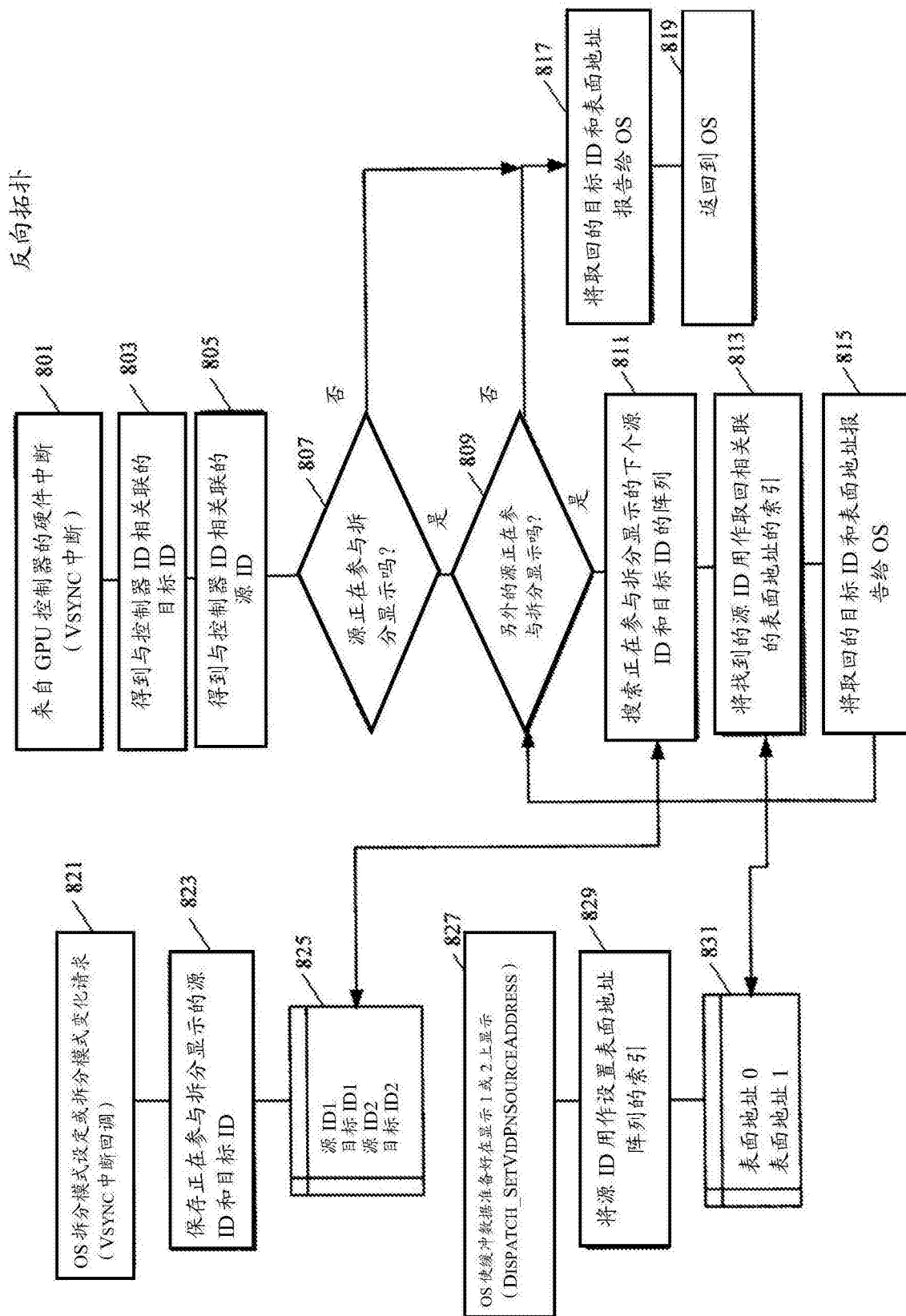


图 8

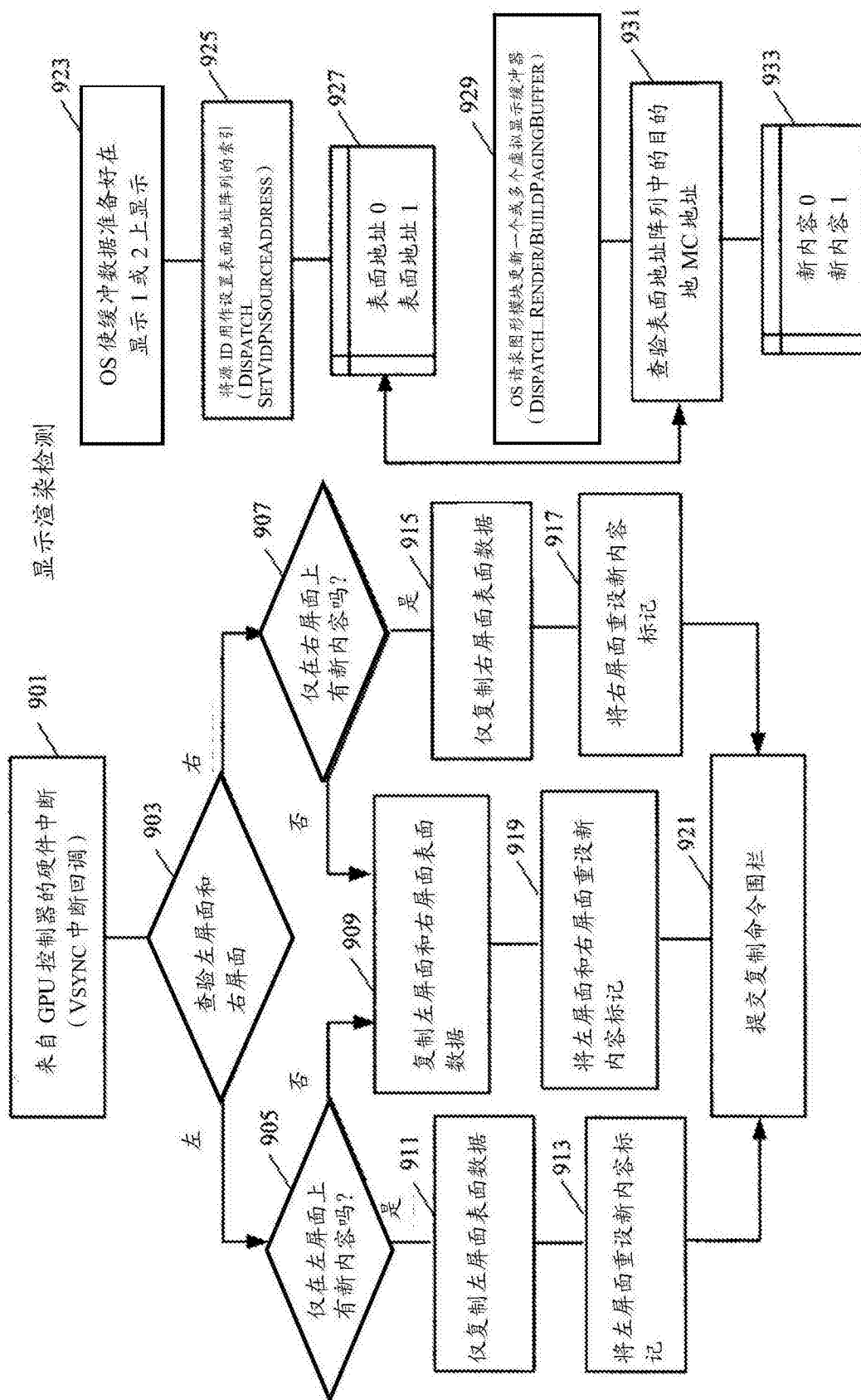


图 9

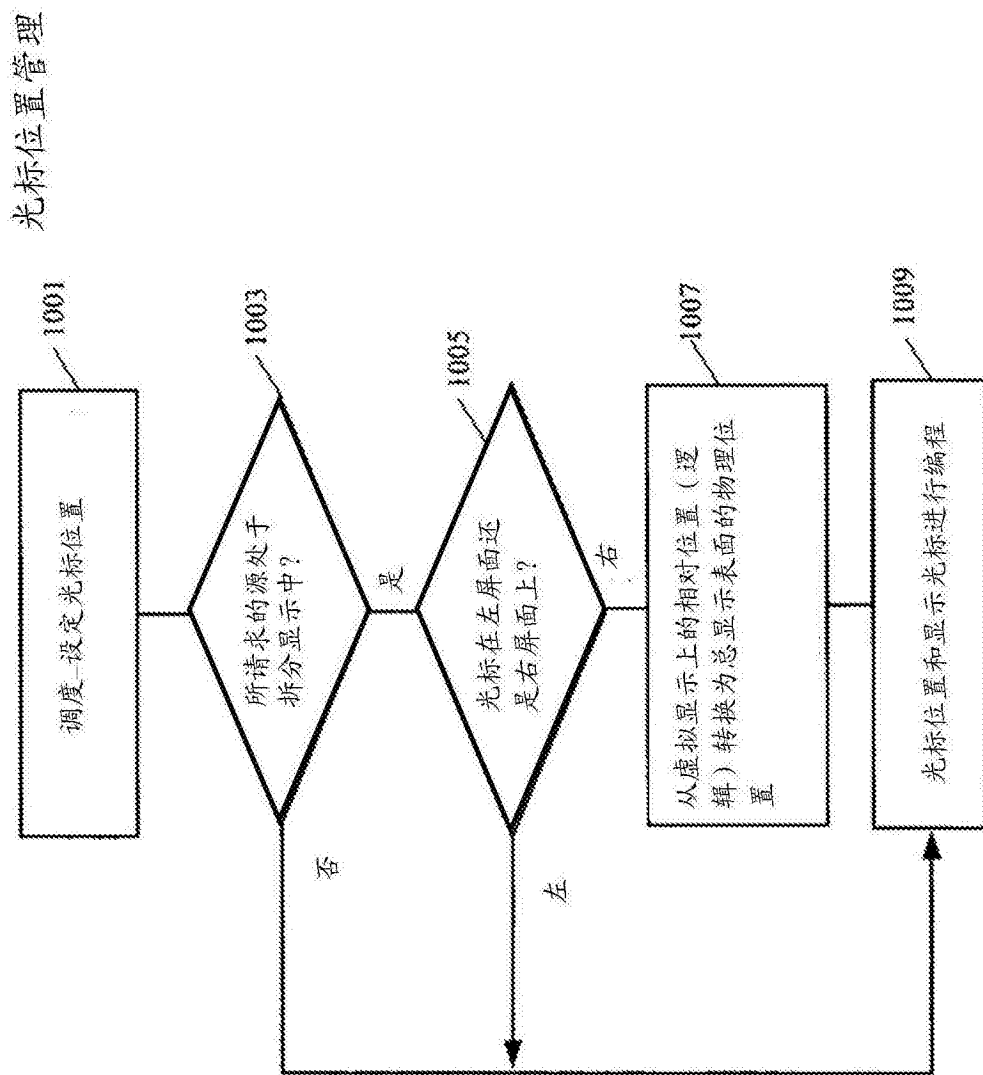


图 10

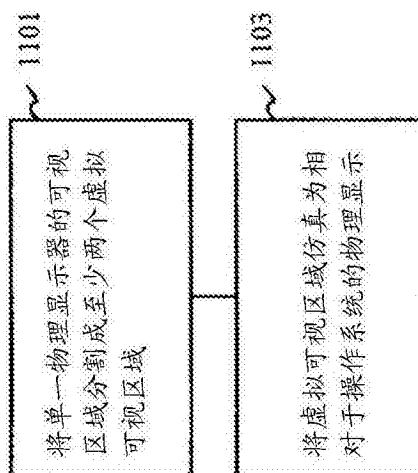


图 11

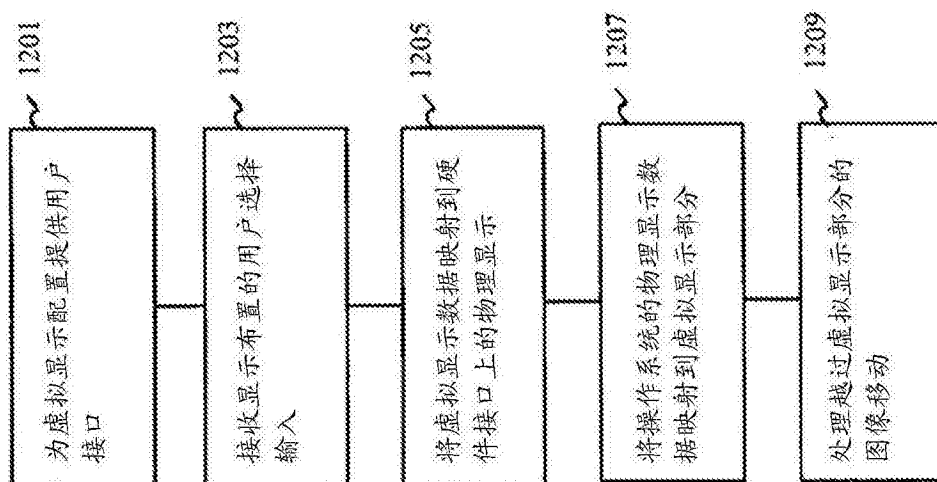


图 12

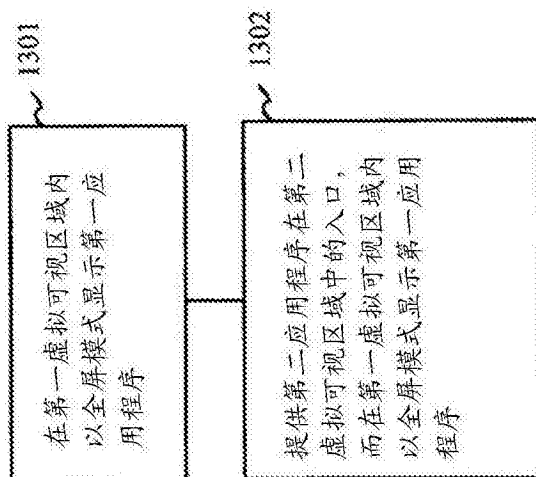


图 13

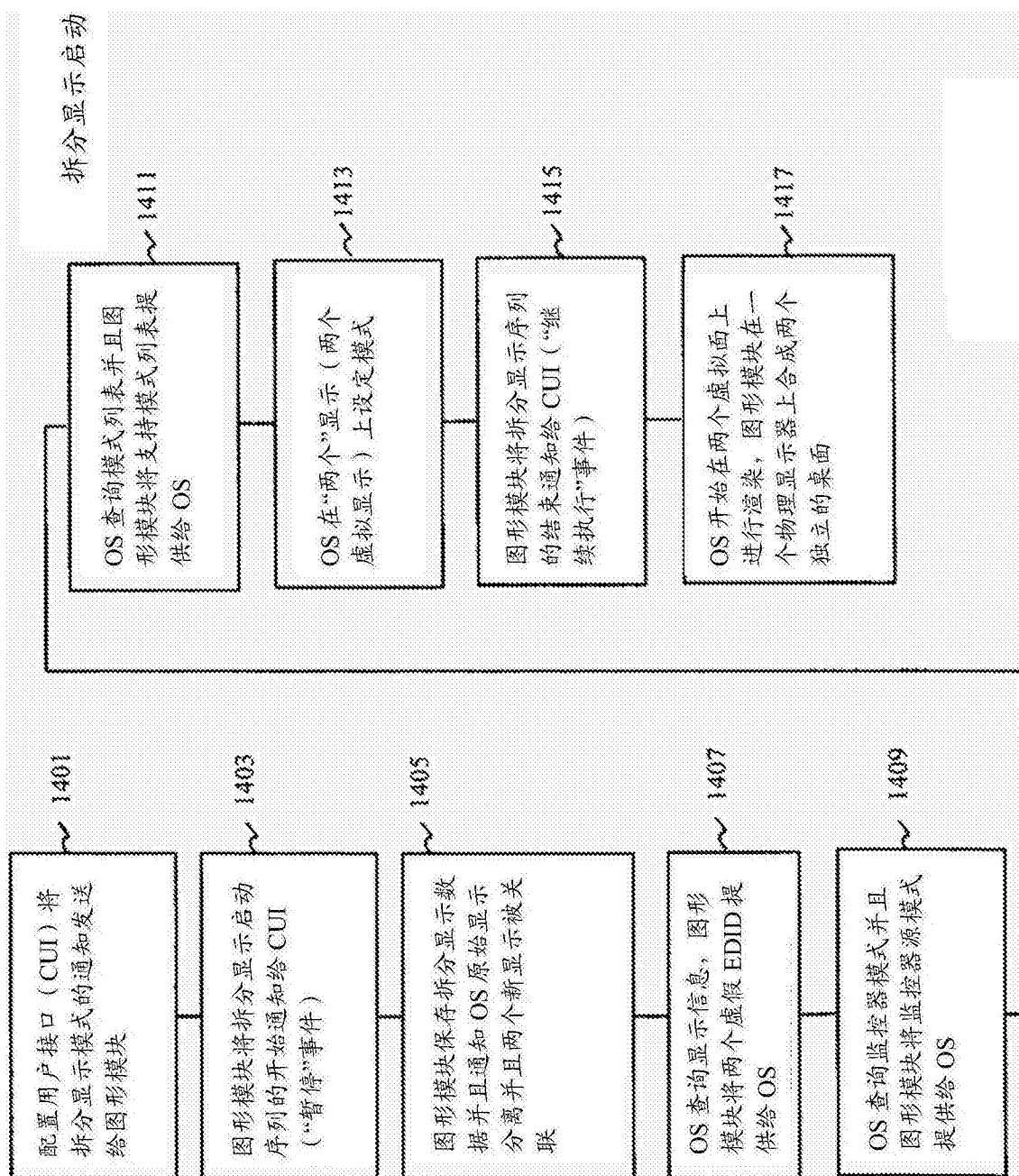


图 14

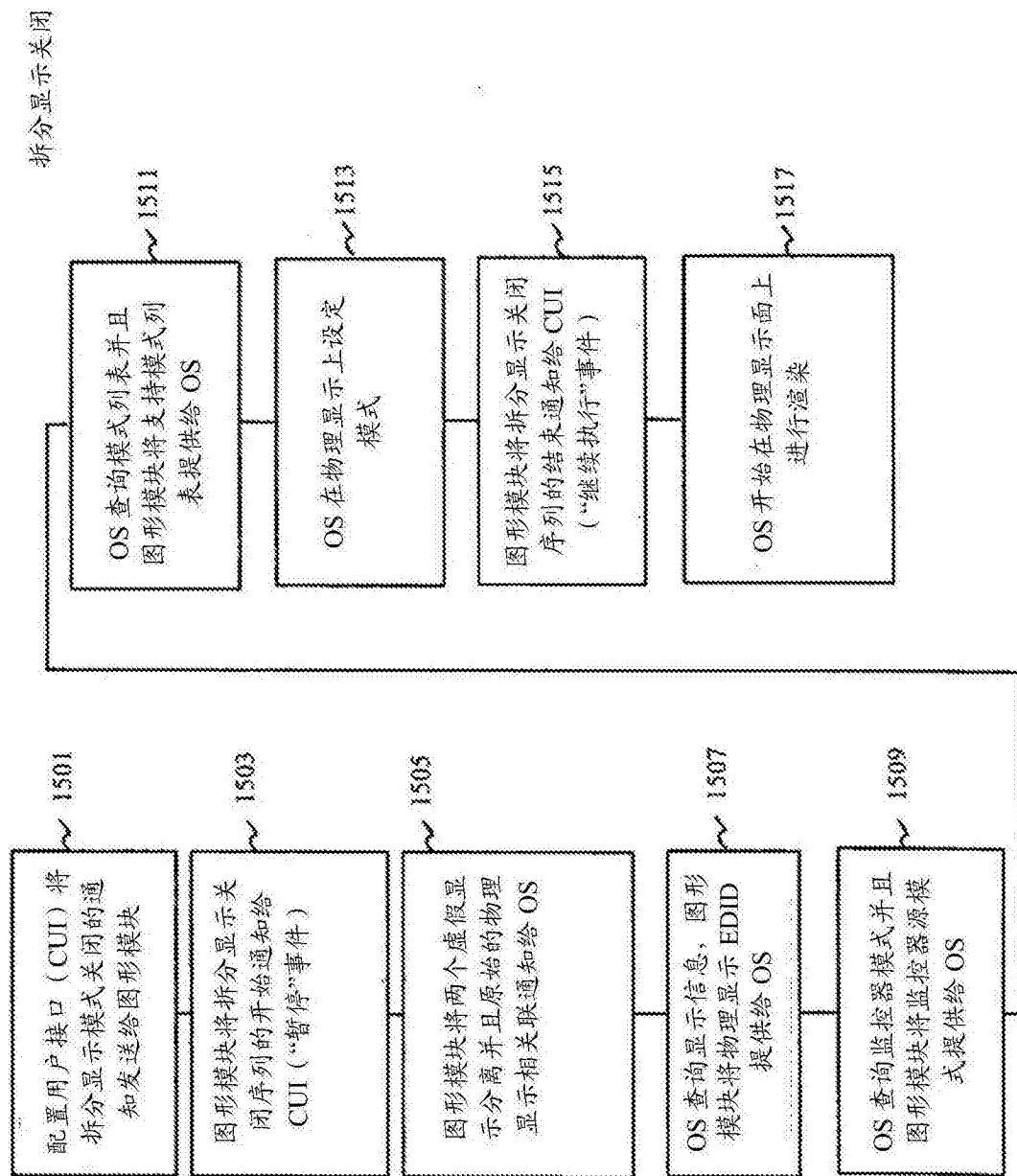


图 15

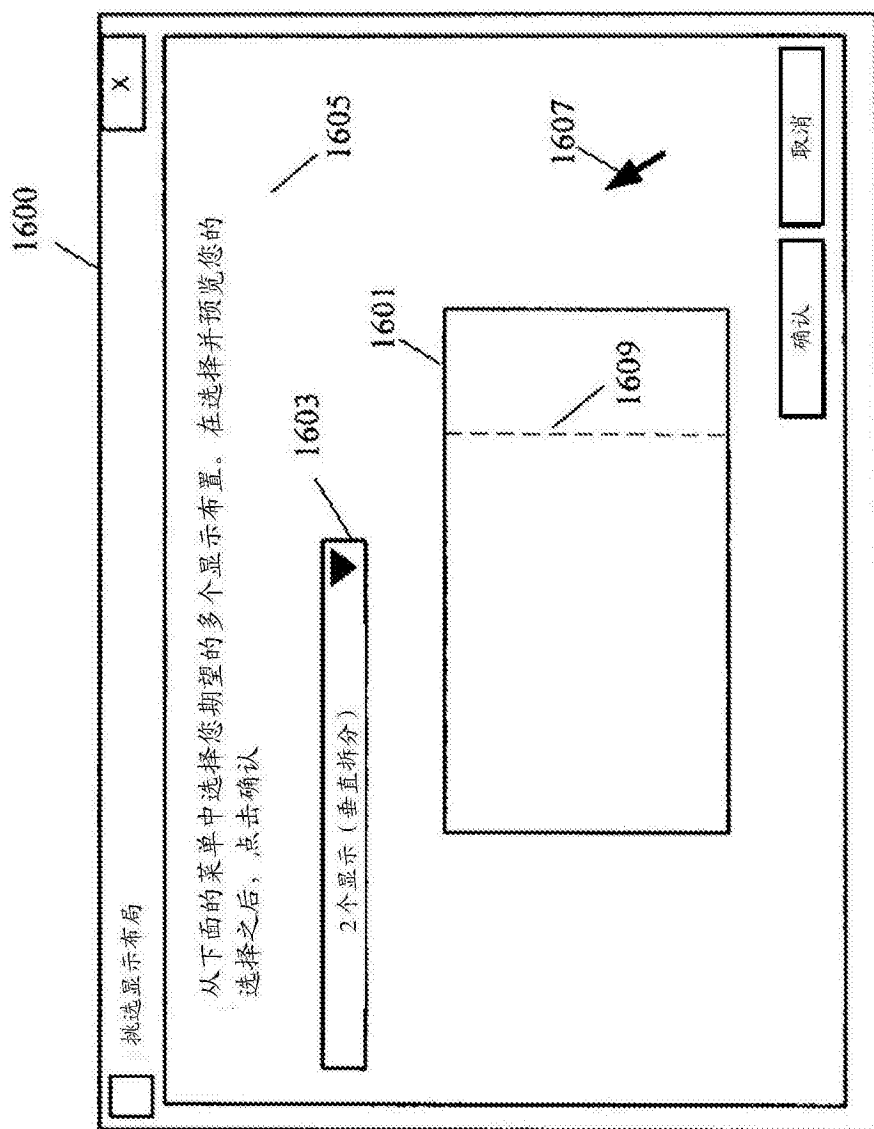


图 16

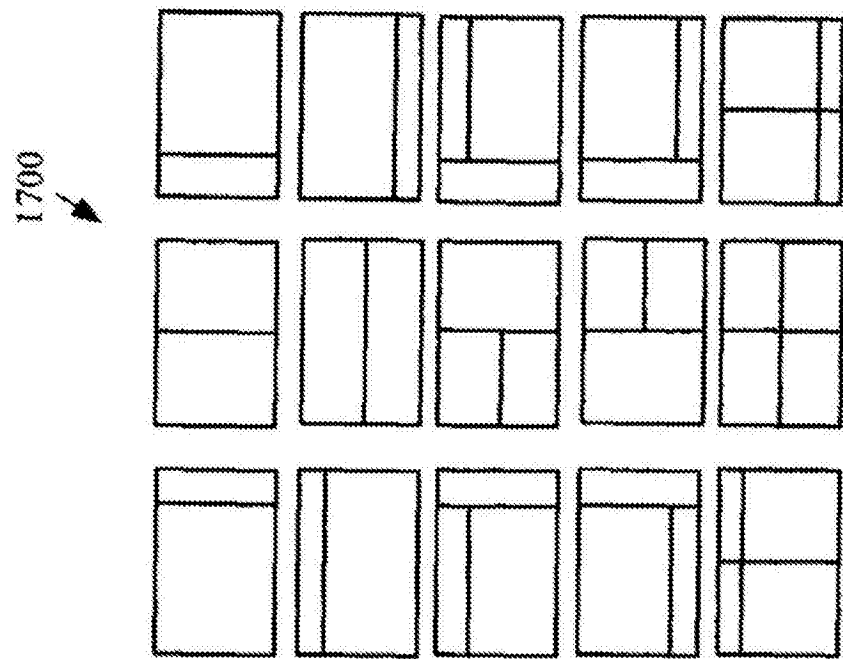


图 17