



(12) 发明专利申请

(10) 申请公布号 CN 105283828 A

(43) 申请公布日 2016. 01. 27

(21) 申请号 201480032100. 0

(51) Int. Cl.

(22) 申请日 2014. 05. 01

G06F 3/041(2006. 01)

G06F 3/0488(2006. 01)

(30) 优先权数据

61/832, 703 2013. 06. 07 US

14/192, 536 2014. 02. 27 US

(85) PCT国际申请进入国家阶段日

2015. 12. 04

(86) PCT国际申请的申请数据

PCT/US2014/036447 2014. 05. 01

(87) PCT国际申请的公布数据

W02014/197142 EN 2014. 12. 11

(71) 申请人 苹果公司

地址 美国加利福尼亚州

(72) 发明人 W·C·韦斯特曼 A·Y·本巴萨特

(74) 专利代理机构 北京市金杜律师事务所  
11256

代理人 王茂华

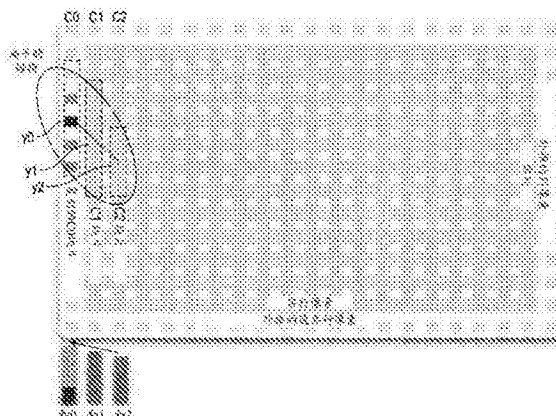
权利要求书4页 说明书9页 附图7页

(54) 发明名称

边框边缘处的触摸检测

(57) 摘要

本发明涉及一种外推接近信息以生成触摸节点(还称为触摸像素)的边界列或行并随后将椭圆拟合到包括所述外推的边界触摸节点的触点块的方法。另外,可基于触点的长轴和所述触点到触摸感测表面的边缘的距离将所述触点标识为拇指。



1. 一种包括触摸感测表面的计算设备的方法,所述方法包括:  
从所述触摸感测表面获取接近图像;  
分割所述接近图像以标识包括所述接近图像的边缘列中的一个或多个触摸节点的至少第一块;  
确定所述边缘列中的所述第一块的所述一个或多个触摸节点的边缘矩形;  
确定相邻列中的所述第一块的一个或多个触摸节点的相邻矩形,所述相邻列与所述边缘列相邻;  
将所述边缘列中的所述第一块的所述一个或多个触摸节点复制到所述接近图像中的边界列,包括基于所述相邻矩形和所述边缘矩形的外推来使所述边界列中的所述已复制的触摸节点偏移;以及  
将椭圆拟合至包括所述边界列中的所述一个或多个触摸节点的所述第一块。
2. 根据权利要求 1 所述的方法,其中基于所述相邻矩形和所述边缘矩形的所述外推来使所述已复制的触摸节点偏移包括从所述边界矩形减去所述相邻矩形以获得位移值并基于所述位移值使所述已复制的触摸节点移位。
3. 根据权利要求 2 所述的方法,其中所述位移值为非整数值,并且基于所述位移值使所述已复制的触摸节点移位包括基于所述位移值将所述已复制的触摸节点内插在所述边界列中。
4. 根据权利要求 1 所述的方法,还包括:  
对所述边缘列中的所述第一块的所述一个或多个触摸节点求和以获得边缘和;  
对所述相邻列中的所述第一块的所述一个或多个触摸节点求和以获得相邻和;以及  
基于所述边缘和以及所述相邻和的外推来缩放所述边界列中的所述触摸节点。
5. 根据权利要求 4 所述的方法,其中所述边缘和以及所述相邻和的所述外推包括线性外推和逻辑外推中的一者。
6. 一种包括触摸感测表面的计算设备的方法,所述方法包括:  
获得所述触摸感测表面上或所述触摸感测表面附近的触点的长轴;  
获得所述触点与所述触摸感测表面的边缘的距离;  
基于拇指轮廓,确定所获得的距离处的拇指触点的参考长轴;以及  
基于所述参考长轴和所述触点的长轴将所述触点标识为拇指。
7. 根据权利要求 6 所述的方法,还包括从所述触点的所述长轴减去所述参考长轴以获得轴差;  
其中基于所述参考长轴和所述触点的所述长轴将所述触点标识为拇指包括基于所述轴差将所述触点标识为拇指。
8. 根据权利要求 7 所述的方法,还包括将所述轴差添加至来自先前时间步长的轴差和;  
其中基于所述轴差将所述触点标识为拇指包括基于所述轴差和将所述触点标识为拇指。
9. 根据权利要求 8 所述的方法,还包括确定所述轴差和为正;  
其中基于所述轴差和将所述触点标识为拇指包括基于所述确定所述轴差和为正而将所述触点标识为拇指。

10. 根据权利要求 6 所述的方法,其中在超过从按下所述触点所测量的时间阈值之后将所述触点标识为拇指。

11. 根据权利要求 6 所述的方法,其中在所述触点已达到与所述触摸感测表面的所述边缘的距离阈值之后将所述触点标识为拇指。

12. 一种非暂态计算机可读介质,所述计算机可读介质包含指令,当所述指令被执行时,执行包括触摸感测表面的计算设备的方法,所述方法包括:

从所述触摸感测表面获取接近图像;

分割所述接近图像以标识包括所述接近图像的边缘列中的一个或多个触摸节点的至少第一块;

确定所述边缘列中的所述第一块的所述一个或多个触摸节点的边缘矩形;

确定相邻列中的所述第一块的一个或多个触摸节点的相邻矩形,所述相邻列与所述边缘列相邻;

将所述边缘列中的所述第一块的所述一个或多个触摸节点复制到所述接近图像中的边界列,包括基于所述相邻矩形和所述边缘矩形的外推来使所述边界列中的所述已复制的触摸节点偏移;以及

将椭圆拟合至包括所述边界列中的所述一个或多个触摸节点的所述第一块。

13. 根据权利要求 12 所述的非暂态计算机可读介质,其中基于所述相邻矩形和所述边缘矩形的所述外推来使所述已复制的触摸节点偏移包括从所述边界矩形减去所述相邻矩形以获得位移值并基于所述位移值使所述已复制的触摸节点移位。

14. 根据权利要求 13 所述的非暂态计算机可读介质,其中所述位移值为非整数值,并且基于所述位移值使所述已复制的触摸节点移位包括基于所述位移值将所述已复制的触摸节点内插在所述边界列中。

15. 根据权利要求 12 所述的非暂态计算机可读介质,所述方法还包括:

对所述边缘列中的所述第一块的所述一个或多个触摸节点求和以获得边缘和;

对所述相邻列中的所述第一块的所述一个或多个触摸节点求和以获得相邻和;以及

基于所述边缘和以及所述相邻和的外推来缩放所述边界列中的所述触摸节点。

16. 根据权利要求 15 所述的非暂态计算机可读介质,其中所述边缘和以及所述相邻和的所述外推包括线性外推和逻辑外推中的一者。

17. 一种非暂态计算机可读介质,所述计算机可读介质包含指令,当所述指令被执行时,执行包括触摸感测表面的计算设备的方法,所述方法包括:

获得所述触摸感测表面上或所述触摸感测表面附近的触点的长轴;

获得所述触点与所述触摸感测表面的边缘的距离;

基于拇指轮廓,确定所获得的距离处的拇指触点的参考长轴;以及

基于所述参考长轴和所述触点的长轴将所述触点标识为拇指。

18. 根据权利要求 17 所述的非暂态计算机可读介质,所述方法还包括从所述触点的所述长轴减去所述参考长轴以获得轴差;

其中基于所述参考长轴和所述触点的所述长轴将所述触点标识为拇指包括基于所述轴差将所述触点标识为拇指。

19. 根据权利要求 18 所述的非暂态计算机可读介质,所述方法还包括将所述轴差添加

至来自先前时间步长的轴差和；

其中基于所述轴差将所述触点标识为拇指包括基于所述轴差和将所述触点标识为拇指。

20. 根据权利要求 19 所述的非暂态计算机可读介质,所述方法还包括确定所述轴差和为正；

其中基于所述轴差和将所述触点标识为拇指包括基于所述确定所述轴差和为正而将所述触点标识为拇指。

21. 根据权利要求 17 所述的非暂态计算机可读介质,其中在超过从按下所述触点所测量的时间阈值之后将所述触点标识为拇指。

22. 根据权利要求 17 所述的非暂态计算机可读介质,其中在所述触点已达到与所述触摸感测表面的所述边缘的距离阈值之后将所述触点标识为拇指。

23. 一种电子设备,包括：

触摸感测表面；

用于执行指令的处理器；和

与所述处理器耦接以存储指令的存储器,当所述指令由所述处理器执行时,使得所述处理器执行操作以生成应用编程接口(API),所述API允许API调用组件执行所述电子设备的方法,所述方法包括：

从所述触摸感测表面获取接近图像；

分割所述接近图像以标识包括所述接近图像的边缘列中的一个或多个触摸节点的至少第一块；

确定所述边缘列中的所述第一块的所述一个或多个触摸节点的边缘矩形；

确定相邻列中的所述第一块的一个或多个触摸节点的相邻矩形,所述相邻列与所述边缘列相邻；

将所述边缘列中的所述第一块的所述一个或多个触摸节点复制到所述接近图像中的边界列,包括基于所述相邻矩形和所述边缘矩形的外推来使所述边界列中的所述已复制的触摸节点偏移；以及

将椭圆拟合至包括所述边界列中的所述一个或多个触摸节点的所述第一块。

24. 根据权利要求 23 所述的电子设备,其中基于所述相邻矩形和所述边缘矩形的所述外推来使所述已复制的触摸节点偏移包括从所述边界矩形减去所述相邻矩形以获得位移值并基于所述位移值使所述已复制的触摸节点移位。

25. 根据权利要求 24 所述的电子设备,其中所述位移值为非整数值,并且基于所述位移值使所述已复制的触摸节点移位包括基于所述位移值将所述已复制的触摸节点内插在所述边界列中。

26. 根据权利要求 23 所述的电子设备,所述方法还包括：

对所述边缘列中的所述第一块的所述一个或多个触摸节点求和以获得边缘和；

对所述相邻列中的所述第一块的所述一个或多个触摸节点求和以获得相邻和；以及

基于所述边缘和以及所述相邻和的外推来缩放所述边界列中的触摸节点。

27. 根据权利要求 26 所述的电子设备,其中所述边缘和以及所述相邻和的所述外推包括线性外推和逻辑外推中的一者。

28. 一种电子设备,包括:

触摸感测表面;

用于执行指令的处理器;和

与所述处理器耦接以存储指令的存储器,当所述指令由所述处理器执行时,使得所述处理器执行操作以生成应用编程接口(API),所述API允许API调用组件执行所述电子设备的方法,所述方法包括:

获得所述触摸感测表面上或所述触摸感测表面附近的触点的长轴;

获得所述触点与所述触摸感测表面的边缘的距离;

基于拇指轮廓,确定所获得的距离处的拇指触点的参考长轴;以及

基于所述参考长轴和所述触点的所述长轴将所述触点标识为拇指。

29. 根据权利要求 28 所述的电子设备,所述方法还包括从所述触点的所述长轴减去所述参考长轴以获得轴差;

其中基于所述参考长轴和所述触点的所述长轴将所述触点标识为拇指包括基于所述轴差将所述触点标识为拇指。

30. 根据权利要求 29 所述的电子设备,所述方法还包括将所述轴差添加至来自先前时间步长的轴差和;

其中基于所述轴差将所述触点标识为拇指包括基于所述轴差和将所述触点标识为拇指。

31. 根据权利要求 30 所述的电子设备,所述方法还包括确定所述轴差和为正;

其中基于所述轴差和将所述触点标识为拇指包括基于所述确定所述轴差和为正而将所述触点标识为拇指。

32. 根据权利要求 28 所述的电子设备,其中在超过从按下所述触点所测量的时间阈值之后将所述触点标识为拇指。

33. 根据权利要求 28 所述的电子设备,其中在所述触点已达到与所述触摸感测表面的所述边缘的距离阈值之后将所述触点标识为拇指。

## 边框边缘处的触摸检测

### 技术领域

[0001] 本发明大体上涉及具有触摸感测能力的电子设备。

### 背景技术

[0002] 在触摸感测系统中,可从触摸感测表面获取接近图像,并且该图像可分割成多个块,每个块对应于触摸感测表面上或触摸感测表面附近的触点。一个椭圆可拟合至多个块中的每个块。椭圆的参数可被用来标识触点块并生成输入。然而,当触点块位于接近图像的边缘时,拟合的椭圆可能不会准确地表示感测到的触摸物体,其中一些触摸物体可能经过了触摸感测表面的边缘。

### 发明内容

[0003] 本发明涉及一种外推接近信息以生成触摸节点(还称为触摸像素)的边界列或行并随后将椭圆拟合到包括所述外推的边界触摸节点的触点块的方法。另外,可基于触点的长轴和所述触点到所述触摸感测表面的边缘的距离将所述触点标识为拇指。

### 附图说明

[0004] 图 1A 和 1B 示出了根据本发明的示例的包括外推的边界节点的示例性接近图像。

[0005] 图 2 示出了根据本发明的示例的外推边界节点以拟合椭圆的示例性方法。

[0006] 图 3 示出了根据本发明的示例的触点的长轴对触点到触摸感测表面的边缘的距离的图。

[0007] 图 4 示出了根据本发明的示例的将触点标识为拇指的示例性方法。

[0008] 图 5 是示出了可用于本发明的一些实施例中的示例性 API 架构的框图。

[0009] 图 6 示出了根据本发明的示例的 API 的示例性软件栈。

[0010] 图 7 是根据本发明的示例的示出了触摸屏和设备的其它组件之间的示例性交互的框图。

[0011] 图 8 是根据本发明的示例的示出了可以在任何便携式或非便携式设备内体现的系统架构的示例的框图。

### 具体实施方式

[0012] 在以下对示例的描述中将引用附图,附图形成以下描述的一部分并且在附图中以举例方式示出了可实施的具体示例。应当理解,在不脱离所公开的示例的范围的情况下,可使用其它示例并且可进行结构性变更。

[0013] 尽管本文中公开的示例主要从电容式触摸感测和接近感测方面进行描述和说明,但应理解,示例并不受此限制,而是可适用于其它触摸感测技术,包括仅感测力和/或使用电阻式触摸感测的那些技术。例如,如本文中所论述的“接近图像”可以是电容的图像、电阻的图像和/或力的图像,以及其它可能性。另外,尽管示例主要从列的角度论述,但本文

中所公开的方法还可从行的角度使用。

[0014] 在触摸感测系统中,可从触摸感测表面获取接近图像,该图像可分割成多个块,每个块对应于触摸感测表面上或触摸感测表面附近的触点。一个椭圆可拟合至多个块中的每个块。椭圆的参数可被用来标识触点块并生成输入。例如,可基于拟合椭圆的长轴的长度将触点块标识为拇指。然而,当触点块位于接近图像的边缘时,拟合的椭圆可能不会准确地表示感测到的触摸物体,其中一些触摸物体可能经过了触摸感测表面的边缘。

[0015] 图 1A 和 1B 示出了一种外推接近信息以生成触摸节点(还称为触摸像素)的边界列或行并随后将椭圆拟合至包括外推的边界触摸节点的触点块的方法。

[0016] 在图 1A 中,示出了拇指的轮廓,并在边缘列 C1 和相邻列 C2 中示出了与该拇指对应的触点块的触摸节点。可分别计算并外推列 C1 和 C2 的矩心  $y_1$  和  $y_2$  以确定  $y_0$ ,即边界列 C0 的外推矩心。拇指块的列 C1 中的触摸节点被复制到列 C0 中并移位,使得矩心在  $y_0$  处。另外,可分别计算并外推拇指块的列 C1 和 C2 中的触摸节点的和,即和 1 及和 2,以确定和 0,并可基于外推的和 0 适当地缩放复制到列 C0 的触摸节点。在一些示例中,还可缩减(在该情况下是 1/3)列 C0 中的触摸节点以降低外推中的不确定性。

[0017] 在图 1B 中,从电容的接近图像方面示出并描述了算法。也就是说,每个触摸节点是表示触点到该触摸节点的接近度的电容值。图 1B 还在 (1) 处示出了覆盖触摸感测表面的边缘的指纹如何会产生误导性接近图像。在 (2a) 至 (4) 处,实际指纹的轮廓被示为虚线。在外推边界像素之后,在 (4) 处估计的实线椭圆更准确地表示在 (1) 处示出的实际指纹。

[0018] 图 1B 还示出了不进行外推的情况下从初始触摸节点估计的普通椭圆 100、从包括通过外推来添加的附加边界节点的触摸节点而估计的外推椭圆 102 以及实际指纹轮廓 104。普通椭圆 100 具有较短的长轴和较长的短轴,而且它的取向点比起实际指纹轮廓 104 的椭圆更靠近接近图像的上方。通过对比,外推椭圆 102 的长轴比普通椭圆 100 的长,而且它的取向更接近于实际指纹轮廓 104 的取向。可以看出,包括外推边界节点允许具有参数的估计椭圆更接近感测到的触摸物体。

[0019] 图 2 示出了根据本发明的示例的外推边界节点以拟合椭圆的示例性方法。可以获取接近图像并将其分割成多个块(200)。多个块中的第一块可包括接近图像的边缘列中的一个或多个触摸节点。另外,第一块可包括与边缘列相邻的列中的一个或多个触摸节点。

[0020] 可确定边缘列中的第一块的一个或多个触摸节点的边缘列矩心(202)。矩心计算类似于质心计算。例如,矩心可被计算为一个或多个触摸节点的位置的加权平均值,每个位置通过该位置处的触摸节点的相对接近值来加权。可通过类似方式确定相邻列中的第一块的一个或多个触摸节点的相邻列矩心(204)。

[0021] 可通过边缘列矩心和相邻列矩心的外推来使边缘列中的第一块的一个或多个触摸节点复制到边界列并偏移(206)。在一些示例中,可通过从边缘列矩心减去相邻列矩心以获得位移值来外推矩心。随后可基于位移值使复制到边界列的触摸节点移位。在一些示例中,位移值可为整数或可四舍五入至整数,使得每个触摸节点可基于位移值而简单地移位。

[0022] 在一些示例中,位移值可以不是整数,并且使边界列中的已复制的触摸节点移位包括基于位移值外推触摸节点。例如,如果位移值为 0.6,则对于每个触摸节点,触摸节点的接近值的 60%可向上移位至下一触摸节点,接近值的 40%可保留在触摸节点中。剩余的 40%可被添加到向上移位的 60%的下方触摸节点。在一些示例中,可使用基于非整位移值

来外推触摸节点的其它方法。

[0023] 可对边缘列中的第一块的一个或多个触摸节点求和以获得边缘和。这可包括对一个或多个触摸节点的接近值求和。以类似的方式,可对相邻列中的第一块的一个或多个触摸节点求和以获得相邻和。随后可基于边缘和以及相邻和的外推来缩放边界列中的触摸节点(208)。可缩放边界列中的触摸节点,使得边界列中的触摸节点的接近值的和等于边缘和以及相邻和的外推。在一些示例中,外推可以为相邻和至边缘和的轨迹的线性外推。在其它示例中,外推可以为相邻和至边缘和的轨迹的逻辑外推,例如基于双弯曲函数的外推。

[0024] 一个椭圆可拟合至包括边界列的触摸节点的第一块(210)。在一些示例中,拟合椭圆可包括确定椭圆的参数,例如椭圆的长轴、短轴和取向角。

[0025] 如上所论述,可基于拟合至触点的椭圆的长轴长度将该触点标识为拇指。然而,覆盖触摸感测表面的边缘的触点可具有较短的长轴,因此该触点可能无法被标识为拇指,直至大多数触点移动到触摸感测表面上。

[0026] 图3示出了触点的长轴对与触摸感测表面的边缘的距离的图。当触点正位于触摸感测表面的边缘时,触点的长轴非常短,但是当触点从边缘移走并且更多的触点触摸所述表面时,长轴的长度增加,直至该长度稳定至完整触点的长轴的实际长度。虚线300表示可能的阈值长轴长度,在该长度之上的触点可标识为拇指。曲线302表示长轴和与普通手指触点的边缘的距离之间的关系。曲线306表示长轴和与拇指触点的边缘的距离之间的关系。

[0027] 曲线304表示可用来将触点标识为拇指的参考拇指轮廓。可在一系列时间步长内针对触点来监控长轴以及与边缘的距离,并且可整合并累积触点曲线和参考拇指轮廓304之间的区域。如果整合的值保持为正,如将通过曲线306所示的那样,则所关联的触点可被标识为拇指。如果整合的值变为负,如将通过曲线302所示的那样,则可以确定所关联的触点不是拇指。该方法相当稳健,因为即使非拇指触点在接近边缘时具有非常大的长轴,当从边缘移开时,长轴将缓慢增大,而且将累积足够的负区域,使得触点将不被标识为拇指。在一些示例中,可累积整合的值,且触点可能不被标识,直至超过时间阈值或超过与边缘的距离阈值,以及其它可能性。

[0028] 图4示出了一种基于触点的长轴和触点与触摸感测表面的边缘的距离而将触点标识为拇指的示例性方法。可获得触点的长轴以及触点与触摸感测表面的边缘的距离(400)。

[0029] 基于拇指轮廓,可在获得的距离处确定拇指触点的参考长轴(402)。拇指轮廓可能只是距离至长轴值的映射,如图3所示。确定参考长轴可包括基于获得的触点与触摸感测表面的边缘的距离从拇指轮廓获得参考长轴。

[0030] 可基于参考长轴以及触点的长轴将触点标识为拇指(404)。例如,参考长轴以及触点的长轴可被用来整合触点的曲线与拇指轮廓的曲线之间的区域,如参照图3所描述。如果整合的值为正,可将触点标识为拇指。

[0031] 在一些示例中,可从触点的长轴减去参考长轴以获得轴差。在一个示例中,如果轴差为正,则可将触点标识为拇指。在其它示例中,可将轴差添加至来自先前时间步长的轴差和。如果轴差和为正,则可将触点标识为拇指。

[0032] 上面论述的示例可在一个或多个应用编程接口(API)中实现。API是由程序代码



组件或硬件组件（在下文中称为“API 实现组件”）来实现的接口，允许不同的程序代码组件或硬件组件（在下文中称为“API 调用组件”）访问和使用由 API 实现组件提供的一个或多个函数、方法、流程、数据结构、类和 / 或其它服务。API 可定义在 API 调用组件和 API 实现组件之间传递的一个或多个参数。

[0033] 上述特征可作为应用编程接口 (API) 的一部分来实现，API 可允许使用触摸输入作为输入机制来将 API 结合到不同的应用程序（例如，电子制表应用程序）中。API 可允许 API 调用组件的开发者（可以是第三方开发者）利用由 API 实现组件提供的指定特征，例如上面描述的那些特征。可以有一个 API 调用组件或可以有多于一个此类组件。API 可以是计算机系统或程序库提供的源代码接口，以便支持来自应用程序的服务请求。操作系统 (OS) 可以具有多个 API，以允许运行于 OS 上的应用程序调用那些 API 中的一个或多个 API，服务（例如程序库）可具有多个 API，以允许使用服务的应用程序调用那些 API 中的一个或多个 API。可按照在构建应用程序时能够解译或编译的编程语言来指定 API。

[0034] 在一些示例中，API 实现组件可提供多于一个 API，每个 API 提供由 API 实现组件实现的功能的不同视图，或提供访问由 API 实现组件实现的功能的不同方面的不同方面。例如，API 实现组件的一个 API 可提供第一组功能，并可暴露于第三方开发者，API 实现组件的另一个 API 可被隐藏（不暴露）并提供第一组功能的子组，并且还提供另一组功能，例如不在第一组功能中的测试或调试功能。在其它示例中，API 实现组件本身可经由下层 API 调用一个或多个其它组件，因而既是 API 调用组件又是 API 实现组件。

[0035] API 定义在访问和使用 API 实现组件的指定特征时 API 调用组件所使用的语言和参数。例如，API 调用组件通过被 API 暴露的一个或多个 API 调用或引用（例如由函数调用或方法调用实现）来访问 API 实现组件的指定特征，并经由 API 调用或引用来使用参数传递数据和控制信息。API 实现组件可响应于来自 API 调用组件的 API 调用而通过 API 返回值。尽管 API 定义 API 调用的语法和结果（例如，如何引起 API 调用以及 API 调用能干什么），但 API 可不揭示 API 调用如何完成由 API 调用指定的函数。经由调用（API 调用组件）和 API 实现组件之间的一个或多个应用编程接口传输各种 API 调用。传输 API 调用可包括发出、发起、引用、调用、接收、返回或响应函数调用或消息；换言之，传输可通过 API 调用组件或 API 实现组件来描述动作。API 的函数调用或其它引用可通过参数列表或其它结构发送或接收一个或多个参数。参数可以是常数、键、数据结构、对象、对象类、变量、数据类型、指针、数组、列表或指向函数或方法的指针或援引要经由 API 传递的数据或其它项目的另一种方式。

[0036] 此外，数据类型或类可以由 API 提供并由 API 实现组件实现。因此，API 调用组件可利用 API 中提供的定义声明变量、使用指向这种类型或类的指针、使用或实例化这种类型或类的恒定值。

[0037] 通常，可使用 API 来访问由 API 实现组件提供的服务或数据或启动执行由 API 实现组件提供的操作或计算。以举例的方式，API 实现组件和 API 调用组件各自可以是操作系统、库、设备驱动程序、API、应用程序或其它模块（应当理解，API 实现组件和 API 调用组件可以是彼此相同或不同类型的模块）中的任一种。在一些情况下，可至少部分地在固件、微码或其它硬件逻辑中实现 API 实现组件。在一些示例中，API 可以允许客户端程序使用由软件开发工具包 (SDK) 库提供的服务。在其它示例中，应用程序或其它客户端程序可使

用由应用程序框架提供的 API。在这些示例中,应用程序或客户端程序可以将调用并入由 SDK 提供和由 API 提供的函数或方法中,或使用在 SDK 中定义并由 API 提供的数据类型或对象。在这些示例中,应用程序框架可为程序提供主要事件循环,该程序对由框架定义的各种事件作出响应。API 允许应用程序利用应用程序框架来指定事件和对事件的响应。在一些具体实施中,API 调用能够向应用程序报告硬件设备的能力或状态,包括与诸如输入能力和状态、输出能力和状态、处理能力、电源状态、存储容量和状态、通信能力等方面相关的能力或状态,API 可部分地由固件、微码或部分地在硬件组件上执行的其它低电平逻辑实现。

[0038] API 调用组件可以是本地组件(即与 API 实现组件在同一数据处理系统上)或远程组件(即在不同于 API 实现组件的数据处理系统上),所述组件经由网络通过 API 与 API 实现组件通信。应当理解,API 实现组件也可充当 API 调用组件(即,它可以对不同 API 实现组件暴露的 API 进行 API 调用),API 调用组件也可以通过实现暴露于不同 API 调用组件的 API 来充当 API 实现组件。

[0039] API 可允许用不同编程语言编写的多个 API 调用组件来与 API 实现组件通信(从而 API 可包括用于转换 API 实现组件和 API 调用组件之间调用和返回的特征);然而,API 可从具体编程语言方面实现。在一个示例中,API 调用组件可调用来自不同提供商的 API,例如来自 OS 提供商的一组 API 和来自插件提供商的另一组 API,以及来自另一提供商(例如软件库的提供商)或另一组 API 的创建者的另一组 API。

[0040] 图 5 是示出了可用于本发明的一些实施例的示例性 API 架构的框图。如图 5 中所示,API 架构 500 包括实现 API 520 的 API 实现组件 510(例如,操作系统、库、设备驱动程序、API、应用程序、软件或其它模块)。API 520 指定可由 API 调用组件 530 使用的 API 实现组件的一个或多个函数、方法、类、对象、协议、数据结构、格式和/或其它特征。API 520 能够指定至少一个调用约定,该调用约定指定 API 实现组件中的函数如何从 API 调用组件接收参数以及函数如何向 API 调用组件返回结果。API 调用组件 530(例如操作系统、库、设备驱动程序、API、应用程序、软件或其它模块)通过 API 520 进行 API 调用,以访问并使用由 API 520 指定的 API 实现组件 510 的特征。API 实现组件 510 可以响应于 API 调用而通过 API 520 向 API 调用组件 530 返回值。

[0041] 应当理解,API 实现组件 510 可包括未通过 API 520 指定且对于 API 调用组件 530 不可用的附加函数、方法、类、数据结构和/或其它特征。应理解,API 调用组件 530 可与 API 实现组件 510 在同一系统上,或者可远程定位并通过网络来使用 API 520 访问 API 实现组件 510。尽管图 5 示出了单个 API 调用组件 530 与 API 520 交互,但应理解,可用不同语言(或相同语言)编写的与 API 调用组件 530 不同的其它 API 调用组件可使用 API 520。

[0042] API 实现组件 510、API 520 和 API 调用组件 530 可存储在非暂态机器可读存储介质中,其包括用于以机器(例如计算机或其它数据处理系统)可读的形式存储信息的任何机构。例如,机器可读介质包括磁盘、光盘、随机存取存储器、只读存储器、闪存设备等。

[0043] 在图 6 所示的示例性软件栈中,应用程序可使用若干服务 API 调用服务 A 或 B,以及使用若干 OS API 调用操作系统(OS)。服务 A 和 B 可使用若干 OS API 对 OS 进行调用。

[0044] 注意,服务 2 具有两个 API,其中一个(服务 2API 1)从应用程序 1 接收调用并返回值,另一个(服务 2API 2)从应用程序 2 接收调用并返回值。服务 1(例如,可以是软件库)对 OS API 1 进行调用并接收返回的值,服务 2(例如,可以是软件库)对 OS API 1 和

OS API 2 两者进行调用并接收返回的值。应用程序 2 对 OS API 2 进行调用并接收返回的值。

[0045] 图 7 是示出了触摸屏和设备的其它组件之间的示例性交互的框图。所述示例可包括触摸 I/O 设备 1001, 该触摸 I/O 设备可经由有线或无线通信信道 1002 来接收用于与计算系统 1003 进行交互的触摸输入。触控 I/O 设备 1001 可用于代替其它输入设备诸如键盘、鼠标等或与其它输入设备结合而向计算系统 1003 提供用户输入。一个或多个触控 I/O 设备 1001 可用于向计算系统 1003 提供用户输入。触摸 I/O 设备 1001 可以是计算系统 1003 的整体部分 (例如, 智能手机或平板 PC 上的触摸屏) 或可与计算系统 1003 分开。

[0046] 触摸 I/O 设备 1001 可包括触摸感测面板, 其是完全透明或部分透明的、半透明的、非透明的、不透明的或它们的任何组合。触摸 I/O 设备 1001 可被实现为触摸屏、触控板、充当触控板的触摸屏 (例如, 替代膝上型电脑的触控板的触摸屏)、与任何其它输入设备组合或结合的触摸屏或触控板 (例如, 设置于键盘上的触摸屏或触控板) 或具有用于接收触摸输入的触摸感测表面的任何多维物体。

[0047] 在一个示例中, 实现为触摸屏的触摸 I/O 设备 1001 可包括部分或完全定位于显示器的至少一部分上方的透明和 / 或半透明的触摸感测面板。根据该示例, 触摸 I/O 设备 1001 用于显示从计算系统 1003 (和 / 或另一个来源) 传输的图形数据, 并且还用于接收用户输入。在其它示例中, 触摸 I/O 设备 1001 可被实现为集成触摸屏, 其中触摸感测组件 / 设备与显示组件 / 设备是一体的。在其它示例中, 可将触摸屏用作补充或附加的显示屏, 以用于显示主显示器的补充或相同图形数据, 并用于接收触摸输入。

[0048] 触摸 I/O 设备 1001 可被配置为基于电容、电阻、光学、声学、感应、机械、化学测量来检测设备 1001 上一个或多个触摸或接近触摸的位置, 或者检测可相对于设备 1001 附近发生的一次或多次触摸或接近触摸来测量的任何现象。可以使用软件、硬件、固件或它们的任何组合来处理所检测到触摸的测量结果, 以便标识和追踪一个或多个手势。手势可对应于触摸 I/O 设备 1001 上的静止的或非静止的、单个或多个触摸或接近触摸。可通过基本上同时、连续或依次地以特定方式在触摸 I/O 设备 1001 上移动一个或多个手指或其它物体来执行手势, 如轻敲、按压、摇动、摩擦、扭转、改变取向、用不同压力按压等。手势可以通过但不限于以下动作来表征: 在手指之间或用任何一个或多个其它手指进行的夹捏、滑动、轻扫、旋转、屈曲、拖拽或轻敲动作。一个或多个用户可用一只或多只手或它们的任何组合来执行单个手势。

[0049] 计算系统 1003 可利用图形数据来驱动显示器以显示图形用户界面 (GUI)。GUI 可被配置为经由触摸 I/O 设备 1001 接收触摸输入。被实现为触摸屏的触摸 I/O 设备 1001 可显示 GUI。另选地, GUI 可被显示在与触控 I/O 设备 1001 分开的显示器上。GUI 可包括在界面内的特定位置处显示的图形元素。图形元素可包括但不限于各种显示的虚拟输入设备, 包括虚拟滚轮、虚拟键盘、虚拟旋钮、虚拟按钮、任何虚拟 UI 等。用户可在触摸 I/O 设备 1001 上的可能与 GUI 的图形元素相关联的一个或多个特定位置处执行手势。在其它示例中, 用户可在与 GUI 的图形元素的位置无关的一个或多个位置处执行手势。在触摸 I/O 设备 1001 上执行的手势可直接或间接地操纵、控制、修改、移动、致动、启动或一般性地影响 GUI 内的图形元素, 该图形元素诸如是光标、图标、媒体文件、列表、文本、所有或部分图像等。例如, 就触摸屏而言, 用户可通过在触摸屏上的图形元素上方执行手势来与图形元素直接进行交互。

互。或者,触控板一般提供间接交互。手势还可影响未被显示的 GUI 元素(例如,使得用户界面显现)或可影响计算系统 1003 内的其它动作(例如,影响 GUI 的状态或模式、应用程序或操作系统)。与所显示的光标结合,手势可在或可不在触摸 I/O 设备 1001 上执行。例如,在触控板上执行手势的情况下,可以在显示屏或触摸屏上显示光标(或指针),并且可经由触控板上的触摸输入来控制光标以与显示屏上的图形对象交互。在直接在触摸屏上执行手势的其它示例中,不管是否有光标或指针显示在触摸屏上,用户都可与触摸屏上的对象直接进行交互。

[0050] 可响应于或基于触摸 I/O 设备 1001 上的触摸或接近触摸而经由通信信道 1002 向用户提供反馈。可通过光学、机械、电气、嗅觉、声学等或它们的任何组合并且以可变或不可变方式传输反馈。

[0051] 现在关注可在任何便携式或非便携式设备内实现的系统架构的示例,便携式或非便携式设备包括但不限于通信设备(例如移动电话、智能电话)、多媒体设备(例如 MP3 播放器、电视、收音机)、便携式或手持式电脑(例如平板电脑、上网本、膝上型电脑)、台式计算机、一体台式计算机、外围设备或任何其它适于包括系统架构 2000 的系统或设备,包括这些类型设备的两个或更多个的组合。图 8 是系统 2000 的一个示例的框图,该系统 2000 通常包括一个或多个计算机可读介质 2001、处理系统 2004、I/O 子系统 2006、射频(RF)电路 2008 和音频电路 2011。这些组件可通过一根或多根通信总线或信号线 2003 来耦接。

[0052] 应当理解,图 8 中所示的架构仅为系统 2000 的一个示例性架构,并且系统 2000 可具有比图示更多或更少的组件、或组件的不同配置。图 8 中所示的各种组件可在硬件、软件、固件或它们的任何组合(包括一个或多个信号处理和/或专用集成电路)中实施。

[0053] 射频电路 2008 用于通过无线链路或网络向一个或多个其它设备发送和接收信息,并且包括用于执行该功能的熟知的电路。射频电路 2008 和音频电路 2010 经由外围设备接口 2016 被耦接到处理系统 2004。接口 2016 包括用于在外围设备和处理系统 2004 之间建立并维持通信的各种已知组件。音频电路 2010 耦接到音频扬声器 2050 和麦克风 2052,并且包括用于处理从接口 2016 所接收的语音信号的已知电路以使得用户能够与其它用户进行实时通信。在一些示例中,音频电路 2010 包括耳机接口(未示出)。

[0054] 外围设备接口 2016 将系统的输入外围设备和输出外围设备耦接到处理器 2018 和计算机可读介质 2001。一个或多个处理器 2018 经由控制器 2020 与一个或多个计算机可读介质 2001 进行通信。计算机可读介质 2001 可以是可存储供一个或多个处理器 2018 使用的代码和/或数据的任何设备或介质。介质 2001 可包括存储器分级结构,包括但不限于高速缓存、主存储器和辅助存储器。可使用 RAM(例如 SRAM、DRAM、DDRAM)、ROM、闪存、磁存储设备和/或光学存储设备(诸如磁盘驱动器、磁带、CD(光盘)和 DVD(数字视频光盘))的任何组合来实现该存储器分级结构。介质 2001 还可包括用于传送指示计算机指令或数据的信息承载信号的传输介质(具有或不具有在其上调制信号的载波)。例如,传输介质可包括通信网络,该通信网络包括但不限于互联网(也称为万维网)、一个或多个内联网、局域网(LAN)、宽局域网(WLAN)、存储区域网(SAN)、城域网(MAN)等。

[0055] 一个或多个处理器 2018 运行存储在介质 2001 中的各种软件组件以执行系统 2000 的各种功能。在一些示例中,软件组件包括操作系统 2022、通信模块(或指令集)2024、触摸处理模块(或指令集)2026、图形模块(或指令集)2028 和一个或多个应用程序(或指

令集)2030。这些模块和以上提及的应用程序中的每一者对应于用于执行上述一种或多种功能以及在本专利申请中所述的方法(例如,本文所述的计算机实现的方法和/或其它信息处理方法)的指令集。这些模块(即指令集)不必被实现为独立的软件程序、过程或模块,并因此在各种示例中可组合或以其它方式重新布置这些模块的各种子组。在一些示例中,介质 2001 可以存储以上所标识的模块和数据结构的子组。此外,介质 2001 可存储上面未描述的附加模块和数据结构。

[0056] 操作系统 2022 包括各种过程、指令集、软件组件和/或驱动器以用于控制和管理一般系统任务(例如,存储器管理、存储设备控制、电源管理等),并且有利于各个硬件组件和软件组件之间的通信。

[0057] 通信模块 2024 有利于通过一个或多个外部端口 2036 或经由射频电路 2008 来与其它设备进行通信,并且包括各种软件组件以用于处理从射频电路 2008 和/或外部端口 2036 所接收的数据。

[0058] 图形模块 2028 包括各种已知的软件组件以用于在显示表面上呈现、以动画方式呈现和显示图形对象。在触摸 I/O 设备 2012 为触摸感测显示器(例如触摸屏)的示例中,图形模块 2028 包括用于在触摸感测显示器上渲染、显示和以动画方式呈现对象的组件。

[0059] 一个或多个应用程序 2030 可包括安装在系统 2000 上的任何应用程序,包括但不限于浏览器、地址簿、联系人列表、电子邮件、即时消息、文字处理、键盘仿真、桌面小程序、支持 JAVA 的应用程序、加密、数字权限管理、语音标识、语音复制、位置确定能力(诸如由全球定位系统(GPS)提供的位置确定能力)、音乐播放器等。

[0060] 触摸处理模块 2026 包括用于执行与触摸 I/O 设备 2012 相关联的各种任务的各种软件组件,包括但不限于通过触摸 I/O 设备控制器 2032 接收和处理从 I/O 设备 2012 接收的触摸输入。

[0061] I/O 子系统 2006 耦接到触摸 I/O 设备 2012 和一个或多个其它 I/O 设备 2014 以用于控制或执行各种功能。触摸 I/O 设备 2012 通过触摸 I/O 设备控制器 2032 与处理系统 2004 通信,该控制器包括用于处理用户触摸输入的各种组件(例如,扫描硬件)。一个或多个其它输入控制器 2034 从/向其它 I/O 设备 2014 接收/发送电信号。其它 I/O 设备 2014 可包括物理按钮、拨号盘、滑动开关、操作杆、键盘、触控板、附加显示屏或它们的任何组合。

[0062] 如果被实现为触摸屏,那么触摸 I/O 设备 2012 在 GUI 中向用户显示视觉输出。视觉输出可包括文本、图形、视频以及它们的任何组合。一些或全部视觉输出可对应于用户界面对象。触摸 I/O 设备 2012 形成从用户接受触摸输入的触摸感测表面。触摸 I/O 设备 2012 和触摸屏控制器 2032(连同介质 2001 中的任何关联模块和/或指令集)检测并跟踪触摸 I/O 设备 2012 上的触摸或接近触摸(以及触摸的任何移动或释放),并将所检测的触摸输入转换成与图形对象诸如一个或多个用户界面对象交互。在将设备 2012 实现为触摸屏的情况下,用户可与显示在触摸屏上的图形对象直接进行交互。或者,在设备 2012 被实现为触摸设备而非触摸屏(例如触控板)的情况下,用户可与实现为 I/O 设备 2014 的独立显示屏上显示的图形对象间接交互。

[0063] 触摸 I/O 设备 2012 可类似于以下美国专利申请中所述的多触摸感测表面: 6,323,846(Westerman 等人)、6,570,557(Westerman 等人)和/或 6,677,932(Westerman), 和/或美国专利公开 2002/0015024A1,它们中的每一者据此以引用方式并入。

[0064] 在触摸 I/O 设备 2012 为触摸屏的示例中,触摸屏可使用 LCD(液晶显示器)技术、LPD(发光聚合物显示器)技术、OLED(有机 LED)或 OEL(有机电致发光),尽管在其它实施例中可使用其它显示技术。

[0065] 可由触摸 I/O 设备 2012 基于用户的触摸输入以及正在显示的内容和 / 或计算系统的一个状态或多个状态提供反馈。可通过光学(例如光信号或所显示的图像)、机械(例如触觉反馈、触摸反馈、力反馈等)、电气(例如电刺激)、嗅觉、声学(例如嘟嘟声等)等或它们的任何组合方式并且以可变或不可变的方式来传输反馈。

[0066] 系统 2000 还包括用于为各种硬件组件供电的电力系统 2044,并且可包括电源管理系统、一个或多个电源、再充电系统、电源故障检测电路、功率转换器或逆变器、电源状态指示器以及通常与便携式设备中的电力的生成、管理和分配相关联的任何其它组件。

[0067] 在一些示例中,外围设备接口 2016、一个或多个处理器 2018 和存储器控制器 2020 可在单个芯片诸如处理系统 2004 上实施。在一些其它示例中,它们可在单独的芯片上实现。

[0068] 本发明的示例可有利于更好地识别和标识触摸感测表面边缘附近的触点块,从而利用具有操作直观且容易的触摸感测表面的电子设备。

[0069] 虽然参照附图对公开的示例进行了全面的描述,但应注意,各种变化和修改对于本领域内的技术人员而言将变得显而易见。应当理解,此类变化和修改被认为包括在由所附权利要求所限定的公开的示例的范围内。

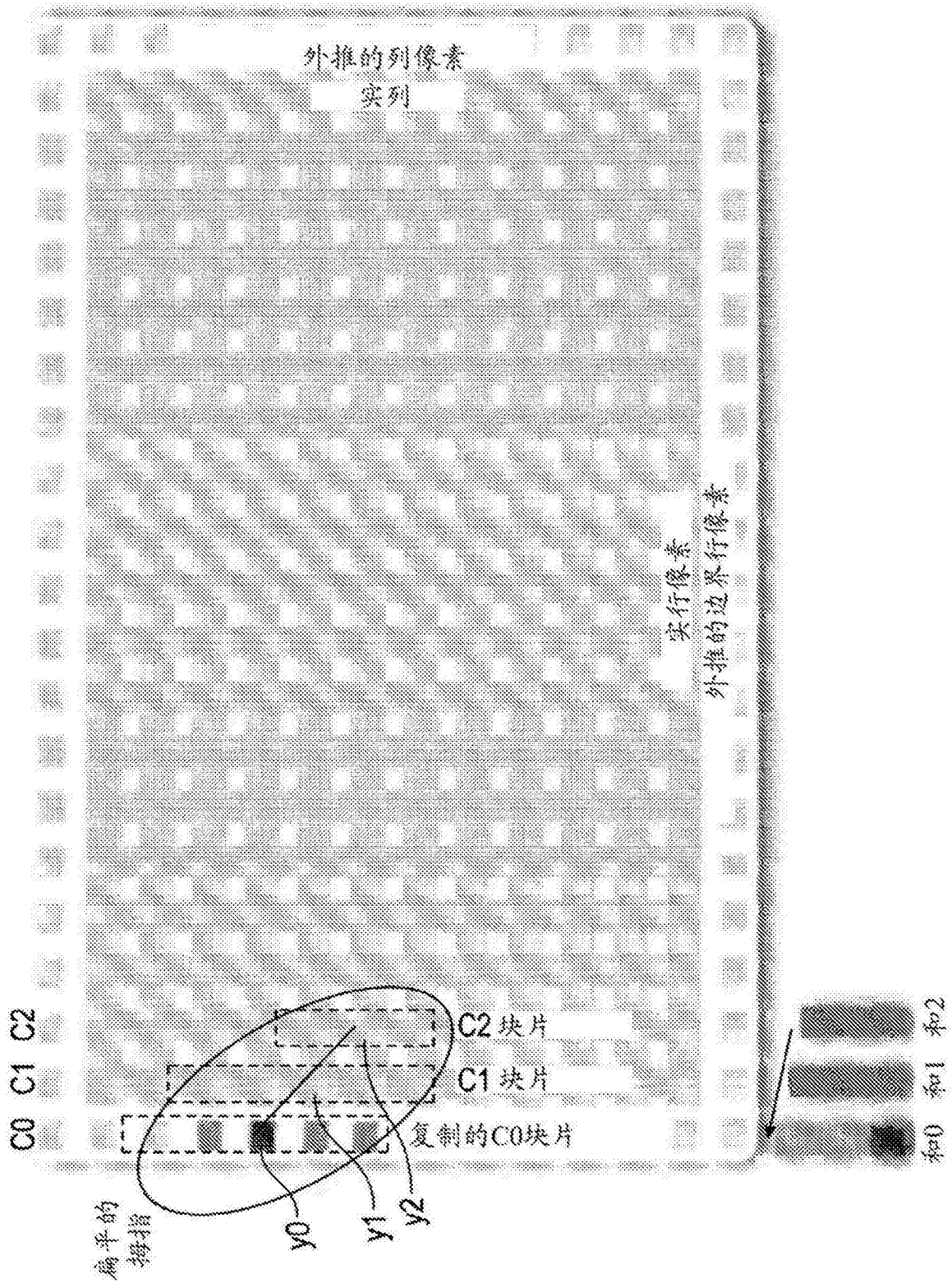


图 1A

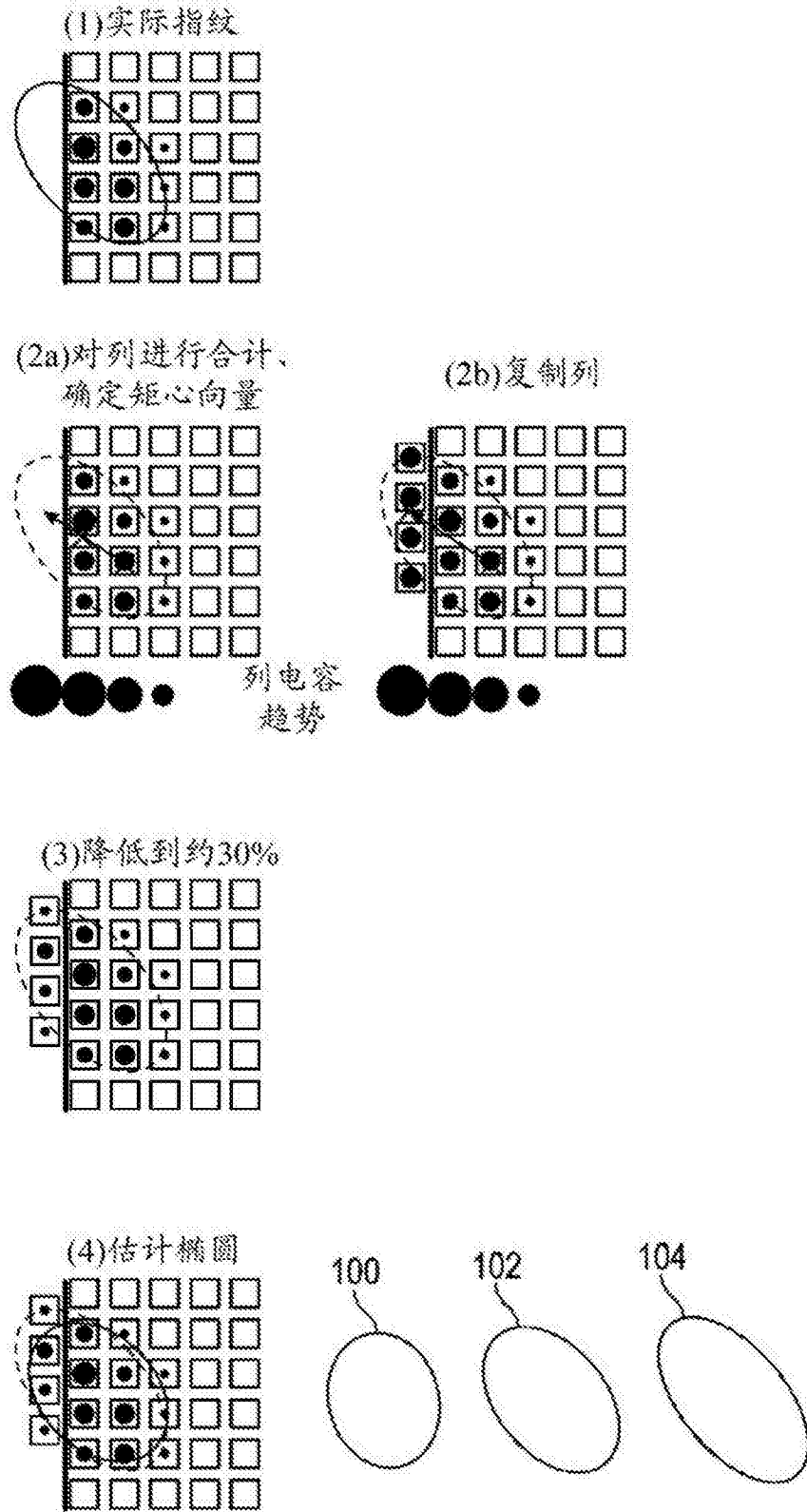


图 1B



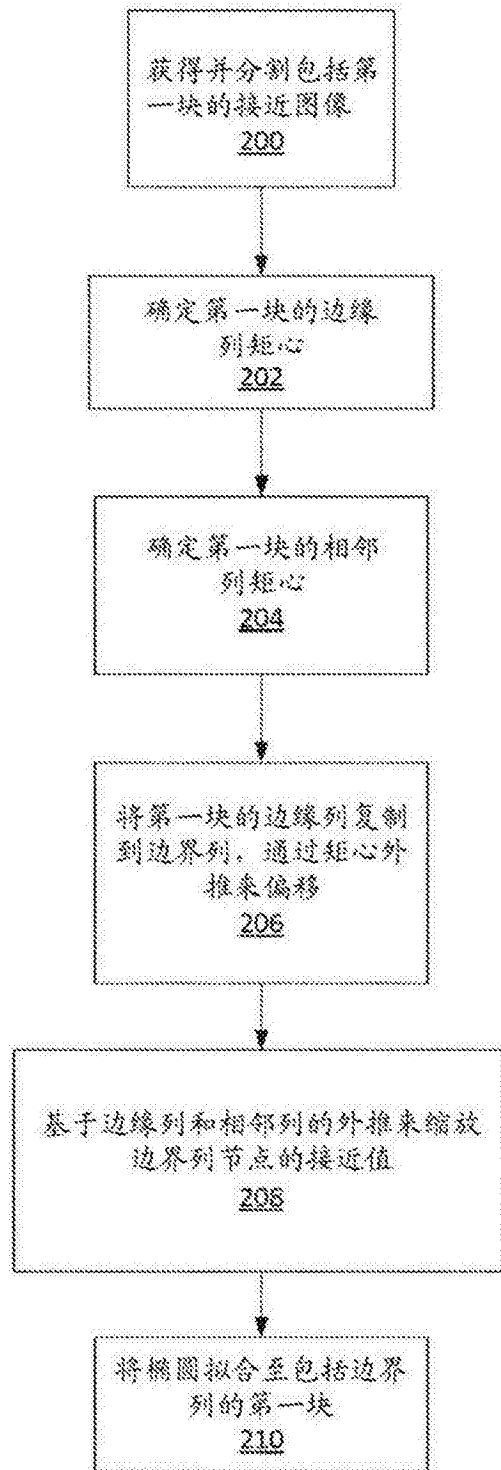


图 2

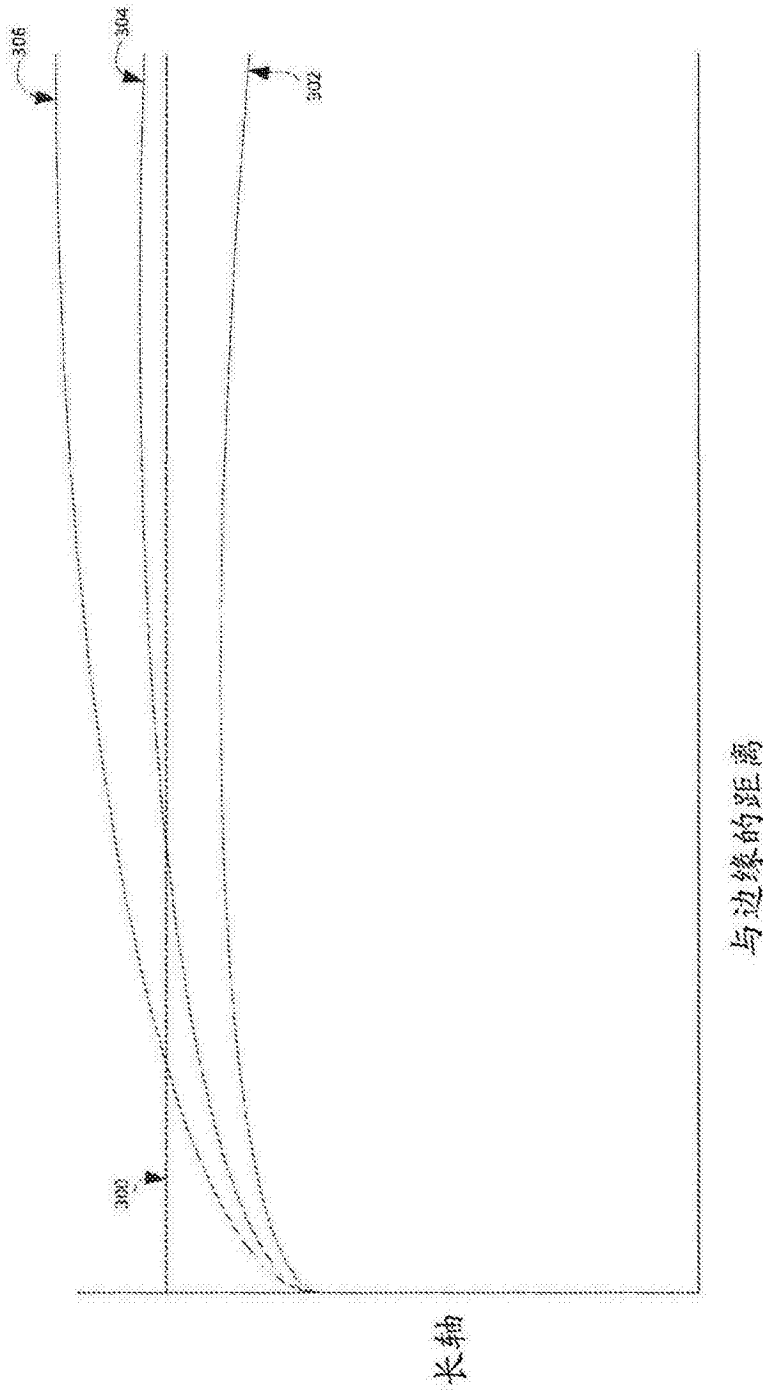


图 3

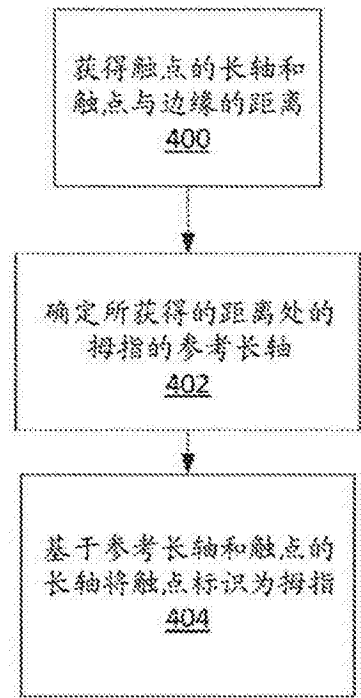


图 4

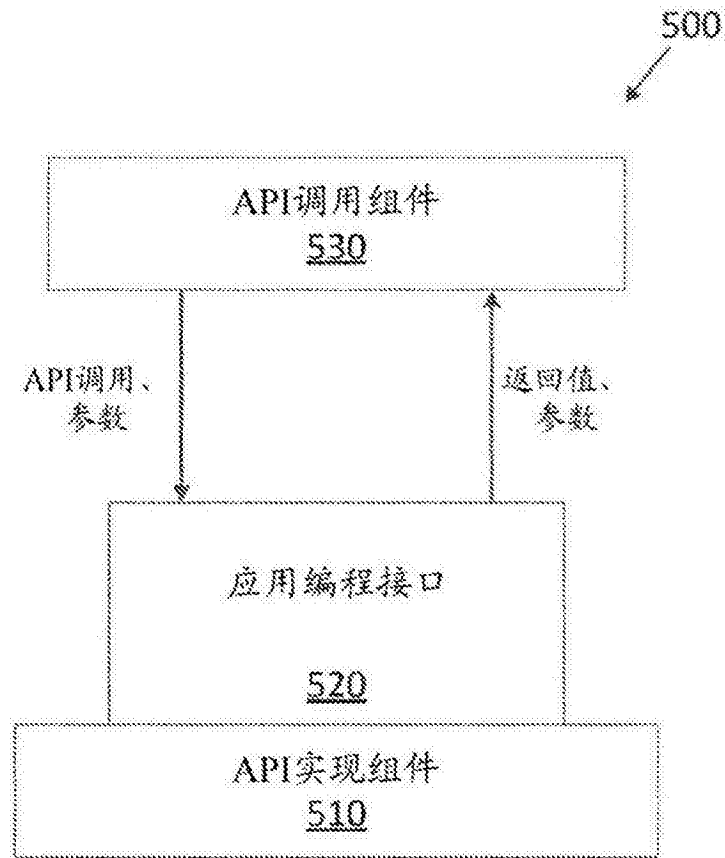


图 5

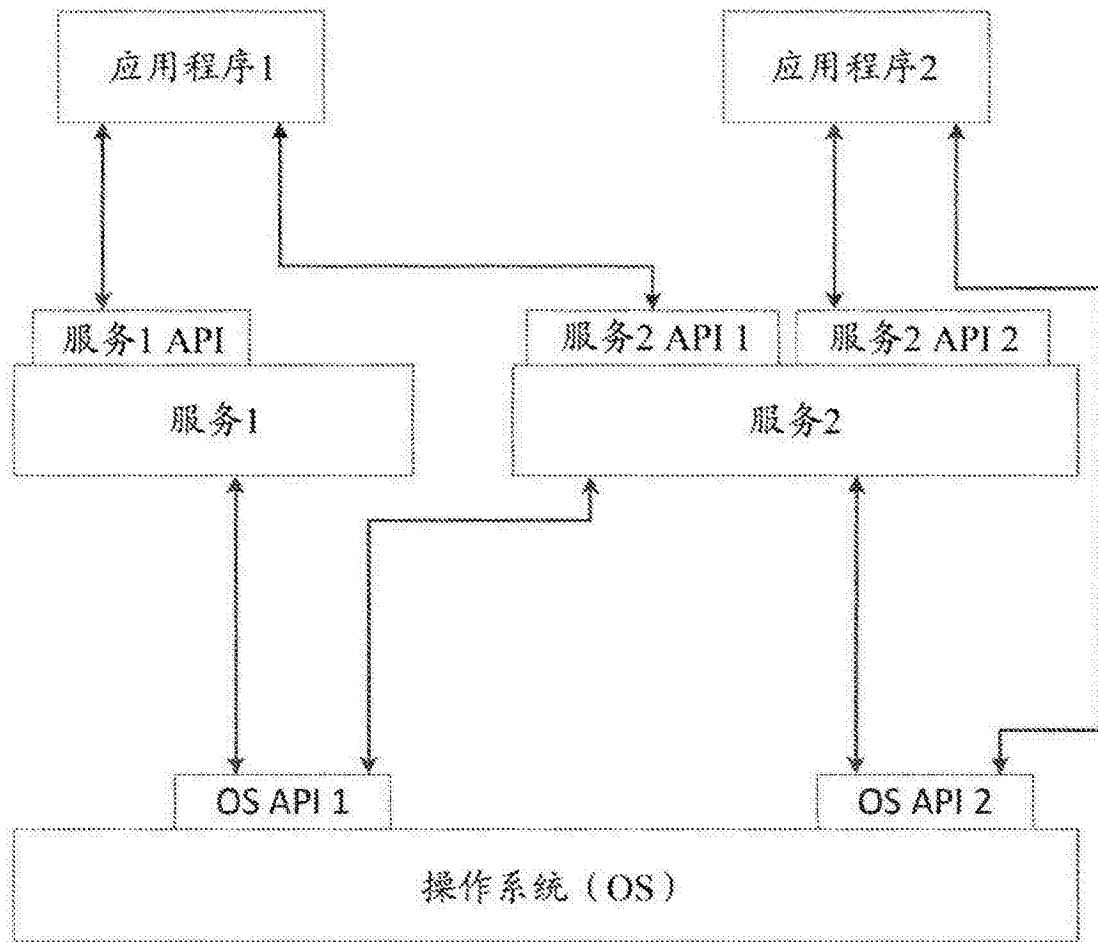


图 6



图 7

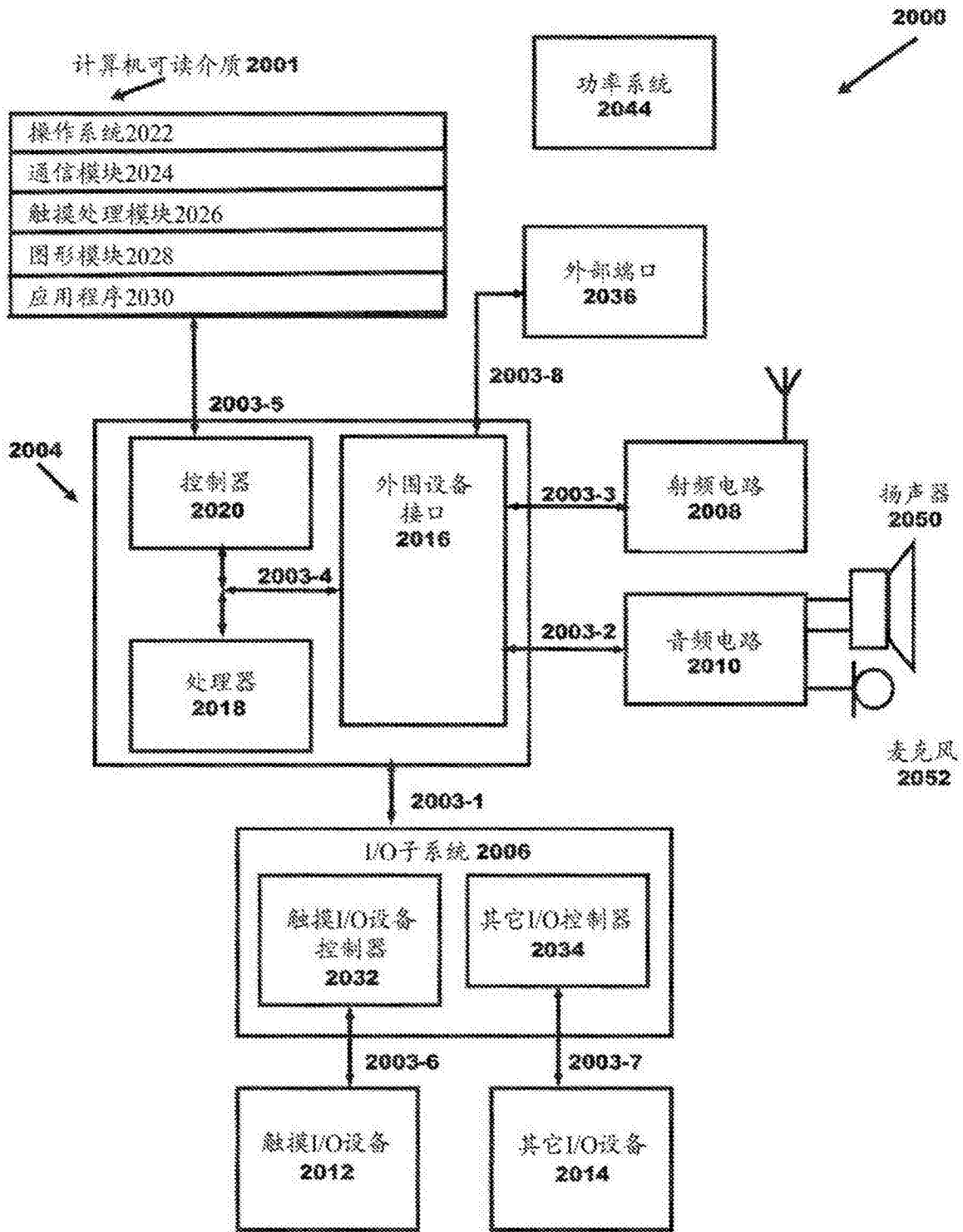


图 8