

[19] 中华人民共和国国家知识产权局

[51] Int. Cl.

G06F 9/44 (2006.01)

G06F 3/00 (2006.01)



[12] 发明专利说明书

专利号 ZL 03801946.9

[45] 授权公告日 2007 年 1 月 3 日

[11] 授权公告号 CN 1293465C

[22] 申请日 2003.8.29 [21] 申请号 03801946.9

[30] 优先权

[32] 2002.9.6 [33] JP [31] 261353/2002

[86] 国际申请 PCT/JP2003/011035 2003.8.29

[87] 国际公布 WO2004/023293 日 2004.3.18

[85] 进入国家阶段日期 2004.7.2

[73] 专利权人 索尼株式会社

地址 日本东京都

[72] 发明人 铃木宏纪

[56] 参考文献

GB2348520A 2000.10.4 G06F3/00

审查员 谢志远

[74] 专利代理机构 北京市柳沈律师事务所

代理人 黄小临 王志森

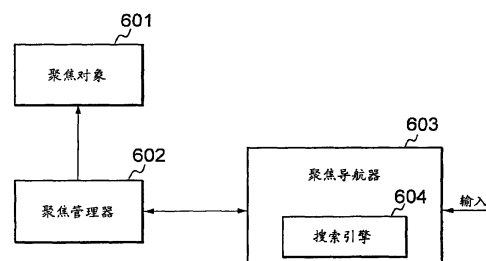
权利要求书 2 页 说明书 27 页 附图 18 页

[54] 发明名称

GUI 显示装置及光标或 GUI 组件动画显示处理方法

[57] 摘要

提供了一种能够根据光标的状态容易地设置和执行各种动画表示处理的装置和方法。设置对应于光标或者光标设置位置组件状态的不同读出图像数据，同时和图形表示程序文件相联系。可以根据光标状态显示不同动画，例如在当光标放置到组件时的聚焦状态中所得到的动画和当光标在组件之间移动时所获得的指示符。在装有 GUI 的设备中，用户将不会看不到光标。实现具有高质量视觉效果 GUI 是可能的。



1.一种 GUI 显示装置，用于在显示器上执行光标或者在其上放置所述光标的 GUI 组件的动画表示，所述 GUI 显示装置特征在于其包括：

 聚焦导航器，用于由搜索引擎根据光标移动指令输入信息执行搜索处理，并确定光标移动目的地；

 聚焦管理器，用于按照由所述聚焦导航器所确定的光标移动目的地指令信息执行光标的移动处理；

 聚焦对象，用于按照与由所述聚焦管理器执行的光标移动同时发生的、所述光标或者其上放置光标的 GUI 组件的状态执行动画显示处理，以执行所述光标或者在其上放置所述光标的 GUI 组件的动画表示，其中所述聚焦对象以如下方式执行所述 GUI 组件的动画表示：朝向移动方向，依次生成具有与所述 GUI 组件的形状类似的形状的图标，并且随着所述光标移动，使后侧的所述光标消失。

2.一种如权利要求 1 所述的 GUI 显示装置，其特征不在于：组成所述动画表示的图像包括具有表示光标移动的可能方向的方向信息的图像，并且特征在于：对于聚焦状态的 GUI 组件，所述聚焦对象执行基于带有所述方向信息的图像的动画显示处理。

3.一种如权利要求 1 所述的 GUI 显示装置，其特征不在于：所述聚焦对象执行由在 GUI 组件上放置光标的聚焦状态和光标在 GUI 组件之间移动的光标移动状态两者之间不同的图像组成的动画显示处理。

4.一种在显示器上执行光标或者在其上放置所述光标的 GUI 组件的动画表示的动画显示处理方法，所述动画显示处理方法特征在于其包括：

 由搜索引擎根据光标移动指令输入信息执行搜索处理，并确定光标移动目的地的步骤；

 按照所确定的光标移动目的地指令信息执行所述光标的移动处理的步骤；以及

 动画显示步骤，用于按照与光标移动同时发生的所述光标或者其上放置光标的 GUI 组件的状态执行动画显示处理，以执行所述光标或者在其上放置所述光标的 GUI 组件的动画表示；以及

 其中在所述动画显示步骤中，以如下方式执行所述 GUI 组件的动画表

示：朝向移动方向，依次生成具有与所述 GUI 组件的形状类似的形状的图标，并且随着所述光标移动，使后侧的所述光标消失。

5.如权利要求 4 所述的动画显示处理方法，其特征在于：组成所述动画表示的图像包括具有表示光标移动的可能方向的方向信息的图像，并且其特征在于：所述动画显示步骤，对聚焦状态的 GUI 组件，执行基于带有方向信息的所述图像的动画显示处理。

6.如权利要求 4 所述的动画显示处理方法，其特征在于：所述动画显示步骤执行由在所述光标放置在 GUI 组件上时的聚焦状态和所述光标在 GUI 组件之间移动的光标移动状态之间不同的图像组成的动画显示处理。

GUI 显示装置及光标或 GUI 组件动画显示处理方法

技术领域

本发明涉及一种 GUI 应用程序开发支持装置、GUI 显示装置、方法和计算机程序。更具体地讲，本发明涉及用来通过实现在图形用户界面(GUI)中所显示的光标的动画显示，改善视觉效果并可能使得光标位置和光标移动更直观显示的一种 GUI 应用程序开发支持装置、GUI 显示装置、方法和计算机程序。

背景技术

诸如 CRT 和 LCD 的显示器用作 PC、通信终端和各种信息设备的数据显示接口或者数据用户输入/输出接口。对于这样的显示器，将为其设置适合信息设备使用的功能的屏幕图像，按例如图形用户界面(GUI)来显示。例如，将诸如输入按钮排列、用于数据输入字段的设置、当按下按钮时将要实现的处理的各功能设置为 GUI。作为开发用于设置这样的用户界面屏幕的设计和功能的应用程序的编程语言，由太阳微系统公司发布的 Java，得到广泛使用。

用 Java 编写的程序被转换为与计算机/平台无关的 Java 可执行文件格式的字节代码，由 Java 虚拟机(Java VM)读取，并转换为(解释为)机器语言并在各种装置上运行。

在 Java 中，为了在短时间内创建应用程序软件，事先准备好了不要求重新编译的软件组件，并通过组合那些组件来创建程序。在 Java 中，构建了用于使这些组件可复用(Java Bean)的技术。

用例如在显示器上将要显示的诸如窗口、按钮、列表、复合框、文本输入字段等各种组件为单元设置组件。对于这些组件，定义了作为诸如例如按钮形状、位置信息等属性信息的“特性”，作为因为组件动作结果的处理的“方法”，以及更进一步地作为用于当诸如数据到达、中断发生、特性变化、方法调用等的有关组件动作发生时，将所述事件通知给另一个组件

的功能的“事件”。这些“特性”、“方法”和“事件”被当作 Java Bean 中组件的元素。

在 Java 中最小的可执行程序单元称作为“类”，而 Java 程序由一个或者多个类构成。类具有作为数据的变量和作为行动的方法。经过诸如因特网或者 LAN 发送和存储设置为类的这种程序组件到 PC 或者信息设备是可能的，并且在其中存储类文件的所述装置的一方，与平台无关的 Java 虚拟机能够运行在所述类文件中保存的程序。

作为 GUI 的关键元素，有光标和指针。光标或者指针基于来自例如作为用户输入设备的鼠标、键盘、遥控的箭头键等的输入信息移动，并且指示有效的输入位置信息。在下面的描述中，光标将当作包含包括光标和指针概念的术语。在传统的设备中光标位置的显示通常由表示光标的条、高亮指定区域、使图标闪烁等来实现。

在传统设备中的光标显示处理，具有以下结构：设置存储与所述光标对应的显示图像数据的文件，并且执行为所述文件定义的诸如重复的闪烁图案或者指定的图像数据显示的显示模式。然而，这样的光标图像大多数是和特定图像数据相联系的，并且是固定的，与光标的状态无关。因此，问题在于用户区分例如光标是否是静止或者移动之间的差别是困难的。

此外，能够将各种信息应用认为是其中可以实现 GUI 的设备，但是这些设备中几乎没有象 PC 的鼠标那样能够执行用于在所有方向上自由移动光标的指令的输入装置。并且，它们中的多数只能通过通过遥控中使用的例如上、下、左和右四个方向的移动指令键和回车键的组合来实现光标的移动控制。

当实现基于如此简单的输入设备的光标移动指令时，光标的移动不一定用相对光标移动的目的地的线性方法来实现，由用户操作引起的光标移动和用户的视线移动不匹配，并且将会出现看不到光标移动等问题。

发明内容

鉴于上面的问题做出本发明，并且使得它的目的在于提供 GUI 应用程序开发支持装置、GUI 显示装置、方法和使得它成为可能的计算机程序，当光标(聚焦)放置在每个组件上，例如，在按钮、数据输入字段、复合框、列表等时，即，放置在图形用户界面中或者相对在每个组件之间的光标(聚

焦)移动, 来执行与它的状态对应的动画表示, 以致用户能够容易地识别出光标位置, 并且, 此外, 以致甚至当光标移动时, 用户能够容易地识别出它的移动路径。

本发明的第一个方面是支持具有光标显示信息的有关图形用户界面(GUI)的 GUI 应用程序开发的 GUI 应用程序开发支持装置,

所述 GUI 应用程序开发支持装置, 其特征在于其包括:

图形表示程序文件, 其存储有关光标或者其上放置所述光标的组件的图形表示的程序;

图像存储部分, 其存储能够根据如上所述的图形表示程序文件的执行而读取的动画图像数据; 以及

编辑处理部分, 其执行以下处理: 关联在根据所述光标或者位于放置所述光标处的组件的状态执行如上所述图形表示程序时所读取的图像数据。

而且, 本发明 GUI 应用程序开发支持装置的一个实施例特征在于: 如上所述的编辑处理部分具有以下结构: 相对于处于光标放置其上的状态中的聚焦状态组件, 执行关联带有表示光标能够移动的方向的方向信息的动画的处理。

而且, 本发明 GUI 应用程序开发支持装置的一个实施例特征在于: 如上所述编辑处理部分具有以下结构: 能够与如上所述的光标放置到组件上的聚焦状态和如上所述光标在组件之间移动的光标移动状态中的每一个状态相关联地设置不同的读取图像数据。

而且, 本发明 GUI 应用程序开发支持装置的一个实施例特征在于: 如上所述的编辑处理部分具有以下结构: 能够与如上所述的光标或者位于放置光标处的组件的状态中的每一个状态相关联地设置不同的读取图像数据。

而且, 本发明 GUI 应用程序开发支持装置的一个实施例特征在于: 如上所述的编辑处理部分具有以下结构: 通过组合与组件相对应的逻辑组件和包括如上所述的图形表示程序文件的图形表示组件来生成 GUI 组件, 并如上所述的编辑处理部分具有以下结构: 如上所述的编辑处理部分生成根据光标或组件状态的图像数据和如上所述的图形表示组件相关联的 GUI 组件。

而且，本发明的第二方面是在显示器上执行光标或者在其上放置光标的 GUI 组件的动画表示的 GUI 显示装置，

所述 GUI 显示装置特征在于其包括：

聚焦导航器，其由搜索引擎根据光标移动指令输入信息执行搜索处理，并确定光标移动目的地；

聚焦管理器，其按照由如上所述的聚焦导航器所确定的光标移动目的地指令信息执行光标的移动处理；以及

聚焦对象，其按照所述光标或者与由如上所述的聚焦管理器执行的光标移动同时发生的其上放置光标的 GUI 组件的状态来执行动画显示处理，以执行所述光标或者在其上放置所述光标 GUI 组件的动画表示，其中所述聚焦对象以如下方式执行所述 GUI 组件的动画表示：朝向移动方向，依次生成具有与所述 GUI 组件的形状类似的形状的图标，并且随着所述光标移动，使后侧的所述光标消失。

而且，本发明的 GUI 显示装置的一个实施例特征在于：组成如上所述的动画的图像包括具有表示光标能够移动的方向的方向信息的图像，并且特征在于：相对于聚焦状态的组件，所述聚焦对象基于如上所述的带有方向信息的图像来执行动画显示处理。

而且，本发明的 GUI 显示装置的一个实施例特征在于：如上所述的聚焦对象执行由在组件上放置光标的聚焦状态和光标在组件之间移动的光标移动状态两者之间不同的图像所组成的动画显示处理。

而且，本发明的第三方面是 GUI 应用程序生成方法，其产生带有有关图形用户界面(GUI)的光标显示信息 GUI 应用程序，

所述 GUI 应用程序生成方法特征在于其包括：

图形表示组件选择步骤，用于选择性地从存储与光标或者光标放置其上的组件的图形表示有关的程序的图形表示程序文件存储部分中抽取图形表示组件；以及

编辑处理步骤，用于与在如上所述的图形表示组件选择步骤中所选择的图形表示组件相关联地，设置对应所述光标或者所述光标放置其上的组件的状态的图像数据。

而且，本发明的 GUI 应用程序生成方法的一个实施例特征在于，如上所述的编辑步骤包括以下步骤：对于处于其上放置光标的状态中的聚焦状

态组件，执行关联带有表示所述光标能够移动的方向的方向信息的动画的处理。

而且，本发明的 GUI 应用程序生成方法的一个实施例特征在于，如上所述的编辑步骤包括以下步骤：与如上所述的光标放置在组件之上的聚焦状态和上述光标在组件之间移动的光标移动状态中的每一个状态相关联地，设置不同的读取图像数据。

而且，本发明的 GUI 应用程序生成方法的一个实施例特征在于，如上所述的编辑步骤包括以下步骤：与如上所述的光标或者在其上放置光标的组件的状态中的每一个状态相关联地，设置不同的读取图像数据。

而且，本发明的 GUI 应用程序生成方法的一个实施例特征在于，如上所述的编辑步骤包括：通过组合与组件相对应的逻辑组件和包括如上所述的图形表示程序文件的图形表示组件来生成 GUI 组件的步骤，以及生成其中对应于所述光标或所述组件的状态的图像数据和如上所述的图形表示组件相关联的 GUI 组件的步骤。

而且，本发明的第四方面是在显示器上执行光标或者在其上放置所述光标的 GUI 组件的动画表示的动画显示处理方法，

所述动画显示处理方法特征在于其包括：

由搜索引擎根据光标移动指令输入信息执行搜索处理，并确定光标移动目的地的步骤；

按照所确定的光标移动目的地指令信息执行所述光标的移动处理的步骤；以及

动画显示步骤，用于按照与光标移动同时发生的所述光标或者其上放置光标的 GUI 组件的状态执行动画显示处理，以执行所述光标或者在其上放置所述光标的 GUI 组件的动画表示，其中在所述动画显示步骤中，以如下方式执行所述 GUI 组件的动画表示：朝向移动方向，依次生成具有与所述 GUI 组件的形状类似的形状的图标，并且随着所述光标移动，使后侧的所述光标消失。

而且，本发明的动画显示处理方法的一个实施例特征在于：组成如上所述的动画的图像包括具有表示光标能够移动的方向的方向信息的图像，并且特征在于：如上所述的动画显示步骤，对聚焦状态的组件，执行基于如上所述的带有方向信息的图像的动画显示处理。

而且，本发明的动画显示处理方法的一个实施例特征在于：如上所述的动画显示步骤执行由在所述光标放置在组件上时的聚焦状态和所述光标在组件之间移动的光标移动状态之间不同的图像组成的动画显示处理。

而且，本发明的第五方面是，执行带有有关图形用户界面(GUI)的光标显示信息的 GUI 应用程序生成处理的计算机程序，

所述计算机程序特征在于其包括：

图形表示组件选择步骤，用于选择性地从存储与光标或者光标放置其上的组件的图形表示有关的程序的图形表示程序文件存储部分中抽取图形表示组件；以及

编辑处理步骤，用于与在如上所述的图形表示组件选择步骤中所选择的图形表示组件相关联地，设置对应所述光标或者所述光标放置其上的组件的状态的图像数据。

而且，本发明的第六方面是，在显示器上执行光标或者在其上放置所述光标的 GUI 组件的动画表示的动画显示处理的计算机程序，

所述计算机程序特征在于其包括：

由搜索引擎根据光标移动指令输入信息执行搜索处理，并确定光标移动目的地的步骤；

按照所确定的光标移动目的地指令信息执行所述光标的移动处理的步骤；以及

动画显示步骤，用于按照与光标移动同时发生的所述光标或者其上放置光标的 GUI 组件的状态来执行动画显示处理。

根据本发明的 GUI 应用程序开发支持装置，因为采用了包括以下部分的结构：存储有关光标或者在其上放置光标的组件的图形表示的程序的图形表示程序文件、存储能够根据图形表示程序的执行读取的动画图像数据的图像存储部分、和执行关联在根据所述光标或者位于所述光标放置处的所述组件的状态执行所述图形表示程序时读取的图像数据的处理的编辑处理部分、以及在所述编辑处理部分，能够与所述光标或者位于所述光标放置处的所述组件的状态中的每一个状态相关联地设置不同的读取图像数据，所以自由地根据所述光标状态设置动画是可能的，并且，可能容易地实现对应于设置适合于装有 GUI 的设备的光标的最佳动画的处理。

而且，根据本发明的结构，因为可能相对放置光标的状态中的聚焦状

态组件，执行关联带有表示所述光标能够移动的方向的方向信息的动画的处理，所以提供用户容易操作的 GUI 是可能的。

而且，根据本发明的结构，因为采用了以下结构：分开构造组件的所述逻辑部分和图形表示部分、生成基于所述逻辑部分和图形表示部分的 GUI 组件、以及使得所述图形表示部分具有基于对应于所述光标或者所述组件的状态的图像的动画执行功能，所以容易地设置或者修改所述组件或者所述光标的动画模式是可能的。

而且，根据本发明的 GUI 显示装置，因为采用了以下结构：由搜索引

引擎执行搜索处理并且基于光标移动指令输入信息确定光标移动目的地、并根据所确定的光标移动目的地指令信息执行光标移动处理、而同时执行对应于所述光标或者在其上放置所述光标的 GUI 组件的所述状态的与光标移动同时发生的动画显示处理，所以实现了具有非常良好的视觉效果并使得可能容易地识别所述光标和在其上放置所述光标的组件的移动状态的 GUI。

而且，对于本发明的 GUI 显示装置，因为采用了执行由带有表示光标能够移动的方向的方向信息的图像组成的动画显示处理的结构，实现了用户容易操作的 GUI。

此外，本发明的计算机程序是能够提供给例如通用计算机系统的计算机程序，所述通用计算机系统通过以计算机可读格式提供程序代码的存储介质和通信介质，能够运行各种程序代码。所述存储介质和通信介质的例子包括诸如 CD、FD、MD 等的记录介质或者诸如网络等通信介质。通过以计算机可读格式提供这样一种程序，在计算机系统上实现了与所述程序对应的处理。

根据下面描述的本发明的实施例的更详细描述和所附的附图，本发明的其他目的、特性和优点应该变得清楚。在本描述中使用的系统指多个装置的逻辑组合装置，并且不限于其中每个组元装置属于同一主体的装置。

附图说明

图 1 是示出了本发明的 GUI 应用程序开发支持装置的硬件结构例子的图；

图 2 是示出本发明的 GUI 应用程序开发支持装置的功能结构的框图；

图 3 是解释本发明的 GUI 应用程序开发支持装置的处理序列的序列图；

图 4 是解释本发明的 GUI 应用程序开发支持装置的处理序列的序列图；

图 5 是解释为执行本发明的 GUI 应用程序开发支持装置的处理所必需的文件结构的例子的图；

图 6 是解释用本发明的 GUI 应用程序开发支持装置生成的 GUI 组件的显示例子的图；

图 7 是解释用本发明的 GUI 应用程序开发支持装置生成的 GUI 组件的显示例子的图；

图 8 是解释用本发明的 GUI 应用程序开发支持装置，执行设置对应于

每个组件的状态的动画处理的例子的图；

图 9 是示出用本发明的 GUI 应用程序开发支持装置设置组件的动画处理的功能结构的框图；

图 10 是解释执行与用本发明的 GUI 应用程序开发支持装置生成的组件对应的动画处理的处理例子的图；

图 11 是示出在本发明的 GUI 应用程序开发支持装置中，执行与光标对应的图像数据设置处理功能的结构的框图；

图 12 是示出在本发明的 GUI 应用程序开发支持装置中，执行与光标移动或者光标对应的显示图像数据设置处理的结构的框图；

图 13 是示出用装有 GUI 的设备来进行并能够指令光标移动的遥控器的结构例子的图；

图 14 是示出在 GUI 显示的显示器上与光标移动同时进行的动画显示的例子的图；

图 15 是示出在 GUI 显示的显示器上的组件之间的光标移动的例子图；

图 16 是示出在 GUI 显示的显示器上，显示对于组件的光标可移动方向标识标志的例子图；

图 17 是示出解释在光标移动处理中确定移动目的地和显示可移动方向信息的步骤的流程图的图；以及

图 18 是示出 GUI 显示装置的配置例子的框图。

具体实施方式

下面，将参照附图详细描述本发明的 GUI 应用程序开发支持装置、GUI 显示装置、方法和计算机程序。

[1.硬件结构]

首先，参照图 1 描述本发明的 GUI 应用程序开发支持装置的硬件结构例子。将参照图 1 描述在 GUI 应用程序开发支持装置 100 中的每一个元素。作为 GUI 应用程序开发支持装置 100 的主控制器的 CPU(中央处理单元)101，在操作系统(OS)的控制下，执行各种 GUI 应用开发程序。例如，CPU 101 进行基于 Java 语言的程序创建处理的实施，尤其是在 GUI 屏幕上放置的诸如按钮、复合框等的每一个组件的设置功能和设置显示图像等的

处理。如附图所示，将 CPU 101 通过总线 107 和其他的组成部分互联。

存储器 102 是用来存储由 CPU 101 运行的程序代码，或者暂时存储在执行情况下的工作数据的存储设备。在同一图中所示的存储器 102 包括诸如 ROM 的非易失存储器以及诸如 DRAM 的易失性存储器两者。

图形处理芯片 150 是其中集成了专用处理电路用来执行各种 GUI 组件的颜色处理的芯片。

显示控制器 103，作为显示控制部分，是用于实际处理由 CPU 101 发布的绘画命令的专用控制器。在例如一旦写入到帧缓冲区之后(未示出)，由显示控制器 103 处理的绘画数据由显示器 108 输出到屏幕。例如，从 HDD(111)再现的图像或由 CPU 101 处理的图像可以在显示器 108 上显示，并且用户可以查看所提供的屏幕。

输入设备接口 104 是用于将诸如键盘 109、鼠标 110 等的用户输入设备连接到 GUI 应用程序开发支持装置 100 的装置。用户能够经过键盘 109 和鼠标 110 输入用于处理或者显示图像等命令。

遵照诸如以太网的预定通信协议，网络接口 105 能够将 GUI 应用程序开发支持装置 100 连接到诸如 LAN(局域网)的本地网，并进而连接到诸如因特网的广域网。

多个主机终端和服务器(未示出)以透明的方式连接到网络，从而构建了分布式的计算环境。能够在所述网络上提供软件程序和数据内容的发行业务。例如，能够经由所述网络下载来自其中存储运动图像，静止图像等的另一个服务器的图像数据到 HDD(111)。

外部设备接口 106 是用于连接诸如数字照相机、硬盘驱动器(HDD)111、介质驱动器 112 等的外部装置到 GUI 应用程序开发支持装置 100 的装置。

HDD 111 是其中固定安装有作为存储介质的磁盘的外部存储装置，优点在存储容量、数据传送速度等方面，并允许随机存取。例如，在可执行条件下将软件程序存储到 HDD 111 的程序安装是可能的。在 HDD 111 上，以非易失方式，存储由 CPU 101 运行的操作系统、应用程序和设备驱动器的程序代码。

介质驱动器 112 是其中能够加载诸如 CD(致密盘)、MO(磁光盘)、DVD(数字多功能盘)等的便携式介质 160 并用于访问其数据记录表面的装置。

便携式介质 160 使用的主要目的是诸如按计算机可读格式的数据保存软件程序、数据文件等，并在系统(即，包括销售、流通和发行)之间转发它们。物理地流通和发行用于执行在多个利用这些便携式介质的设备之间的各种处理的应用程序是可能的。

此外，诸如如图 1 所示的装置能够按兼容机，或者 IBM 公司的个人计算机“PC/AT(个人计算机/先进技术)”的后续机器来实现。当然，应用装备有不同结构的计算机同样也是可能的。

[2.功能结构]

接下来，参照图 2 描述本发明通过独立地构造在图形用户界面(GUI)中设置的每个组件的逻辑部分和图形表示部分来开发 GUI 应用程序的 GUI 应用程序开发支持装置的功能结构。

如图 2 所示，本发明的 GUI 应用程序开发支持装置具有组件存储部分 200，其中存储了在作为用户界面的屏幕上放置的各种部分(组件)，诸如按钮、复合框、列表、窗口、文本框等。组件存储部分 200 包括逻辑组件存储部分 201、图形表示组件存储部分(简单)202 和图形表示组件存储部分(丰富)203。

逻辑组件存储部分 201 是存储有关在用做用户界面的屏幕上放置的各种部分(组件)——诸如按钮、复合框、列表、窗口、文本框等——的逻辑信息的组件。例如，假如它是按钮，则将它构建为定义作为按钮的功能的诸如用于当接通它时的处理、用于当关掉它时的处理等的功能的软件程序组件。

另一方面，将图形表示组件存储部分(简单)202 和图形表示组件存储部分(丰富)203 构建为存储组件的显示图像信息的软件程序组件，即存储图形表示程序的软件程序组件。图形表示组件存储部分(简单)202 和图形表示组件存储部分(丰富)203 是每个都存储执行不同图形表示的程序的组件。

图像数据存储部分 205 存储诸如例如 jpeg, gif 等图像数据。通过这些图像数据和每个组件的图形表示组件联系起来，根据各种组件操作的基于图像数据的动画显示成为可能。随后将描述这样的动画设置处理。

作为 GUI 应用程序的研发人员的操作者经由数据输入/输出部分 220 执行有关编辑处理部分 210 的各种数据输入，并通过设置诸如按钮、复合框等每个组件的功能和图形图像来构建 GUI 组件。

组件的创建处理执行程序是例如 JAVA 程序的一个类(工厂类)。编辑处理部分 210 通过执行所述工厂类来创建各种组件。此外,在如图 1 所示的硬件结构的情况下,编辑处理部分 210 的处理是在 CPU 101 的控制下执行的处理。

在编辑处理部分 210,编辑作为 GUI 组件的组件,为其定义了作为诸如例如按钮形状、位置信息等属性信息的“特性”,作为因为组件动作结果的处理的“方法”,以及更进一步地作为用于当诸如数据到达、中断发生、特性变化、方法调用等有关组件的动作发生时,将所述事件通知给另一个组件的功能的“事件”。

编辑处理部分 210 包括编辑组件特性组的特性编辑部分 211、编辑方法的丰富编辑部分 212 以及编辑事件的事件编辑部分 213。

在设置外表,即将要产生的 GUI 组件的图形表示中,操作员输入有关执行所述工厂类的编辑处理部分 210 的外表和感觉包规格数据。换言之,操作员规定“丰富”或者“简单”。

根据这种规格数据,在编辑处理部分 210 执行工厂类,并根据所述规格数据,从图形表示组件存储部分(简单的)202 或图形表示组件存储部分(丰富)203 读取对应的程序组件,同时用从逻辑组件存储部分 201 读取的组件来生成所述组件。

例如,在生成有关按钮的 GUI 组件中,从存储其中存储有关组件逻辑信息的组件的逻辑组件存储部分 201 中读取按钮组件,并进一步根据来自操作员的外表和感觉包规格数据,即“丰富”或“简单”的规格数据,从图形表示组件存储部分(简单)202 或图形表示组件存储部分(丰富)203 读取存储按钮的图形表示程序的组件,并通过组合所述逻辑组件和所述“丰富”或“简单”图形表示组件来生成单个的按钮组件。

对不是按钮的诸如复合框、列表、窗口、文本框等各种组件执行类似的编辑处理,并在编辑处理之后,将所设置的 GUI 应用程序存储在程序存储部分 221,在程序执行部分 222 执行,并在显示部分 223 显示所生成的 GUI。

此外,在如上所述的实施例中,描述了为存储图形表示程序的组件只设置“丰富”和“简单”两个类型的例子,但是类型数同样也可以设置为 3 及 3 以上的任意数,并且操作员可以输入某些设置图形表示组件的规格数

据，由此根据规格数据选择图形表示组件，并创建各种 GUI 组件。

[3.处理序列]

接下来，参照图 3 和图 4 描述通过应用本发明的 GUI 应用程序开发支持装置来生成 GUI 组件的处理的步骤。

图 3 是示出其中选择“简单”图形表示组件和生成 GUI 组件的处理序列的图。在图 3 中，从左开始，示出了操作员、GUI 生成应用程序、组件工厂、图形表示部分和逻辑部分的处理中的每一个处理。图形表示部分和逻辑部分的处理对应于从如图 2 所示的图形表示组件存储部分(简单) 202 和逻辑组件存储部分 201 中读取的组件中存储的程序的执行，并且在图 2 的编辑处理部分 201 执行这些处理以及组件工厂的处理。

首先，在步骤 S101，操作者输入 GUI 组件的创建命令(Create (创建))给 GUI 生成应用程序。这种创建命令(Create (创建))与规定诸如例如“按钮”、“列表”、“复合框”、“单选按钮”、“窗口”等各种组件的数据一起输入。

而且，在步骤 S102，从操作员将用于指定图形表示的“外表和感觉”规格经由应用程序输出到组件工厂。这里，假定将“简单”指定为“外表和感觉”的规格。

接下来，在步骤 S103，将组件创建命令从应用程序输出到组件工厂。表示在前面步骤 S101 中指定的组件类型的识别数据包括在这种命令中。

在步骤 S104，将所述组件创建命令(Create (创建))从有关图形表示部分(简单)和逻辑部分的组件工厂输出，并根据所述逻辑部分和图形表示部分所存储的程序执行组件创建。

在步骤 S105 之后的处理示出了在由应用程序处理组件的情形下，例如，象在当执行作为组件的“窗口”打开处理命令时的情形下的处理序列。将来自应用程序的组件打开处理命令传递给逻辑部分，并基于逻辑部分程序的执行，将绘画命令传递给图形表示部分(简单)。图形表示部分(简单)基于从逻辑部分输入的命令，执行组件的绘画(Paint)处理。

这里，因为为组件生成采用的图形表示部分是“简单”，在显示器所显示的生成组件，即按钮等，采用了由绘画进行色彩设置的简单结构。

接下来，参照图 4 描述其中选择“丰富”图形表示组件和生成 GUI 组件的处理序列。和图 3 一样，从左开始，示出了操作员、GUI 生成应用程序、组件工厂、图形表示部分和逻辑部分的处理中的每一个处理。和图 3

一样，图形表示部分和逻辑部分的处理对应在从如图 2 所示的图形表示组件存储部分(简单)202 和逻辑组件存储部分 201 中读取的组件中存储的程序的执行，并且在图 2 的编辑处理部分 201 执行这些处理以及组件工厂的处理。

首先，在步骤 S201，操作者输入 GUI 组件的创建命令(Create (创建))给 GUI 生成应用程序。这种 GUI 组件的创建命令(Create (创建))与规定诸如例如“按钮”、“列表”、“复合框”、“单选按钮”、“窗口”等组件类型的数据一起输入。

而且，在步骤 S202，从操作员将用于指定图形表示的“外表和感觉”规格经由所述应用程序输出到组件工厂。这里，假定将“丰富”指定为“外表和感觉”规格。

接下来，在步骤 S203，将组件创建命令从应用程序输出到组件工厂。表示在前面步骤 S201 中指定的组件类型的识别数据包括在这种命令中。

在步骤 S204，相对于图形表示部分(丰富)和逻辑部分，将组件创建命令(Create (创建))从组件工厂输出，并根据逻辑部分和图形表示部分(丰富)所存储的程序执行组件创建。

在步骤 S205 之后的处理示出了在由应用程序处理组件的情形下，例如，象在当执行作为组件的“窗口”打开处理命令时的情形下的处理序列。将来自应用程序的组件打开处理命令传递给图形表示部分部分，并基于在图形表示部分中存储的动画程序的执行，执行动画(S206)。

而且，在步骤 S207，将打开处理命令传递给逻辑部分，并根据逻辑部分的程序的执行，将绘画命令传递给图形表示部分(简单)，并且图形表示部分(丰富)根据从逻辑部分输入的命令执行组件的绘画处理。

在图 4 所示的序列图中，因为为组件生成采用的图形表示部分是“丰富”，在显示器上显示的生成组件，即按钮等，采用了不仅由绘画进行色彩设置，而且也执行动画表示的结构。

[4.文件结构例子]

将描述用于执行如上所述的组件显示的文件结构例子。

例如，作为在象有作为 GUI 组件例子的按钮功能的图标的“按钮控件 (button)”组件配置为 GUI 组件的情形下的文件，例如，下面文件的每一个，将是必要的。

Cbutton.java:实现按钮控件的逻辑部分的类,

Button.java:用于实现作为按钮控件的图形表示部分的外表和感觉部分的类,

button_normal_off.gif:当按钮控件“关”时的图像,

button_normal_on.gif:当按钮控件“开”时的图像,

button_pushed.gif:当推按钮控件时的图像,

CComponentFactory.java:工厂类, 应用程序使用来创建按钮控件;

Application.java:用于显示所创建的按钮控件的应用程序。

类指 Java 中可执行程序单元。

在图 5 中示出了上面文件的每一个的目录结构。作为工具箱的 [customtool kit(定制工具箱)]文件存在于某些 GUI 应用程序文件 [application(应用程序)]之下, 并且在这个文件中包括了 CButton.java: 实现按钮控件的逻辑部分的类, CComponentFactory.java: 应用程序使用来创建按钮控件的工厂类, 以及进一步作为图形表示信息的外表和感觉文件[laf]。

在外表和感觉文件[laf]包括了 Button.java:用于实现作为按钮控件的图形表示部分的外表和感觉部分的类, 即, 对应于如上所述的图形表示组件的类。而且, 作为图像数据文件[image(图像)], 包括了 button_normal_off.gif:当按钮控件“关”时的图像、button_normal_on.gif:当按钮控件“开”时的图像、button_pushed.gif:当推按钮控件时的图像中的每一个。

根据用户输入信息, GUI 应用程序执行用于应用包括在这些文件的每一个中的处理程序和图像数据的处理。具体地讲, 执行依照逻辑部分类的程序的各种功能执行处理, 以及依照图形表示部分类的程序的各种图形表示处理。

[5.显示例子]

接下来, 参照图 6 和 7, 描述基于其中每个组件的逻辑部分和图形表示部分分开的结构所生成的 GUI 组件显示处理的例子。

图 6 示出了当应用图形表示部分(简单)时生成和显示每个组件的例子。将按钮 311 和复合框 312 作为 GUI 组件的例子来给出描述。

当操作按钮 311 时, 例如当将按钮开、关时, 或者当进行聚焦设置, 换言之光标设置时, 由按钮组件的逻辑部分的处理来执行基于按钮的开和关操作的处理。而且, 通过例如来自逻辑部分的绘画命令, 执行图形表示

部分(简单), 并且, 进行绘画处理, 即改变按钮显示颜色的处理。

当操作复合框 312 时, 例如, 当对复合框 312 进行开、关等时, 由复合框的逻辑部分的执行基于复合框开和关操作的处理。而且, 通过来自例如逻辑部分的绘画命令, 执行复合框组件的图形表示部分(简单), 并执行绘画处理, 即诸如改变复合框显示颜色的处理。

此外, 可以为每个 GUI 组件不同地设置通过操作员的处理执行的图形表示类型。换言之, 通过改变图形表示部分(简单)的处理程序, 适合每个组件的图形表示处理成为可能。

图 7 示出了当应用图形表示部分(丰富)时生成和显示每个组件的例子。将按钮 320 和复合框 330 作为 GUI 组件的例子给出描述。

当操作按钮 320 时, 例如当将按钮开、关时, 或者当进行聚焦设置, 换言之光标设置时, 除了如上所述的绘画处理外, 还执行作为按钮组件的图形表示部分(丰富)处理的动画 321 和 322。尽管从图中难以看到, 但是, 在(b)和(c)中, 执行了围绕按钮的波纹传播的动画。此外, 通过逻辑部分的处理, 执行了基于按钮开和关操作的处理。

当操作复合框 330 时, 例如当将按钮开、关等时, 通过复合框组件的逻辑部分的处理, 执行基于复合框开和关操作的处理, 而同时执行作为复合框组件的图形表示部分(丰富)处理的动画 331 和 332。尽管从图中难以看到, 但是, 在(b)和(c)中, 执行了设置来使得其从复合框的较低部分向下延伸的动画。

[6.动画设置和执行处理]

接下来, 描述设置和执行诸如窗口、按钮、复合框、列表等 GUI 组件的动画的处理。

图 8 是示出用于设置执行有关组件动画的图像数据的处理屏幕的图。采用了以下结构: 在组件限定字段 351 中设置组件[按钮]、在它下面配置图像数据设置字段 352, 并能够单独地设置在每个组件状态(即在正常状态下、或者在开、关和聚焦设置等)下所调用的图像(jpeg)。聚焦设置指将光标放置在那个组件上的状态, 并且该组件处于其能够进行诸如开和关操作的状态。

例如, 当实现处理时, 根据为图形表示组件所定义的程序调用并执行动画索引方法。通过动画索引方法的执行, 从图像文件[Image]读取 Jpeg 图

像 11 至 21，并在动画中显示基于这些多个图像的在 GUI 上所显示的按钮图像。当执行关处理时，从图像文件[Image]读取 Jpeg 图像 31 至 38，并在动画中显示基于这些多个图像的在 GUI 上所显示的按钮图像。

在如图 8 所示的例子中，示出了作为组件状态的正常、开、关和聚焦 4 个状态的例子。然而，状态定义不限于这些，并且例如，可以按下面的方式定义，而且可以采用根据每个状态设置图像数据的配置。

NORMAL: 正常状态

NORMAL DEACTIVATED: 没有激活的正常状态

FOCUSED: 聚焦状态

FOCUSED DEACTIVATED: 没有激活的聚焦状态

FUSHED: 被按的状态

PUSHED DEACTIVATED: 没有激活的被按状态

DISABLED: 不可操作状态

DISABLE DEACTIVATED: 没有激活的不可操作状态

通过设置与上面 8 个状态的每一个相关联的组成动画的不同图像数据，用户根据动画理解组件的状态是可能的。此外，没有激活的状态指诸如其中在没有操作的窗口中显示它的状态的情形。

因此，对于每个组件，能够设置单独的图像文件以对应处理，并且自由地设置对应各种处理的最优动画表示成为可能。在附图中，示出了设置按钮动画的例子，但是，类似地，设置其他组件的对应各种处理的动画同样也是可能的。

例如，通过从动画在开时、动画在关时等的图像文件中为复合框分别单独选择和设置图像，来设置各种动画。

通过前面参照图 1 和图 2 描述的结构，实现用于通过设置对应于每个组件的状态的图像数据来执行如上所述的动画显示设置处理的 GUI 应用程序开发支持装置。不过，将参照如图 9 所示的框图来描述，限于执行根据每个组件状态来设置图像文件的处理的功能的处理结构。

如图 9 所示，GUI 应用程序开发支持装置具有存储在作为用户界面的屏幕上设置的诸如按钮、复合框、列表、窗口、文本框等各种组件的组件存储部分 400。组件存储部分 400 有作为定义组件的处理功能的软件程序组件的逻辑组件存储部分 401，和作为定义组件图形表示的软件程序组件的图

形表示组件存储部分 402。此外，如上所述，可以将图形表示组件构造为多个不同的组件，该不同的组件执行诸如简单、丰富等多种类型的图形表示。

图像数据存储部分 403 存储根据动画索引方法读取的图像数据，所述动画索引方法通过执行作为存储在图形表示组件存储部分 402 中所存的图形表示组件中的程序的图形表示类来调用。图像数据是例如 jpeg,gif 等数据文件，并对应于在图形表示组件存储部分 402 中存储的图形表示组件的每一个。

作为 GUI 应用程序的研发人员的操作员执行经由数据输入/输出部分 420 相对编辑处理部分 410 的各种数据输入，并通过设置诸如按钮、复合框等每个组件的功能和图形图像来构造 GUI 组件。

组件的创建处理执行程序是例如 JAVA 程序的一个类(工厂类)。编辑处理部分 410 通过执行工厂类来创建各种组件。此外，在如图 1 所示的硬件结构的情形中，编辑处理部分 401 的处理是在 CPU 101 的控制下执行的。

在设置动画的处理中，正如参照图 8 所描述的那样，在编辑处理部分 410 中执行关联图像数据以根据组件的状态读取的处理。将此处理作为组件的“特性”、“方法”和“事件”的编辑处理的一部分来执行。

在设置将要生成的 GUI 组件的外表(即图形表示)的过程中，操作员相对于执行工厂类的编辑处理部分 410 输入外表和感觉包规格数据。换言之，操作员指定“丰富”或“简单”，并进而，如参照图 8 所述，设置根据组件的状态将要读取的图像数据。

按照这样的设置信息，生成具有各种组件的 GUI 应用程序，将所生成的 GUI 应用程序存储在程序存储部分 421，在程序执行部分 422 中执行，并将所生成的 GUI 在显示部分 423 上显示。

参照图 10，将描述在动画执行时的处理，即在 GUI 显示装置的处理。将描述作为在显示器 513 上所示 GUI 组件的按钮的动画 514 的显示处理。

例如，如果组件是按钮组件，则由在按钮组件中包括的图形表示组件的图形表示类 511 来执行动画显示处理。按照为图形表示类 511 所定义的程序，调用动画索引方法，从图像文件[Image(图像)]512 读取前面参照图 8 描述的设置处理中所设置的 Jpeg 图像 11 至 21，并将基于这些多个图像进行 GUI 显示的按钮图像，作为动画 514 来显示。

这里，示出了其中显示在按钮周围波纹传播动画的例子。能够通过获得存储器 102、HDD 111 或者图 1 的硬件结构的某些其他存储装置的部分区域，并将它设置为图像文件存储部分，来存储图像文件[Image(图像)]512。

在总结如上所述的作为在显示器上所显示的 GUI 组件的动画显示处理方法系列的处理中，能够将它归纳为执行步骤，包括：

- (1)类执行步骤，用于执行与组件相联系的图形表示组件的图形表示类，
- (2)图像读取步骤，用于按照在类执行步骤中所调用的动画索引方法，从图像文件中读取预置的图像数据；以及
- (3)在显示器上显示在图像读取步骤中所读取的图像数据。

将这些处理步骤作为用于执行 GUI 应用程序的程序来记录，并根据用户的 GUI 操作来执行，换言之，执行动画显示。

此外，对于像按钮的组件的动画，通过修改所调用的图像数据能够容易地设置各种动画。例如，可以设置各种动画，诸如生成在按钮中凹或凸形的变形，或者当按钮开和关操作时改变颜色。此外，同样也可采用以下结构：当在其上设置的光标为聚焦状态时，而不是仅仅在开或关操作时连续地执行动画。这通过在重叠按钮组件位置和光标位置的条件下，调用动画索引方法、以及从图像数据读取和执行聚焦图像文件来实现。

此外，正如参照图 2 所述，在甚至用于例如按钮的相同组件的图形表示组件存储部分中也预备了存储不同图形表示处理执行程序的多个图形表示组件，如同在简单、丰富等情形中那样。在图 2 的例子中，仅仅示出了简单和丰富两种类型，但是，进一步地，设置诸如丰富 1、丰富 2、...等多个图形表示组件是可能的。

对于这些图形表示组件的每一个，将每个都不同的单独读取图像文件联系起来是可能的。甚至对于有相同功能的按钮，也由初始的外表和感觉包规格确定将要应用的图形表示组件，并显示对应那个组件所设置的动画。因此，根据外表和感觉包规格进行各种不同的动画显示成为可能。

[7.光标的动画表示]

如上所述，诸如按钮、复合框等的组件每个都有作为单独软件程序的逻辑部分和图形表示部分，而且，所述图形表示程序具有其中可能从诸如简单、丰富等多个文件中选择的结构。

接下来，描述用于按照来自诸如鼠标、键盘、由上/下/左/右指令键组成

的箭头键等输入装置的输入而可移动的光标的图形表示处理。在本描述中，光标用作包含用于表示在窗口上的指令位置的标识显示的术语，并将作为包括例如指针、光标等的概念来描述。此外，当光标处于诸如按钮、列表、复合框等各种组件位置时，将这样的组件说成是处于聚焦状态。

至于组件的图形表示，正如前面所述，读取根据所述组件的各种状态（诸如，更具体地讲，例如如果所述组件是按钮，所述按钮组件处于被推按状态、聚焦状态等）来单独设置的图像文件，并根据所读取的图像文件执行动画显示。

不仅当将光标放到特定组件上，换言之，当聚焦组件时，而且当在组件之间移动时，例如，当将光标从一个按钮组件移到另一个按钮组件时，光标执行动画表示。

按照光标或者处于设置光标处的组件的状态来设置这些各种动画的处理，能够通过基本上类似于前面描述的用于设置每一个组件状态的图像文件的处理来执行，并通过前面参照图 1 和图 2 描述的结构实现执行这种设置处理的 GUI 应用程序开发支持装置。然而，将参照图 11 所示的框图来描述限于用于按照光标执行设置图像文件处理的功能的处理结构。

如图 11 所示，GUI 应用程序开发支持装置具有存储在作为用户界面的屏幕上设置的诸如按钮、复合框、列表、窗口、文本框等各种组件的组件存储部分 560。组件存储部分 560 具有逻辑组件存储部分 561 和图形表示组件存储部分 562。图形表示组件存储部分 562 包含对应于光标的、存储有关用于光标或者其上放置光标组件的图形表示的程序的图形表示程序文件。

图像数据存储部分 563 是存储按照动画表示程序的执行而读取的动画图像数据的存储部分。图像数据存储部分 563 包含光标图像、用于将光标放置在组件上时的图像，换言之，用于在聚焦设置期间的图像，以及用来显示在组件上放置的光标的能够移动的方向的箭头的箭头显示图像，这将在随后进行描述。图像数据是例如诸如 jpeg、gif 等图像数据文件。

作为 GUI 应用程序研发人员的操作员经由数据输入/输出部分 580 对编辑处理部分 570 进行各种数据、命令和参数的输入，并执行将根据光标或者位于放置光标处的组件的状态将要读取的图像数据相关联的处理。

在编辑处理部分 570，执行设置作为光标控制程序的聚焦导航器和聚焦

管理器的处理, 所述聚焦导航器用于由搜索引擎根据光标移动指令输入信息执行搜索处理, 并确定光标移动目的地, 所述聚焦管理器用于按照由聚焦导航器所确定的光标移动目的地指令信息执行光标的移动处理, 而且, 相对聚焦导航器执行设置光标移动目的地选择算法的处理, 所述光标移动目的地选择算法用于根据光标移动指令输入信息选择单个的光标移动目的地。随后将描述聚焦导航器和聚焦管理器的细节。

此外, 在编辑处理部分 570, 相对于处于放置光标状态的聚焦状态组件, 执行具有表示光标能够移动的方向的方向信息的图像相联系的处理。在编辑处理部分 570 中的处理是在 CPU 101 的控制下, 在图 1 的硬件结构图中所执行的处理。

根据这样的设置信息, 生成有各种组件的 GUI 应用程序, 在程序存储部分 581 中存储并在程序执行部分 582 中执行所生成的 GUI 应用程序, 而且在显示部分 583 上显示所生成的 GUI。

接下来, 将参照图 12 的框图描述用于根据所生成的 GUI 应用程序执行光标的动画表示处理的结构, 换言之, 用于通过诸如信息器件等装有 GUI 的设备, 或者 GUI 显示装置进行处理的功能结构。

聚焦对象 601 是具有其中描述光标的动画表示或者光标位置的程序的程序文件。聚焦管理器 602 是管理程序, 用来按照从聚焦导航器 603 输入的光标移动目的地指令信息来执行光标移动处理, 并且还执行诸如将动画执行命令通知给聚焦对象 601 等处理。聚焦导航器 603 具有搜索引擎 604, 并且基于来自诸如鼠标、箭头键等的输入部分的用户输入, 由搜索引擎 604 执行搜索处理, 确定光标的移动目的地, 并将所确定的光标的移动目的地等信息输出到聚焦管理器 602。

聚焦导航器 603 根据依照光标移动指令输入信息而选择单个的光标移动目的地的光标移动目的地选择算法来执行光标移动目的地确定处理。光标移动指令输入信息是例如在上/下/左/右 4 个方向之一上的移动方向指令信息, 并且由聚焦导航器 603 所执行的光标移动目的地选择算法是例如, 从存在于对应于移动方向指令信息的方向中的多个组件中确定一个组件作为光标移动目的地的算法。

作为指定光标移动的用户输入设备, 例如, 具有诸如红外等的输出部分的遥控器是可使用的。在图 13 中示出了遥控器的结构例子。如图 13 所

示的控制器包括用于具有 GUI 的受控设备的电源开关 701、由红外输出装置等组成的输出部分 702、显示器 703、用于实现各种规定的输入按钮 704 和带有指示光标上/下/左/右方向移动的功能的箭头键 705。如图 13 所示的控制器只是一个例子，并且适合装有 GUI 的设备的功能的各种结构是可能的。此外，输入部分不限于遥控器，并且可以是集成到装有 GUI 的设备自身的输入部分。

将描述与动画表示同时进行的光标移动处理序列。当生成来自诸如箭头键等输入部分的用户输入时，聚焦导航器 603 接收输入信息。聚焦导航器 603 基于输入信息执行应用搜索引擎 604 的移动目的地搜索处理。

基于搜索引擎 604 的搜索结果，聚焦导航器 603 确定移动目的地组件。对于移动目的地确定处理模式，可以应用各种模式，其具体例子随后将进行描述。

一旦确定了移动目的地，聚焦导航器 603 就将移动目的地指令输出到聚焦管理器 602，并且聚焦管理器 602 执行朝向移动目的地的光标移动处理。而且，聚焦管理器 602 输出动画执行指令到聚焦对象 601，进而聚焦对象 601 运行动画处理程序，换言之，设置为对应于光标的光标文件的读取处理和显示处理。

在图 14 中示出了在光标移动期间的动画表示例子。正如参照图 12 所述，根据图像数据的读取处理，由聚焦对象执行动画执行处理。在图 14 中的处理例子示出，从在显示器上显示按钮 721 开始，光标移动到按钮 722、按钮 721 的聚焦状态终止，以及到按钮 722 进入聚焦状态的显示器上的显示例子序列((a)到(d))。

在图 14 的例子中，执行动画表示，其中连续地在光标移动的方向上生成非常像按钮的长方形框，以及随着光标移动，尾随的长方形框消失。这些动画表示是靠由聚焦对象读取的图像数据而执行的动画表示。这些图像数据的设置是基于由类似于前面参照图 8 所描述的、按照组件状态设置图像数据处理的处理来设置的图像数据的动画显示。

聚焦导航器 603 接收诸如如图 13 所示的遥控或者作为设备自身的输入部分的箭头键的输入，并由基于输入信息的应用搜索引擎 604 的搜索处理来确定光标移动目的地。参照图 15 将描述用于确定光标移动目的地处理的例子。图 15 示出两个不同的处理例子(a)和(b)。两者都显示了其中生成指定

右方向移动的用户输入，而光标放置于按钮 751 并且按钮 751 处于聚焦状态的处理例子。

图 15(a)是在按钮 A 至 C 位于按钮 751 的右边情况下，将最上面的按钮 A 确定为移动目的地的例子，而图 15(b)是在按钮 A 至 C 位于按钮 751 的右边的情况下，将中间按钮 B 确定为移动目的地的例子。

图 15(a)是从位于按钮 751 的右边的按钮中选择最上面的按钮的处理，并且基于每个按钮组件的坐标信息(X,Y)由聚焦导航器 603 执行这种处理。例如，通过比较按钮 751 的 X 坐标和每个按钮组件的 X 坐标的处理，来执行选择位于按钮 751 右边的按钮的处理，并且进而，将所选择按钮组件中有最大 Y 坐标的一个确定为光标移动目的地。

图 15(b)是其中从位于按钮 751 的右边的按钮中选择距离按钮 751 最近的按钮的处理，并且这种处理是基于由聚焦导航器 603 进行的、根据每个按钮组件的坐标信息(X,Y)计算和按钮 751 距离的处理和比较计算结果的处理的移动目的地选择处理。

如何根据用户输入信息确定移动目的地取决于由聚焦导航器 603 和搜索引擎 604 执行的程序，并且通过程序设置，能够用各种方法设定基于用户输入信息的移动目的地确定算法。例如，如上所述，可以构造如下算法：从存在于在对应于移动方向指令信息的方向中的多个组件中，选择和目前光标放置其上的组件距离最近的组件，或者可以构造如下算法：从存在于在对应于移动方向指令信息的方向中的多个组件中，选择最远端的组件的算法。在 GUI 应用程序开发支持装置中的编辑处理部分 570(见图 11)执行这些算法设置处理。

此外，在诸如按钮等的组件的逻辑组件中，事先设定对应输入信息的移动目的地同样也是可能的，并且在此情形中，不执行聚焦导航器 603 的移动目的地搜索处理，并且根据组件带有的移动目的地设置信息确定移动目的地。

因此，聚焦导航器执行基于输入信息应用搜索引擎的移动目的地确定处理，但是，进一步地，使之可能获取有关每个组件所能够移动的方向的信息，以及相对于已经进入聚焦状态的组件，显示诸如箭头等表示光标能够移动的方向的图形图像。能够在对应于前面所述的组件或者作为对应于光标的聚焦对象的图形表示组件中设置这种图像显示处理程序。

在图 16 中示出了显示箭头的例子。在(a)的显示例子中,光标能够从放置光标、并处于聚焦状态的按钮组件 781 移至其上的组件存在于上/下/左/右四个方向。对于处于这样一种状态的组件,显示表示在上/下/左/右四个方向存在可移动组件这一事实的箭头。在(b)中,光标能够从处于聚焦状态的按钮组件 782 移至其上的组件存在于上和右两个方向。对于处于这种状态的组件,显示表示存在在上和右两个方向可移动组件这一事实的箭头。

在(c)中,相对处于聚焦状态的按钮组件 783,光标能够移至其上的组件仅仅存在于朝下方向。对于处于这样一种状态的组件,仅仅在朝下的方向上显示箭头。在(d),相对处于聚焦状态的按钮组件 784,光标能够移至其上的组件仅仅存在于朝上方向。对于处于这样一种状态的组件,仅仅在朝上的方向上显示箭头。

将参照如图 17 所示的流程描述在光标移动处理中的移动目的地确定和可移动方向信息显示步骤。首先,在步骤 S301,确定是否已经存在来自诸如箭头键等的输入部分的任何输入,并且如果确定已经存在输入,则处理进行到步骤 S302,并且确定在光标目前位置的组件是否具有表示移动目的地指令的移动信息,例如,移动目的地组件信息是否包含在组件的逻辑部分文件中。如果所述信息存在,则处理进行到步骤 S310,根据移动信息确定移动目的地,并且在步骤 S307,执行相对于所确定的移动目的地的光标移动。

如果在步骤 S302,确定处于聚焦状态中的组件不具备表示移动目的地指令的信息,则处理进行到步骤 S303,执行基于用户输入信息的搜索引擎的移动目的地搜索,并在步骤 S304,聚焦导航器基于搜索引擎的搜索结果确定移动目的地。而且,在步骤 S305,聚焦导航器根据在围绕移动目的地组件的 4 个方向中的组件位置信息获取移动目的地组件的光标可移动方向信息。

在步骤 S306,将由聚焦导航器所确定的移动目的地信息和移动目的地组件的可移动方向信息传递给聚焦管理器,并在步骤 S307,执行相对移动目的地的光标移动处理。

在步骤 S308,确定聚焦管理器是否已经从聚焦导航器接收到移动目的地组件的光标可移动方向信息,并且如果没有收到,终止所述处理,然而,如果接收到,则将根据可移动方向信息执行用于标识信息(例如,箭头)的显

示处理的命令输出到聚焦对象，而聚焦对象则根据所输入的命令执行箭头显示动画。

换言之，聚焦对象根据从聚焦管理器所输入的可移动方向信息选择图像数据，并执行例如由其中在可移动方向中设置箭头的图像组成的动画显示，这一点前面参照图 16 描述过了。

[8.GUI 显示装置结构例子]

接下来，将参照图 18 描述诸如带有执行符合上面描述的各种显示处理图形用户界面的装置的结构例子。

作为装有 GUI 的设备，可以想到各种家用设备和通信设备。如图 18 所示的装置结构例子示出了一种这样的例子，并且示出了带有如下结构的 GUI 显示装置结构例子：通过经由网络接收、或者通过从存储介质读取运动图像，能够显示该运动图像。

如图 18 所示的 GUI 显示装置 800 包括用于接收因特网发布的数据、广播数据等的通信接口 801，并从数据发布站点接收 HTML 文档、电影内容等。另外，存储介质接口 802 是用于诸如 DVD、快闪存储器等各种存储介质 820 的数据转发接口，并且是用于在存储介质 820 和数据处理装置 800 之间输入和输出诸如运动图像数据等各种内容的接口。

当对从图像发布站点发送的内容或者存储在记录介质 820 上的内容编码时，在解码/编码处理部分 803 执行解码处理之后，在显示控制部分 808 的控制下，在显示装置 810 上显示它们。

另一方面，从控制器 811 输入用户的各种命令或者前面所述的光标移动指令信息，由接收部分 807 接收，并输出到控制部分(CPU) 804。控制部分 804 根据前面所述的对应每个组件的逻辑组件、图形表示组件等，执行各种处理。另外，执行聚焦管理处理，换言之，执行与光标移动同时发生的光标移动目的地组件确定处理。而且，按照聚焦对象执行由表示可移动的方向的（例如箭头）图像数据组成的动画显示处理。

在 GUI 显示装置 800 的存储部分是用来存储所接收数据、用来存储各种处理程序、用来存储预置显示图像程序的存储部分，作为 CPU 804 等的工作区的存储区域、并由 RAM、ROM 等组成。定时器 806 用来测量在 GUI 处理中的各种程序的运行时间。

上面已经参照具体实施例描述了本发明。然而，对于本领域的普通技

术人员来说，显而易见，在不脱离本发明的范围内对这些实施例进行修改和替代是可能的。换言之，已经以例子的形式披露了本发明，但不应认为这些例子是限制性的。在确定本发明的范围的过程中，应该考虑权利要求。

另外，能够通过硬件、软件或者两者组合的结构来实现在本说明书中描述的处理序列。在通过软件执行这些处理的过程中，可以将其中记录了处理序列的程序安装到集成为专用硬件的计算机的存储器中并执行，或者可以将所述程序安装到能够执行各种处理的通用计算机中并执行。

例如，将程序预先记录在作为记录介质的硬盘或 ROM(只读存储器)上。或者，可以在诸如软盘、CD-ROM(致密盘只读存储器)、MO(磁光盘)、DVD(数字多功能盘)、磁盘、半导体存储器等的可移动记录介质上暂时或者永久地存储所述程序。可以将这样的可移动记录介质作为所谓的封装软件提供。

另外，除了从如上所述的可移动记录介质安装到计算机之外，所述程序也可以从下载网站无线转发到计算机，或者可以通过电线经由诸如 LAN(局域网)和因特网等网络转发到计算机，并且所述计算机可以接收这样转发的软件，并将它安装到诸如内置硬盘等的记录介质上。

另外，在本说明书中所描述的各种处理不仅可以按如上所述的时间顺序来执行，而且取决于执行所述处理的装置的处理能力、或者按照要求，可以以并行方式或者单独地执行。另外，在本说明书中所使用的系统指多个装置的逻辑组合结构，并不限于其中每个组成装置都处于同一主体的结构。

工业适用性

如上所述，因为本发明的 GUI 应用程序开发支持装置具有以下结构：该装置包括：用于存储有关光标或者光标放置其上的组件的图形表示的程序的图形表示程序文件、用于存储能够根据图形表示程序的执行而读取的动画图像数据的图像存储部分、和用于执行将图像数据相关联以在根据光标或者位于光标放置处的组件的状态执行图形表示程序时读取的处理的编辑处理部分，并且在该结构中，它使得在编辑处理部分，能够根据光标或者位于光标放置处的组件的每一个状态设置不同的读取图像数据，并且使得自由地设置适合于光标状态的动画成为可能，并且，可能容易地实现对应于适合于装有 GUI 的设备的光标的最佳动画设置处理。

而且，根据本发明的结构，因为对于放置光标的状态中的聚焦状态组件，可以执行以下处理：将带有表示光标能够移动的方向的方向信息的动画相关联，所以提供用户容易操作的 GUI 是可能的。

而且，因为采用了以下结构：分开构造组件的逻辑部分和图形表示部分、生成基于逻辑部分和图形表示部分的 GUI 组件、以及付与图形表示部分基于对应于光标或者组件的状态的图像的动画执行功能，所以能够通过独立于逻辑部分的处理容易地设置或者修改所述组件或者光标的动画模式。

而且，对于本发明的 GUI 显示装置，因为采用了以下结构：由搜索引擎执行基于光标移动指令输入信息的搜索处理、确定光标移动目的地、并根据所确定的光标移动目的地指令信息执行光标移动处理、而同时执行对应于光标或者在其上放置光标的 GUI 组件的状态的与光标移动同时发生的动画显示处理，所以实现了具有非常良好的视觉效果 GUI，能够容易地识别光标和其上放置光标的组件的移动状态。

而且，对于本发明的 GUI 显示装置，因为采用了执行由带有表示光标能够移动的方向的方向信息的图像所组成的动画显示处理的结构，所以实现了用户容易操作的 GUI。

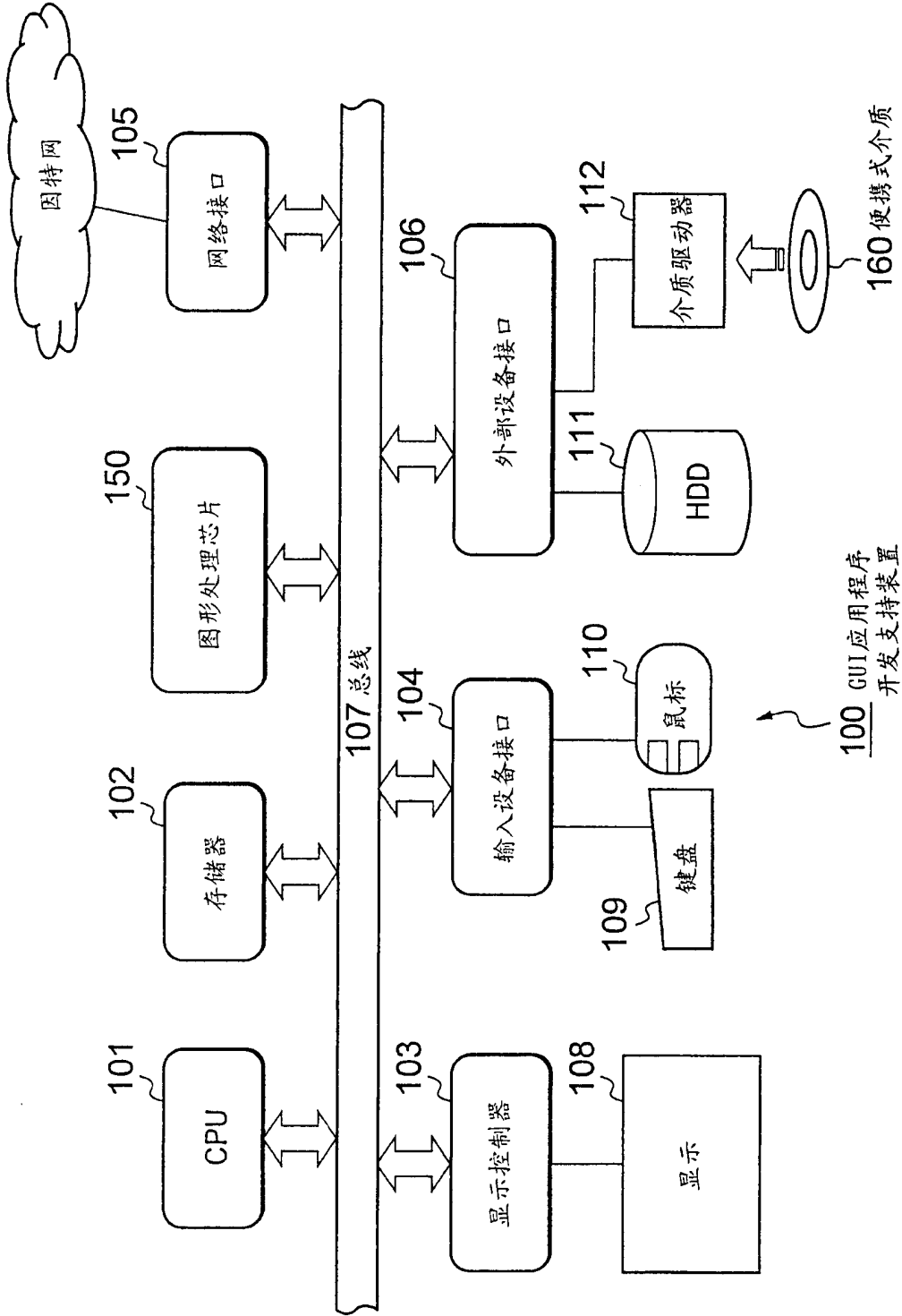


图 1

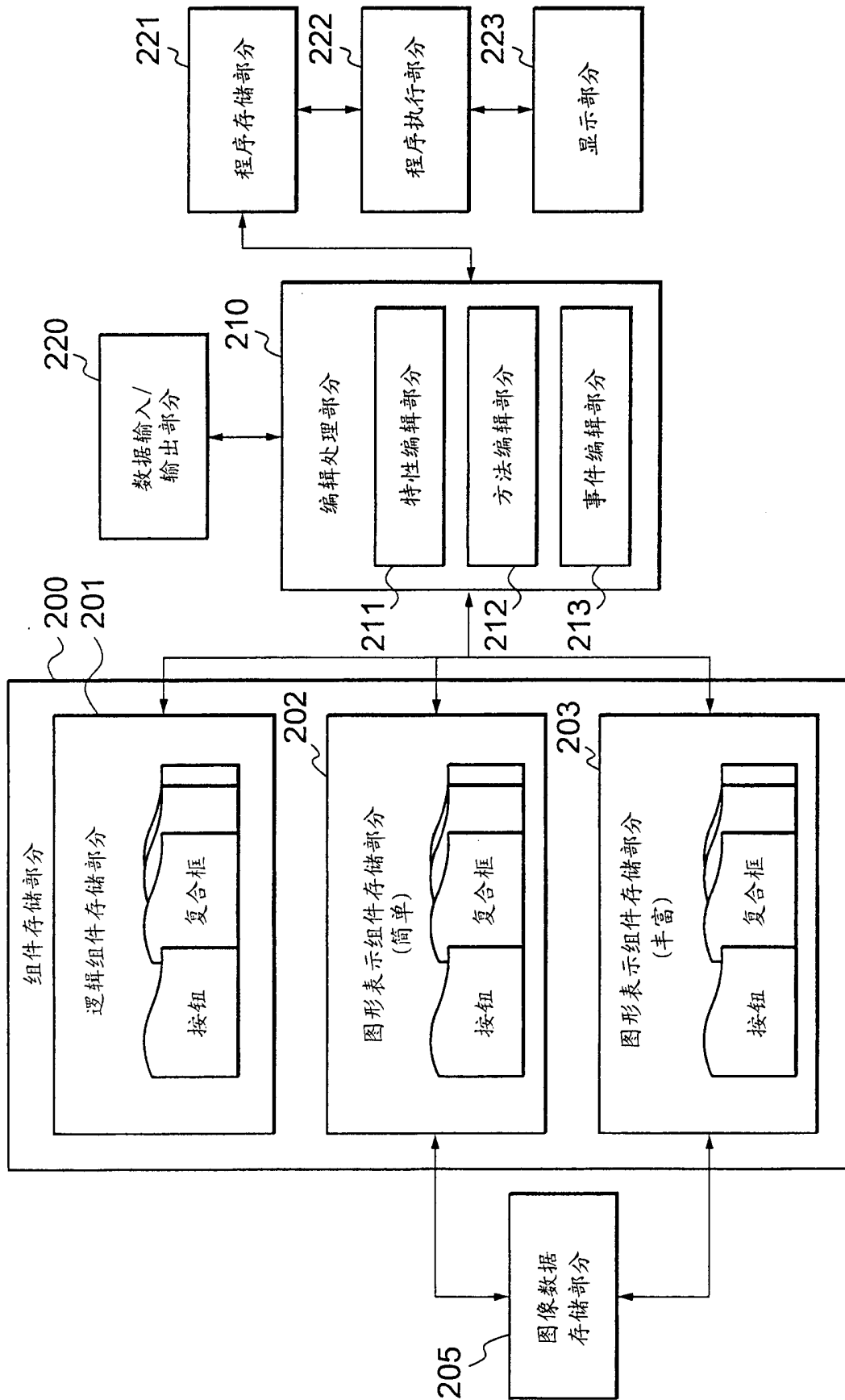


图 2

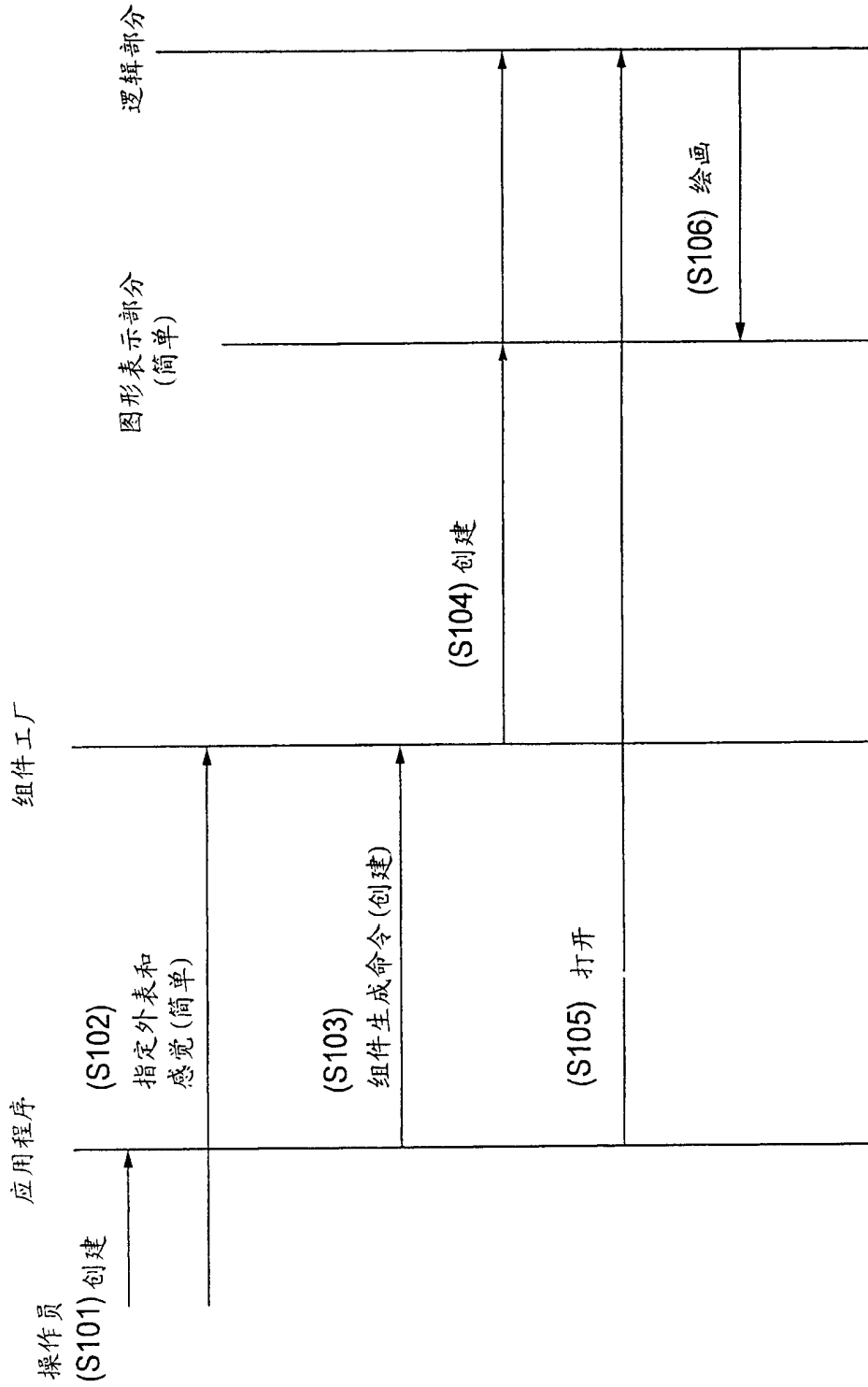


图 3

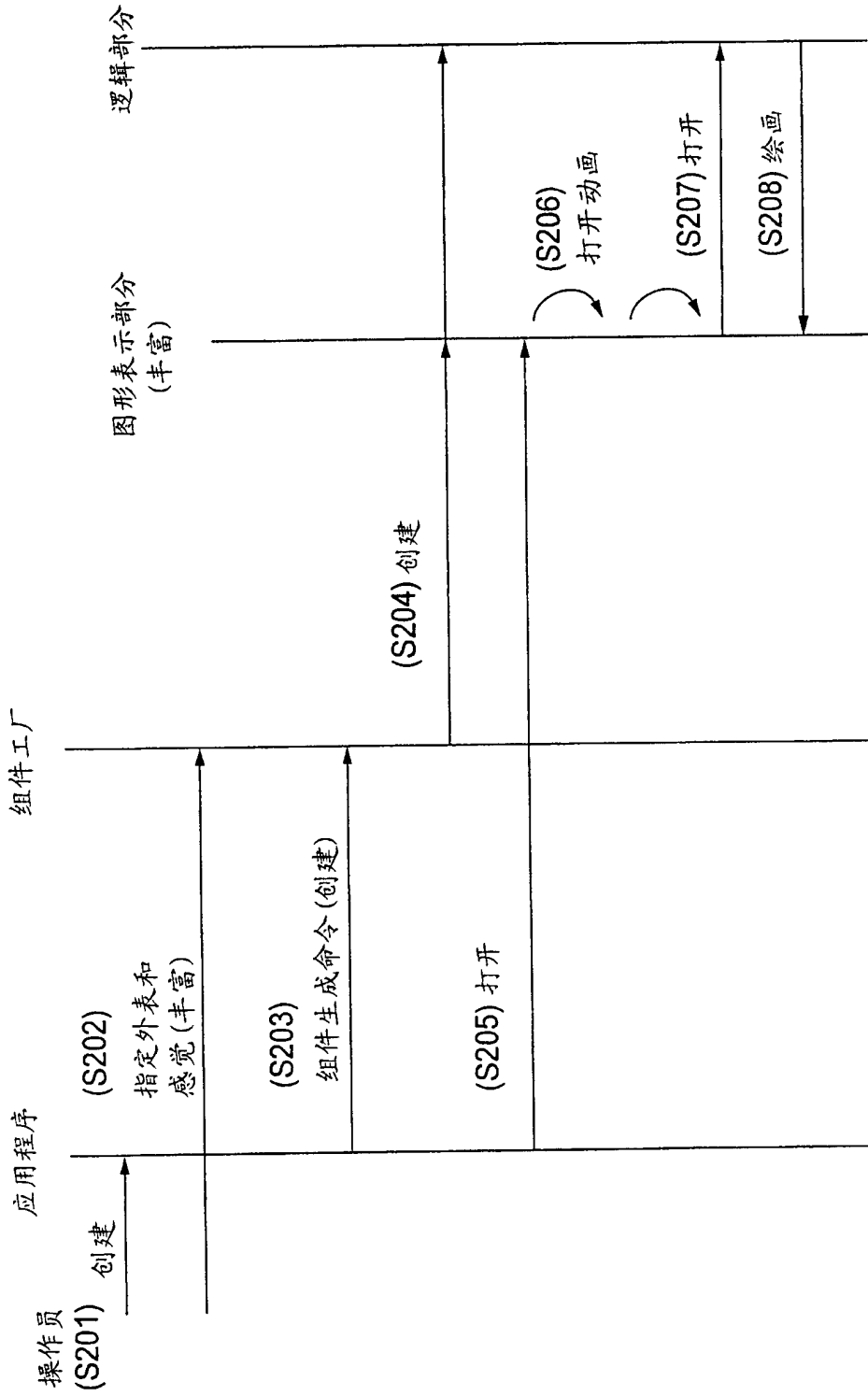


图 4

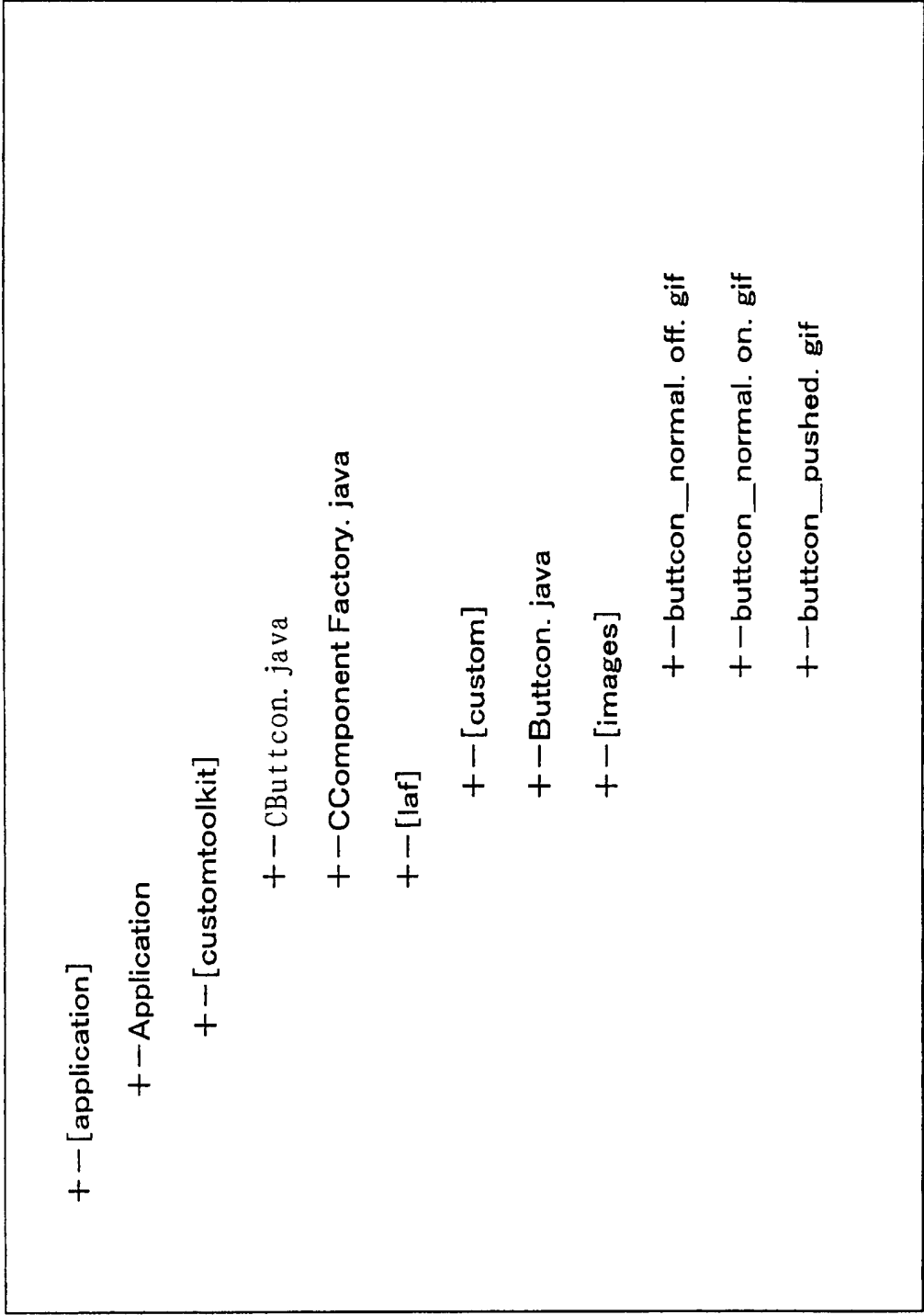


图 5

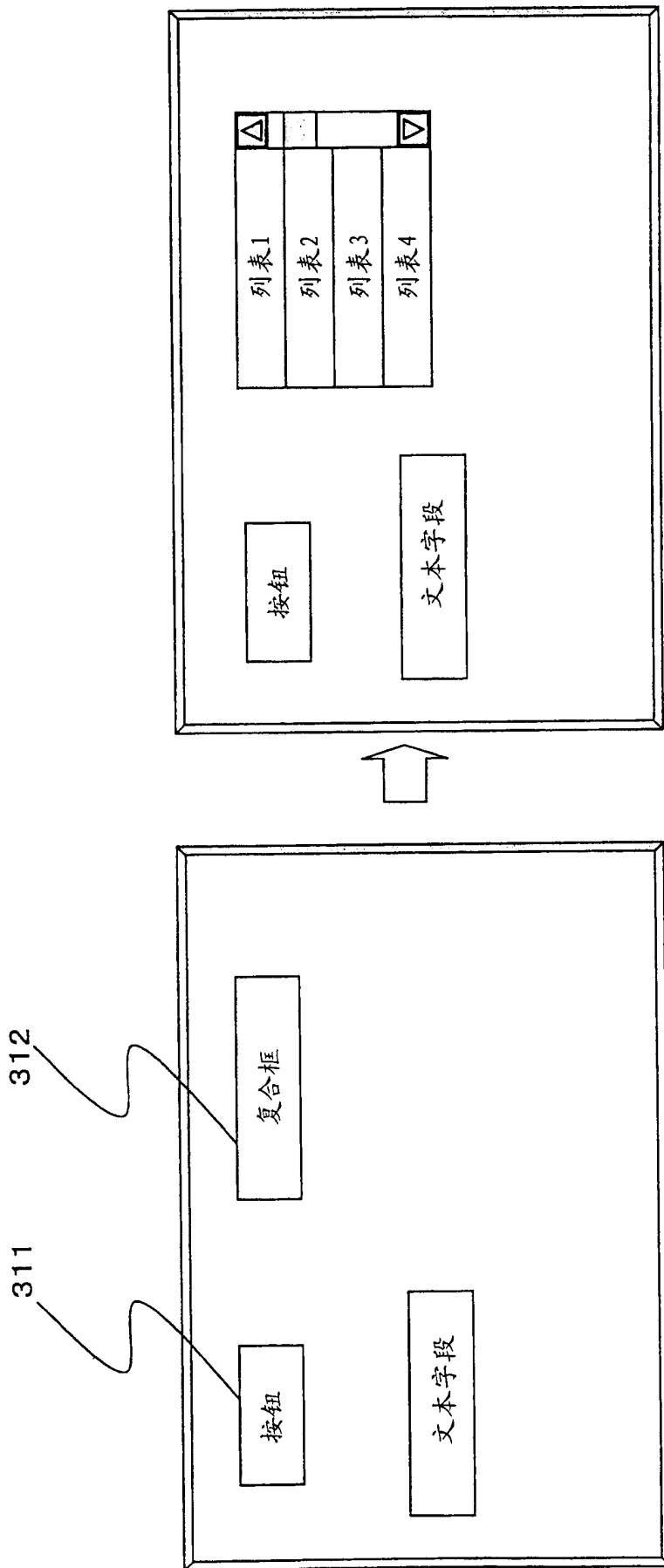
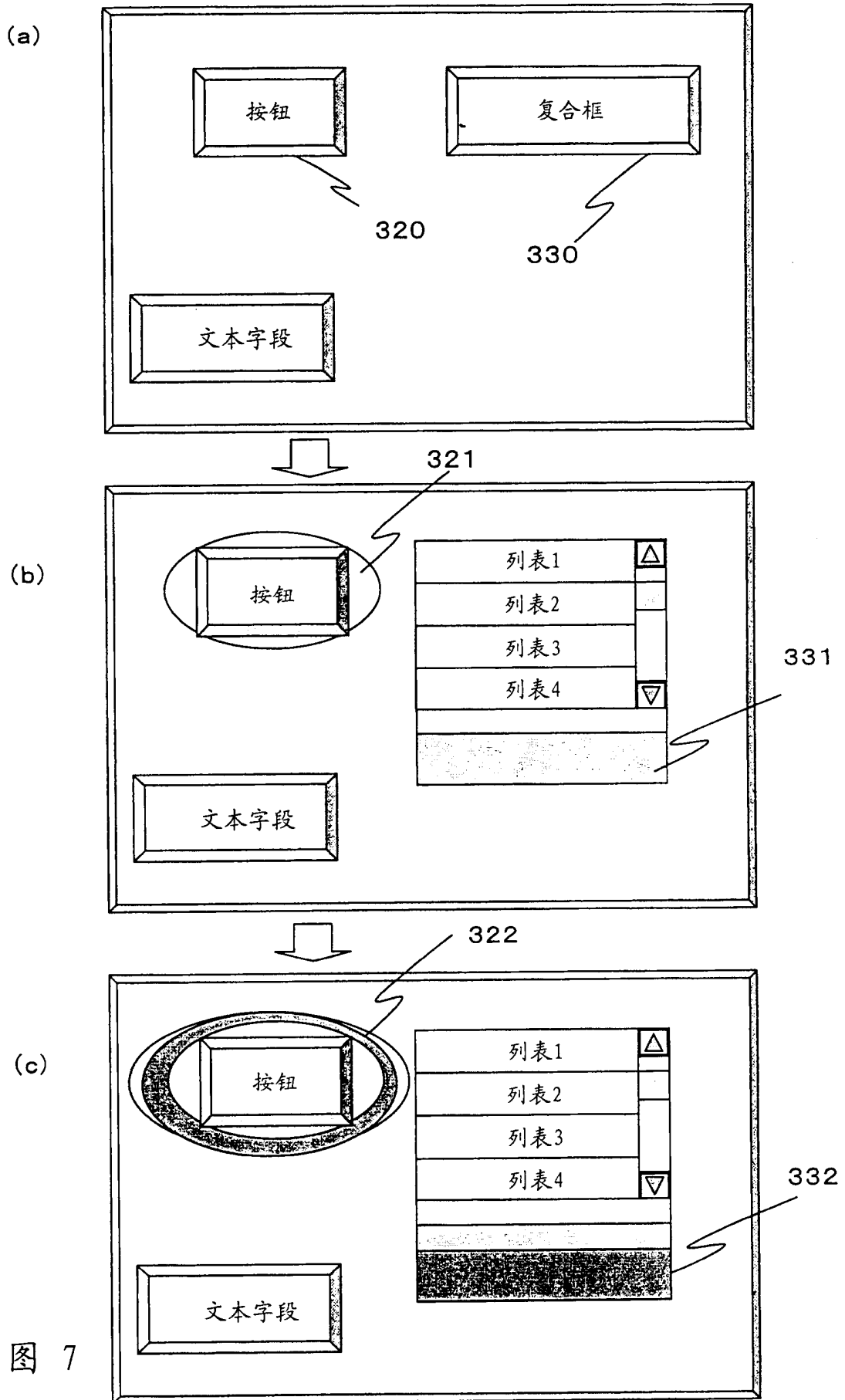


图 6



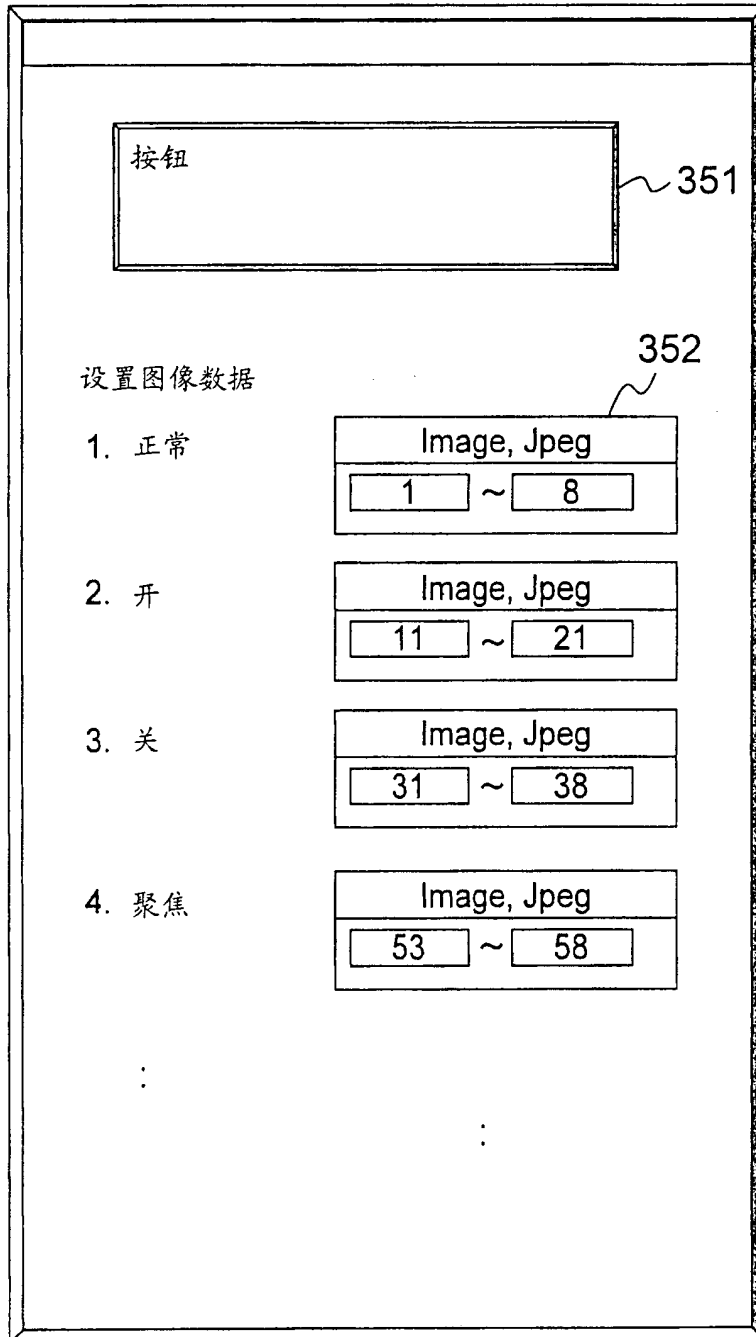


图 8

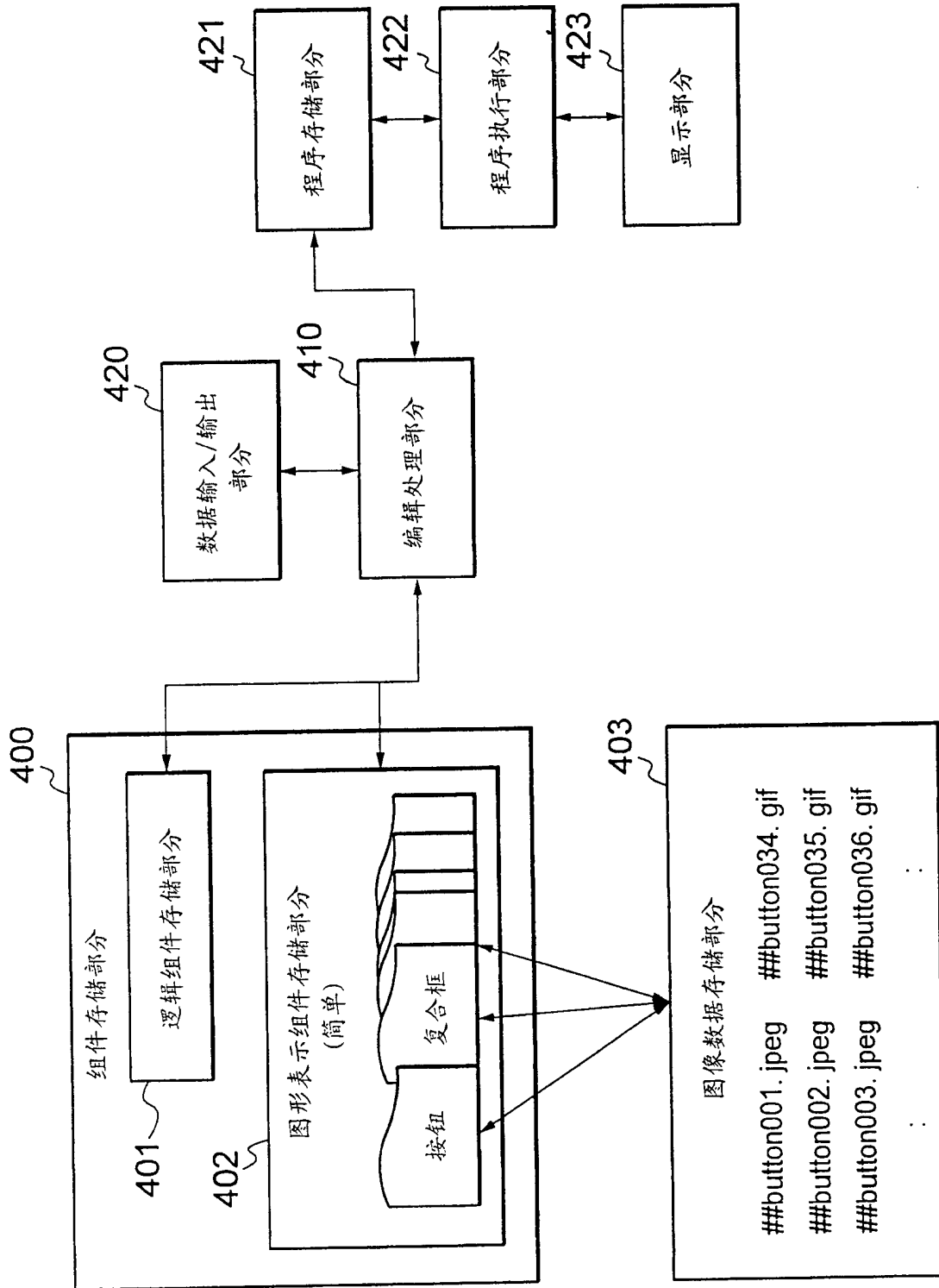


图 9

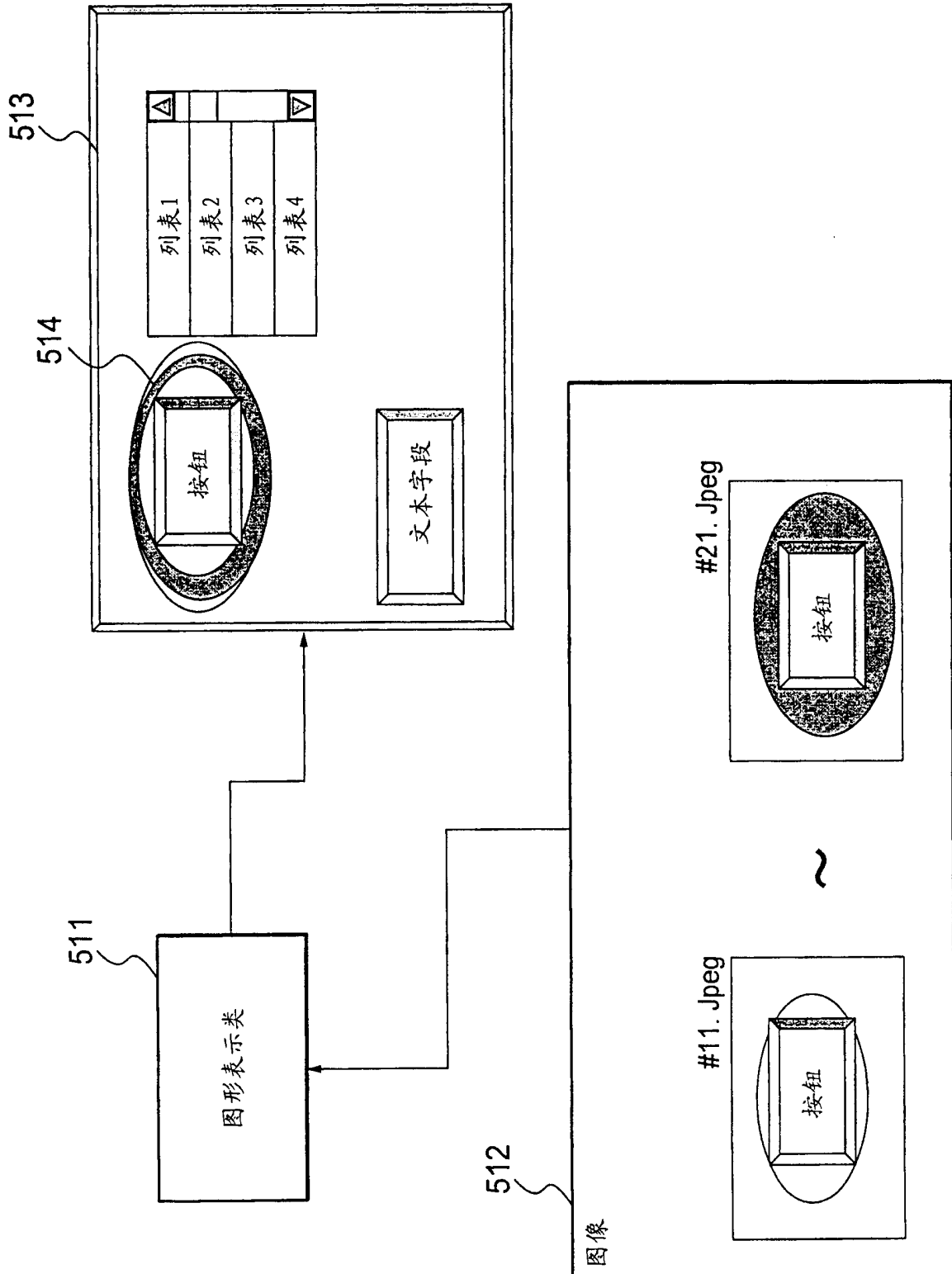


图 10

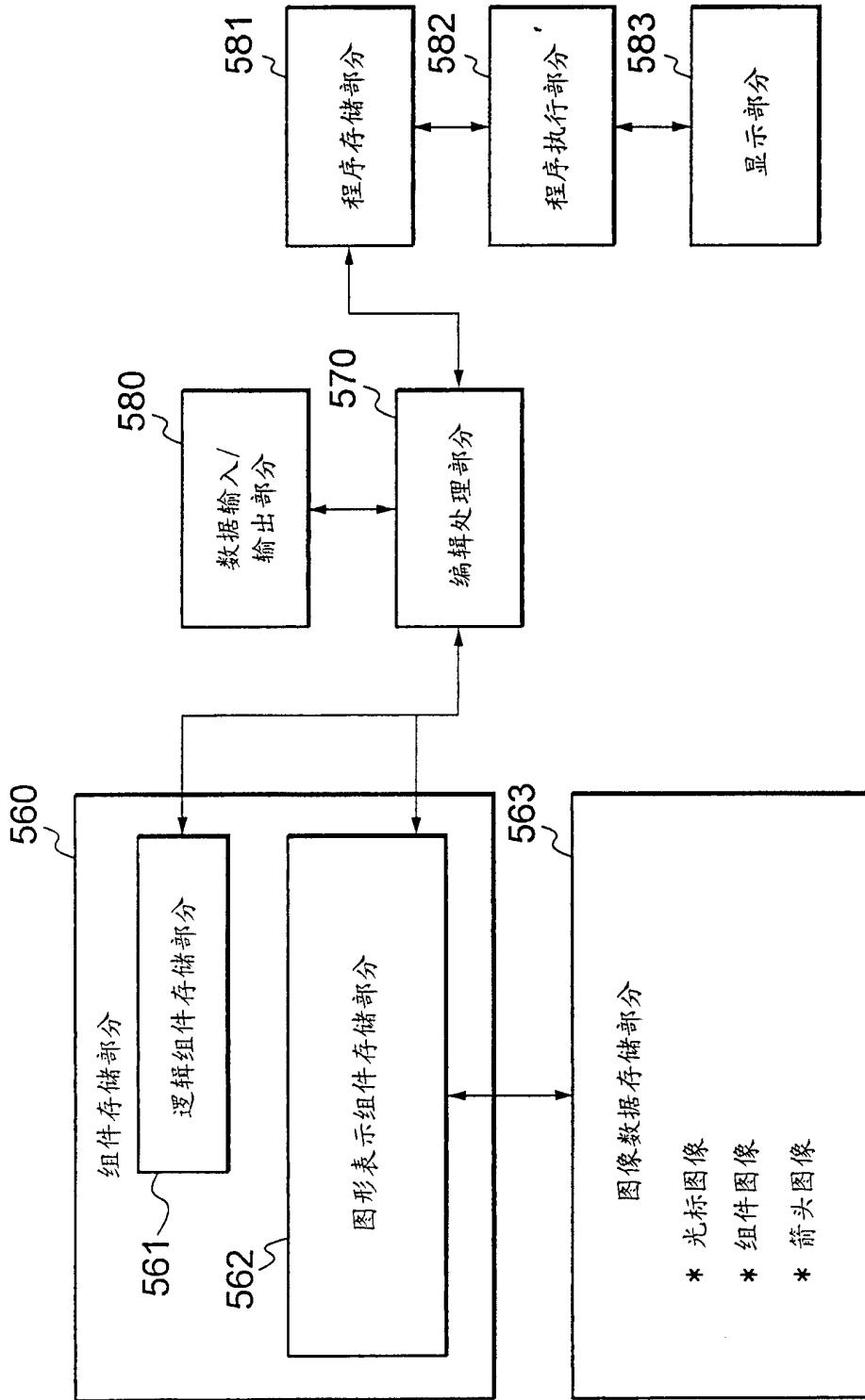


图 11

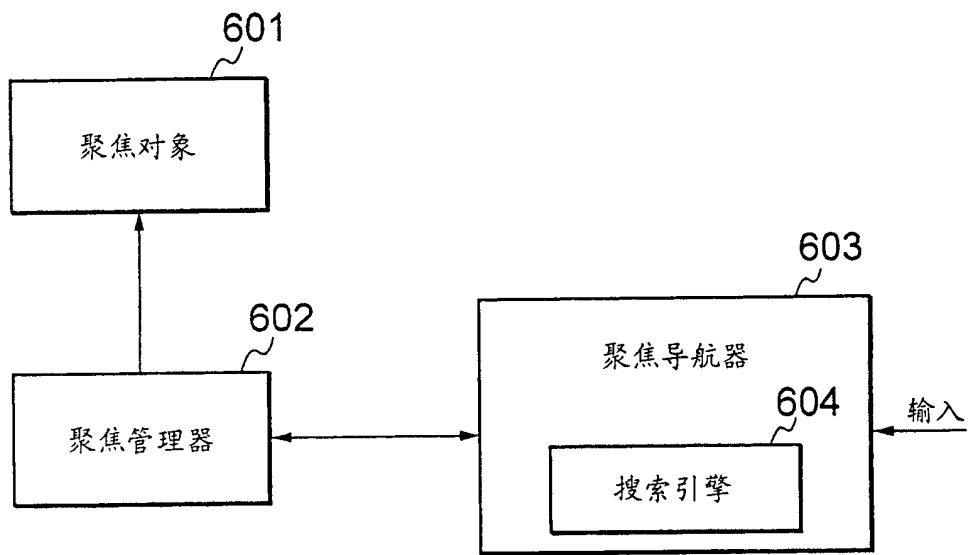


图 12

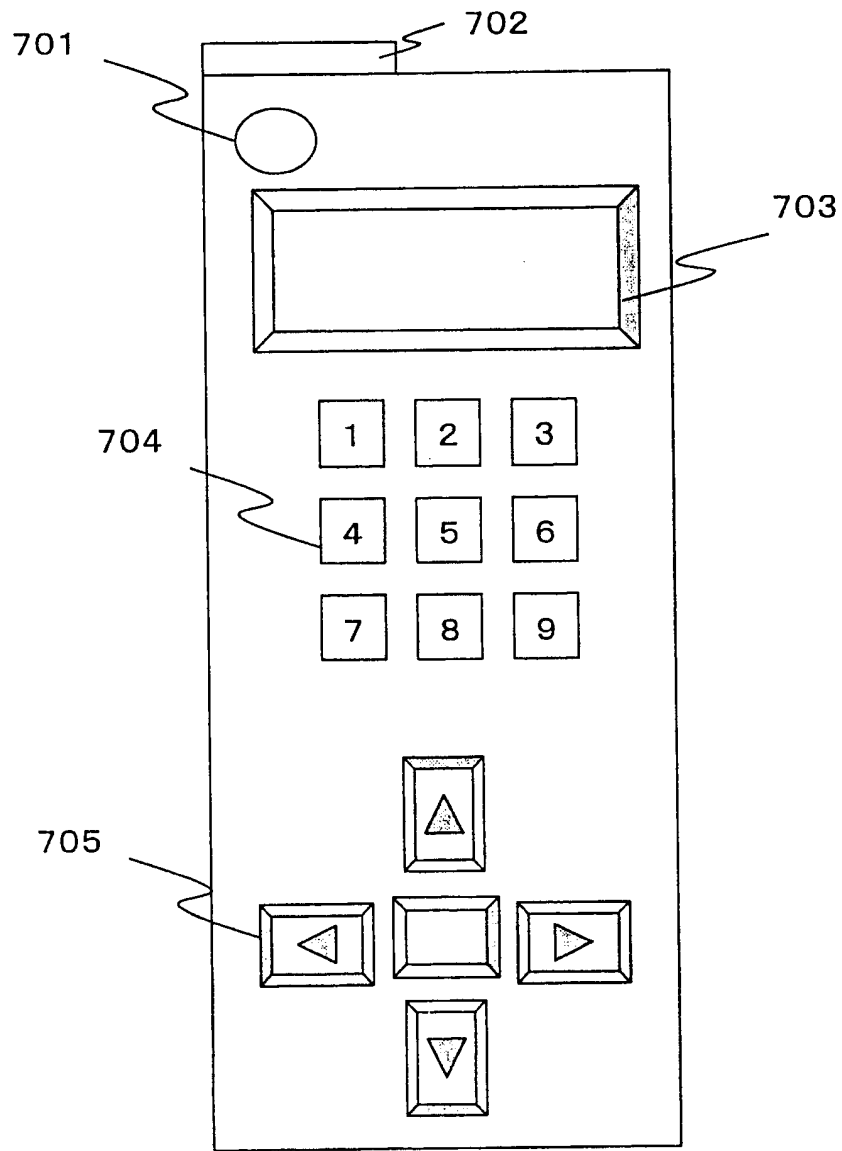


图 13

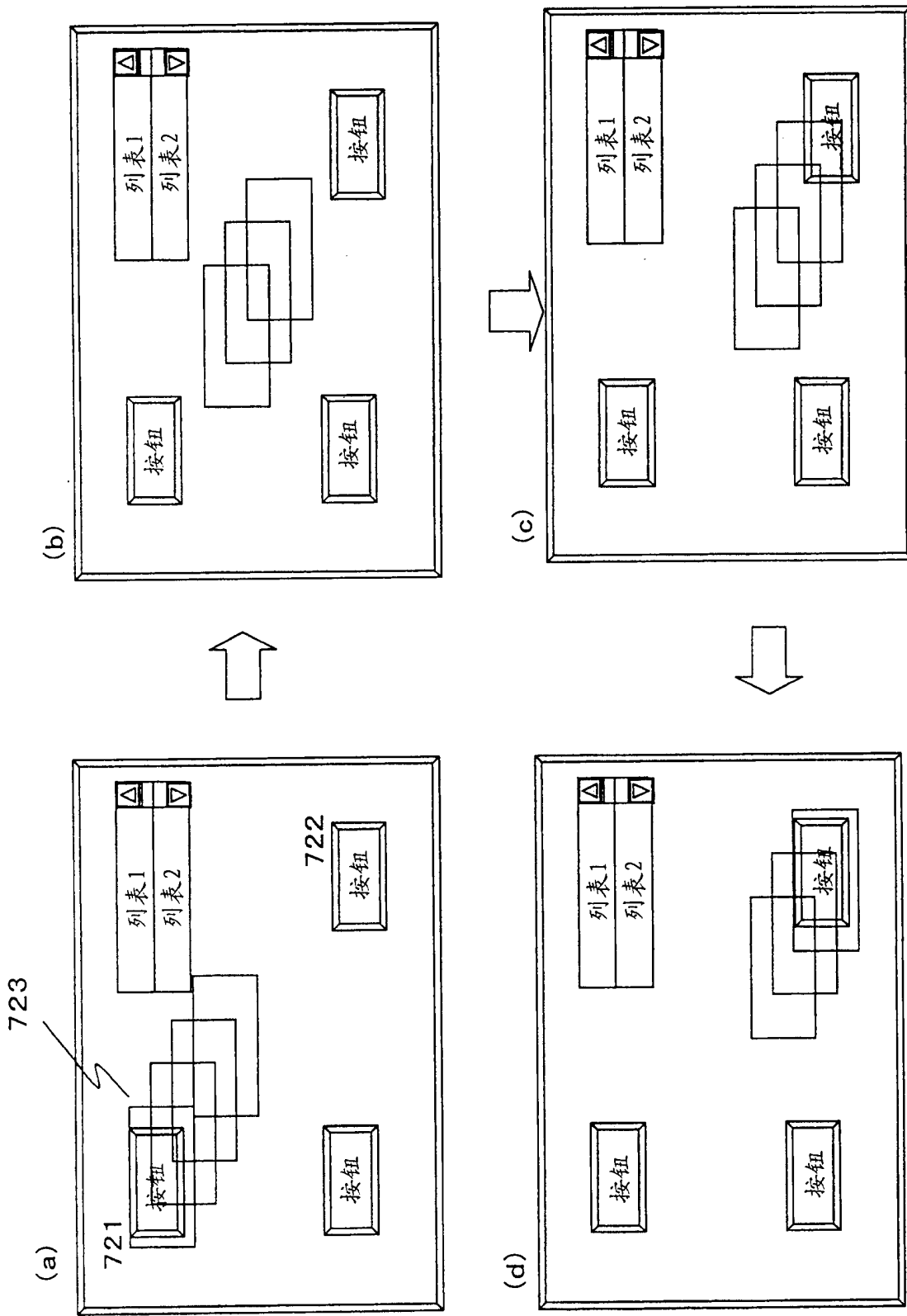


图 14

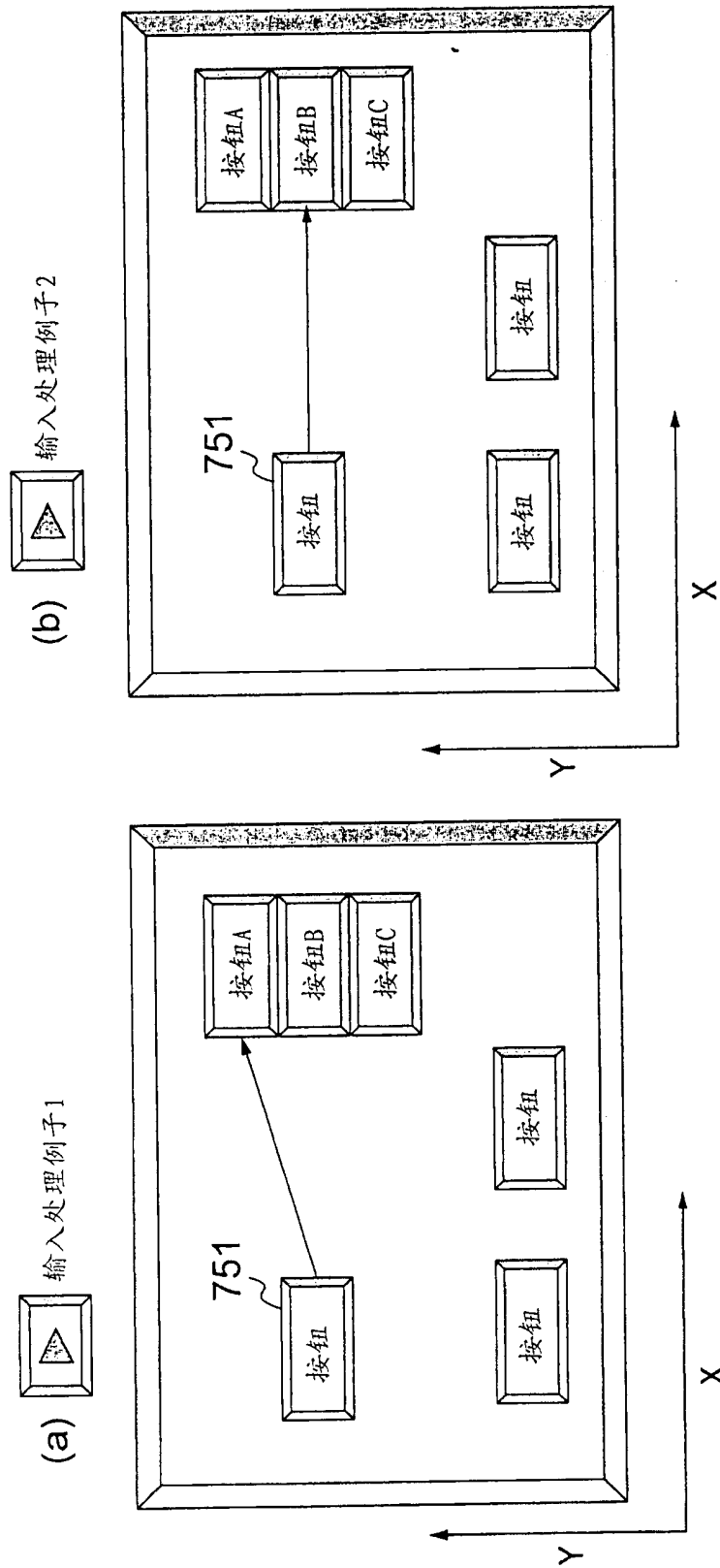


图 15

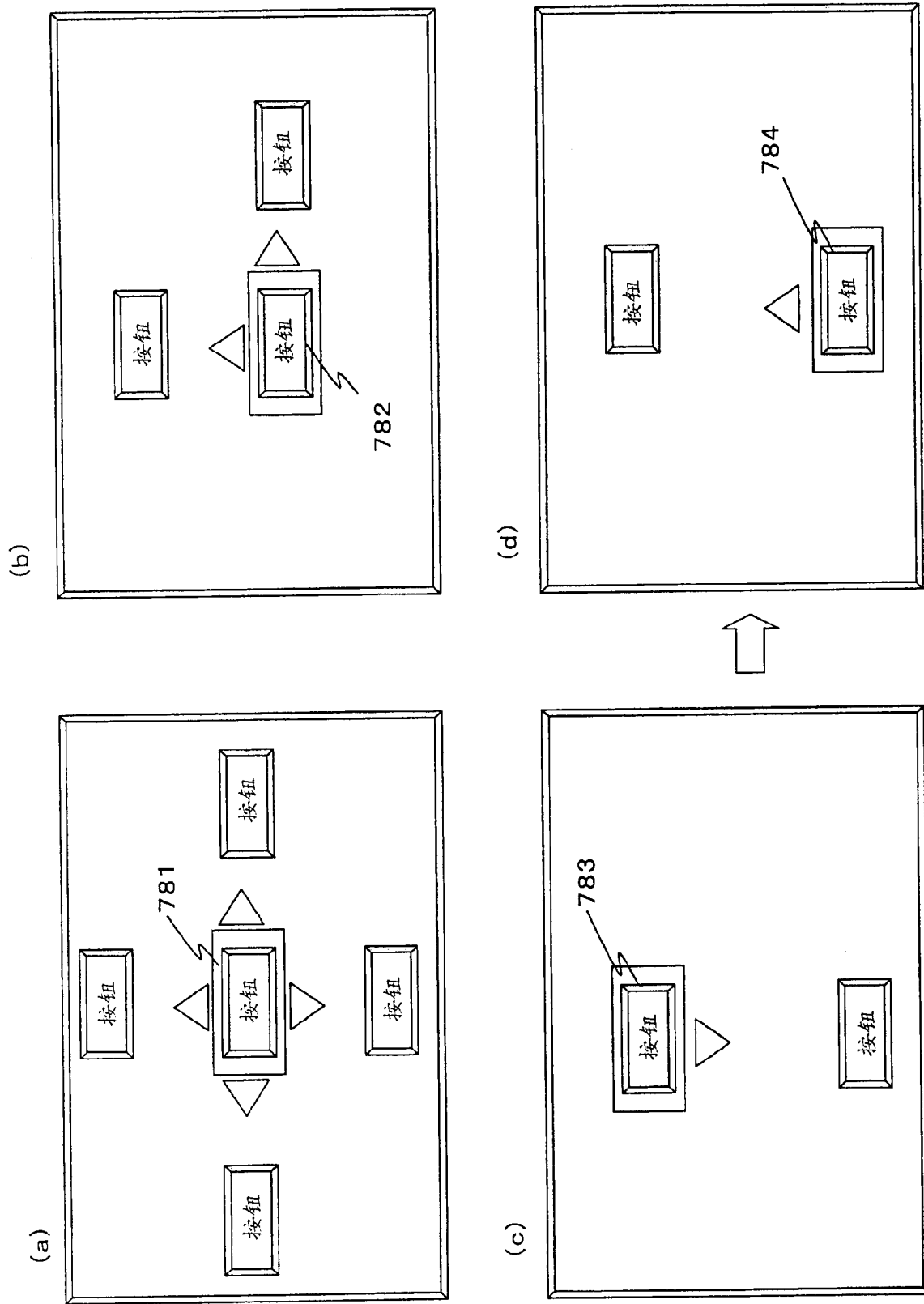


图 16

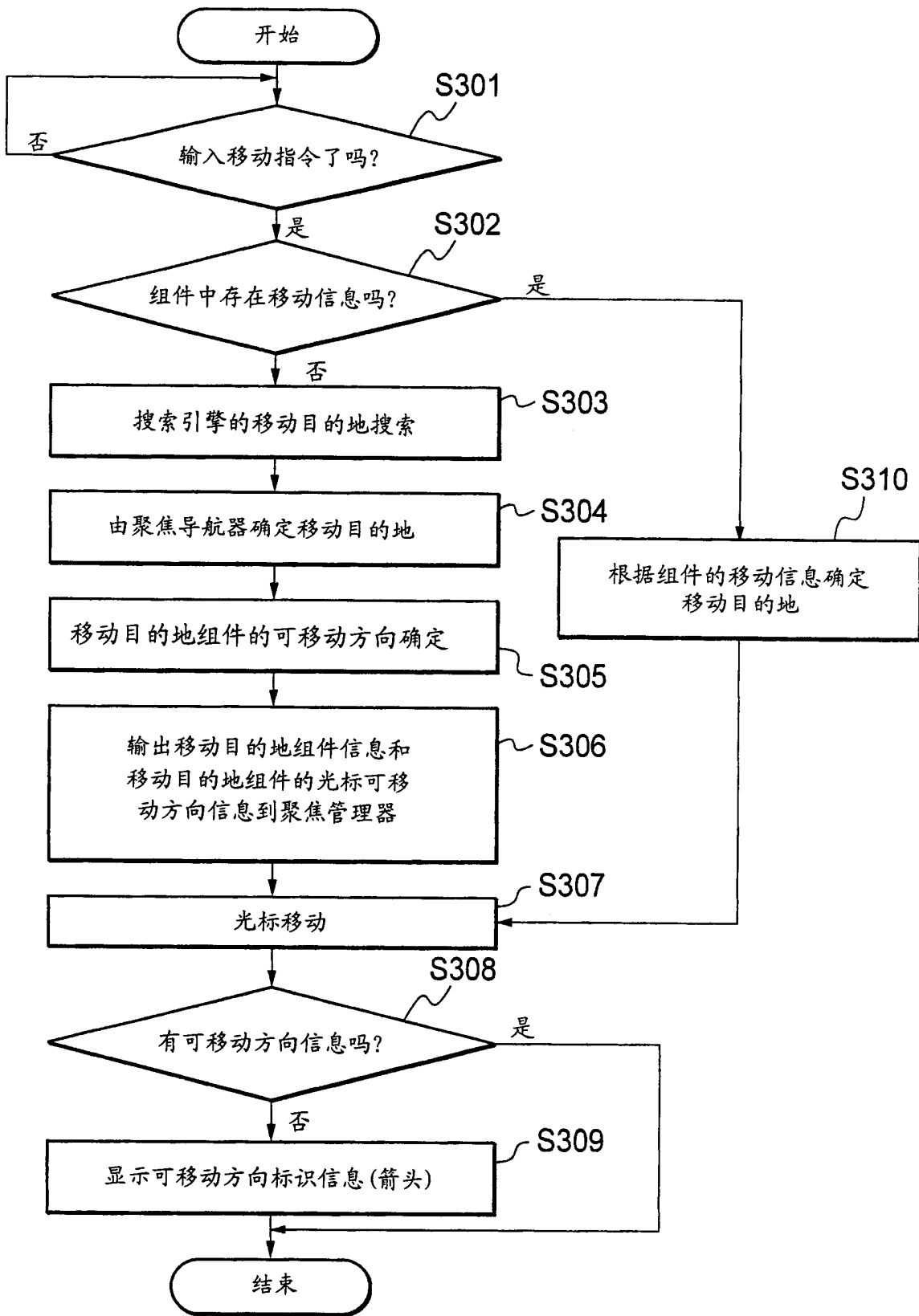


图 17

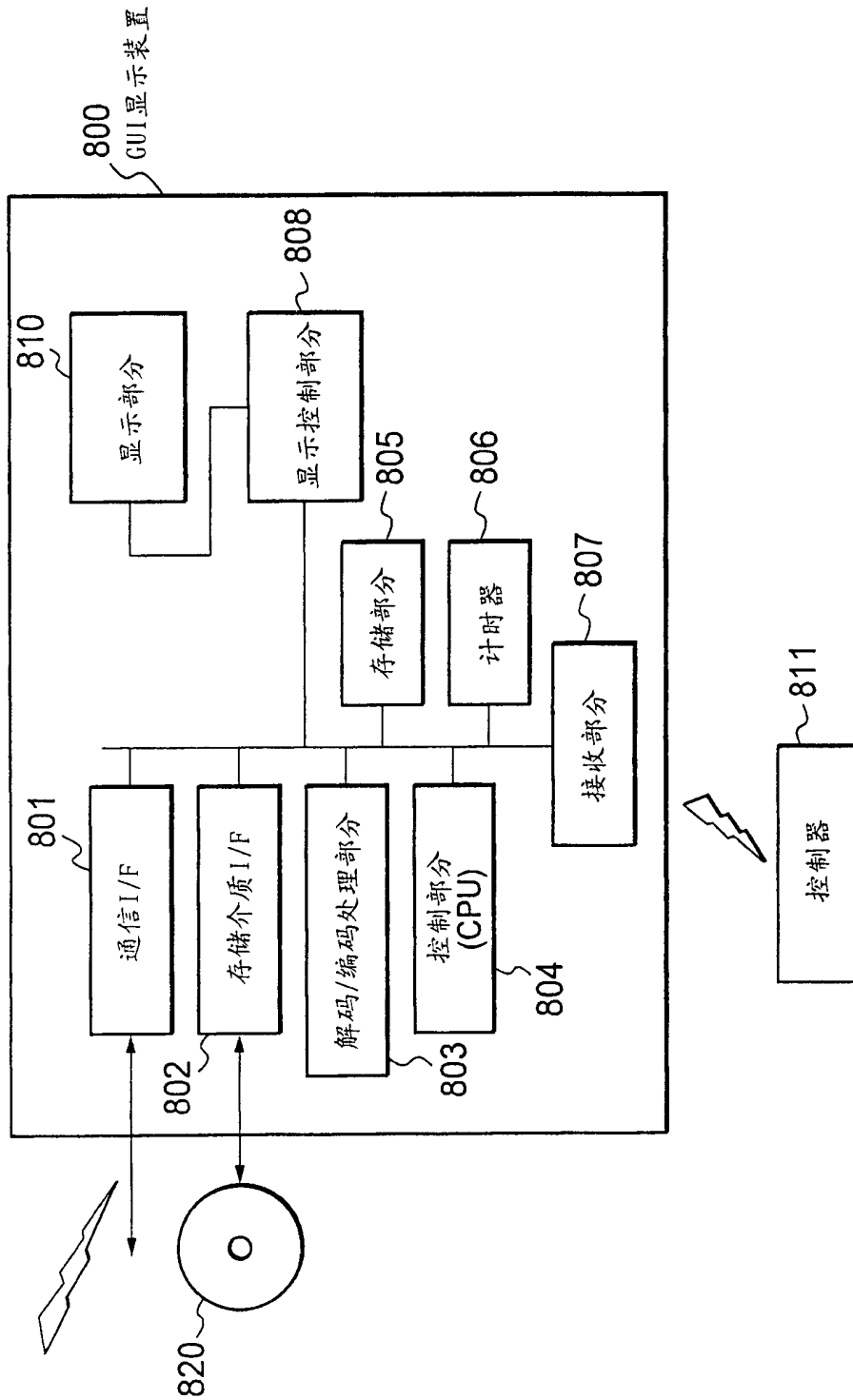


图 18