

[19] 中华人民共和国国家知识产权局

[51] Int. Cl.
G06T 17/00 (2006.01)



[12] 发明专利申请公布说明书

[21] 申请号 200780052188.2

[43] 公开日 2010 年 2 月 10 日

[11] 公开号 CN 101647045A

[22] 申请日 2007. 6. 28

[21] 申请号 200780052188.2

[30] 优先权

[32] 2007. 3. 15 [33] US [31] 60/918,131

[86] 国际申请 PCT/US2007/015149 2007. 6. 28

[87] 国际公布 WO2008/115196 英 2008. 9. 25

[85] 进入国家阶段日期 2009. 9. 15

[71] 申请人 汤姆森特许公司

地址 法国布洛涅 - 比扬库尔

[72] 发明人 拉尔夫·A·西尔伯斯坦

戴维·萨赫克

唐纳德·J·奇尔德斯

[74] 专利代理机构 北京市柳沈律师事务所

代理人 吕晓章

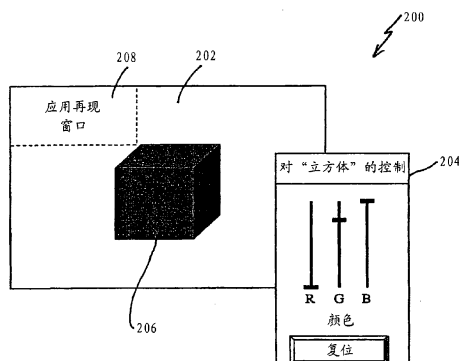
权利要求书 2 页 说明书 9 页 附图 5 页

[54] 发明名称

用于场景图中的参数的访问及控制的方法和系统

[57] 摘要

用于提供对场景图内的参数的访问和控制的方法和系统，包括重新定义场景图内的分量或节点的语义。要求分量或节点组(依赖于场景图结构)使得能够从应用用户界面来访问所选的场景图信息。在一个实施例中，产生用于控制场景图参数的用户界面。另外，可以实现允许或不允许访问某些场景图参数并限制其值的范围的约束。



1. 一种用于控制场景图的方法，包括：
重新定义场景图内的一组元素；
根据重新定义的元素访问场景图中的所选信息；以及
提供用于控制所选信息的手段。
2. 如权利要求 1 所述的方法，其中所述重新定义的一组元素包括依赖于场景图的结构至少一个分量或至少一个节点。
3. 如权利要求 1 所述的方法，其中所述提供用于控制的手段包括向用户界面提供与被访问的所选信息有关的控制。
4. 如权利要求 1 所述的方法，其中所述重新定义包括：
标识要向其提供控制的所述一组元素；以及
将与所标识的所述一组元素对应的控制信息链接到场景图中的现有结构。
5. 如权利要求 4 所述的方法，其中所述重新定义还包括添加一个或多个额外的分量或节点，以使得能够从应用用户界面访问所选的场景图信息。
6. 如权利要求 1 所述的方法，其中所述提供用于控制的手段包括对与场景图的用户交互定义至少一个约束。
7. 如权利要求 6 所述的方法，其中所述重新定义包括使用包含值的范围、增量和用户或平台简档中的至少一个的一组元素。
8. 如权利要求 1 所述的方法，还包括：
确定场景图是否提供了通用的字段类型；以及
当场景图未提供通用的字段类型时，使用场景图中的最大域用于对元素的所述重新定义。
9. 如权利要求 4 所述的方法，其中所述链接额外的控制信息包括：
寻找场景图中分量或节点的出现；
确定要控制的信息是否是所述分量或节点的一部分；
确定来自场景图的信息数据类型；以及
将数据类型和对数据的指向存储到用于接收用户事件的应用中。
10. 如权利要求 1 所述的方法，其中在使用场景图创作工具的场景图创作期间执行所述重新定义、访问和提供。

11. 如权利要求 1 所述的方法，其中在完成场景图的创作后执行所述重新定义、访问和提供。

12. 如权利要求 11 所述的方法，还包括：

从创作工具输出场景图；

向输出的场景图中输入控制；

保存控制被使能的场景图；以及

将控制被使能的场景图加载到应用中。

13. 一种用于控制场景图的系统，包括：

用于重新定义场景图内的一组元素的部件；

用于根据重新定义的元素访问场景图中的所选信息的部件；以及

用于控制所选信息的部件。

14. 如权利要求 13 所述的系统，其中所述重新定义的一组元素包括依赖于场景图的结构至少一个分量或至少一个节点。

15. 如权利要求 13 所述的系统，其中所述用于控制的部件包括具有与被访问的所选信息有关的控制的图形用户界面。

16. 如权利要求 13 所述的系统，其中所述用于重新定义的部件还包括用于标识要向其提供控制的所述一组元素的部件和用于将与所标识的所述一组元素对应的额外的控制信息链接到场景图中的现有结构的部件。

17. 如权利要求 16 所述的系统，其中所述重新定义部件还包括用于添加一个或多个额外的分量或节点以使得能够从应用用户界面访问所选的场景图信息的部件。

用于场景图中的参数的访问及控制的方法和系统

技术领域

本发明涉及场景图 (scencegraph), 更具体地, 涉及在场景图再现 (rendering) 期间的参数的访问 (accessibility) 及控制。

背景技术

场景图是由节点、也被称作对象组成的分级结构, 该节点可以是分组节点 (分支之间的交叉点) 或叶。这种结构通常用作存储 2D 或 3D 图形信息 (例如几何模型) 及其组成, 以便再现它们。

场景图可以包括通过时间插值方式或基于帧的动画 (animation) (例如在 Macromedia Flash 情况下) 而形成的动画信息。其还可以包含表示所允许的、与场景或者在场景中所包括的对象进行的用户交互 (User Interaction) 的传感器。必须在场景图内完全描述这种动画并且正确进行交互触发路由, 以便正确操作。

在计算机图形 (Computer Graphics) 领域中使用场景图已经普及了若干年。大多数 3D 创作工具 (例如 Autodesk 3D Studio 或 Maya) 和 3D 再现引擎 (例如 Pixar RenderMan) 内在地 (即在应用内部) 使用场景图作为数据结构 and 作为用于交换的文件格式 (例如 VRML、COLLADA)。在场景图再现期间的用户控制是相对有限的, 这是由于必须在场景图中嵌入用户控制的描述的事实。即, 不幸的是, 在当前可用技术中没有简单的方式用于在场景图内或在场景图再现期间对动画和参数定义直接应用用户界面控制。

发明内容

本发明的实施例提供了使能场景图的参数控制的简单方式和按应用级别处置这些控制并将其呈现给用户的方法。本发明的实施例还使得能够使用用于允许或不允许某些控制的值范围、增量、默认值和用户或平台简档 (profile) 来定义对用户交互的约束。

在本发明的一个实施例中, 用于控制场景图的方法包括: 重新定义场景

图内的一组元素；根据重新定义的元素访问场景图中的所选信息；以及提供用于控制所选信息的手段。

在本发明的替换实施例中，用于控制场景图的系统包括：用于重新定义场景图内的一组元素的部件；用于根据重新定义的元素访问场景图中的所选信息的部件；以及用于控制所选信息的部件。

附图说明

通过考虑以下结合附图的详细描述，能够容易地理解本发明的教导，在附图中：

图 1 绘出根据本发明实施例的、使能控制场景图参数的处理的流程图；

图 2 绘出根据本发明实施例的场景图及其相应控制的图形表示；

图 3 绘出根据本发明实施例的、用于场景图中参数的访问和控制的方法的流程图；

图 4 绘出根据本发明实施例的图 3 的方法的重新定义步骤的流程图；以及

图 5 绘出根据本发明实施例的图 4 的方法的链接步骤的流程图。

应当理解，附图是用于图示本发明的构思的目的，并不是必须用于图示本发明的唯一可能的配置。为了便于理解，在可能的情况下，使用相同的参考标记来指定图中共有的相同要素。

具体实施方式

已经定义了很多用于场景图描述的标准，包括 VRML（虚拟现实建模语言）、X3D（可扩展 3D）、MPEG-4 系统、COLLADA 和 Java3D。这种标准使用文本表示或 XML。另外，还存在一些特性（proprietary）标准（例如微软 DirectX 格式、NVIDIA NVSG 等）。但是，在所有这些表示中，用于使能用户交互的主要方式是在场景图内集成所有交互（传感器）和行为以将整个模型嵌入超结构中（例如 VRML/X3D 的原型），或者使用场景图应用界面（称作 VRML 的 EAI-外部应用界面、X3D 的 SAI-场景访问界面，DirectX 是用于微软格式的界面，MPEG-J 是 MPEG-4 系统中的界面）来开发特定应用。

如在首先列出的方案中，对场景图内的全部交互和控制的描述要求场景图的再现引擎包括执行引擎以处理动画和交互事件（见 VRML 手册中的描

述)。这是因为, 这些情况下的应用用户界面不具有场景图的知识。同样, 必须开发特定的源代码以便从场景图中提取数据和以便给出对其的用户访问。上述第二方案包括使用如在 VRML/X3D 中定义的原型(见 VRML 手册)。如在 VRML 手册中描述的, 该方案是“允许作者限制可以对对象做什么的保护机制”。但是, 问题是, 该“原型节点”技术不允许: 1) 简档管理; 2) 参数值的简单限制(在此情况下唯一可能的方式是使用脚本); 以及 3) 不为控制字段(field)提供对创建用户界面的任何简单的容纳(containment)(即, 在执行引擎中, 除了用于处理事件的标识符名称之外, 没有给予参数的真正名称, 并且原型技术仅有助于场景内的对象(节点)而不是用户之间的交互)。另外, 使用该技术不允许简单添加要呈现给用户的新参数, 这是因为“原型节点”向用户隐藏了实际的场景图结构。

上述第三种技术使用诸如用于场景图访问界面(SAI)的 X3D 规范之类的 SAI。将这种界面中的每一个与语言绑定(即 VRML EAI 是 Java 绑定的)。然后, 该技术在此用于开发特定用户界面, 以得到对场景图信息的访问。该技术允许对场景图的任意组件(member)的访问。但是, 访问所有字段意味着用户必须知道整个场景图结构, 并具有其所有结构要素(节点、字段等)的知识。此外, 场景图中的名称对于用户可能不是足够清楚, 这是因为其语法成分(composition)的名称例如不允许空格字符。

如果实施与所选取的场景图文件格式语法匹配, 则本发明的实施例可以应用于任何现有的场景图描述语言(例如文本的或 XML)。然后, 解析器可以决定处置或不处置由当前原理定义的节点, 以允许或不允许本发明的特征。应用还可以选取不显示由当前原理定义的用户界面控制。例如, 由于所有节点都具有有时表示为括号(例如“{”和“}”)或 XML 标签括号(例如“<node>”和“</node>”)的边界(bound), 因此, 不处置本发明的新特征对解析方将是非常直接的。然后, 解析器仅需要忽略场景图中的完全的特征陈述, 限于其括号。

本发明的实施例包括一组附加分量(component)或节点(依赖于场景图结构), 其是使能对从应用用户界面中选择的场景图信息的访问而不修改原始场景图描述所需要的。新分量或节点例如利用关于用户界面的可理解的名称和修饰语(modifier)来使能重新定义可呈现给用户的目标场景图信息的语义(semantic)。然后, 由通过其图形用户界面提供给用户的、新创建的图形界

面窗口小部件 (widget) 来控制该信息。

例如, 在本发明的一个实施例中, 可以利用符合场景图语义而定义的匹配的闭合和打开的括号来表示特征标识符。另外, 可以由描述简档的字符串或标识符的列表来表示简档标识符。简档可以包括例如描述访问权的用户或例如描述性能和/或简档的平台。对这些简档标识符字符串的容纳可以是依赖于应用的, 或者是名称描述或数字值。如果该字符串列表是空, 则对于所有用户或平台, 所有定义的控制都将可用。但是, 本发明的替换实施例的某些实施可以根本不使用该信息 (即不在如像解码器 STB-机顶盒-之类的平台上呈现简档的情况), 而其他实施例可以在应用中单独的层上管理简档 (即在如像交换机 (switcher) 的平台中的静态简档的情况下)。本发明的实施例中的每一个还可以包括用于场景图中的控制项的列表。

场景图中的标识符包括定义场景图中的目标信息的字符串或标识符的列表, 并通常由要控制的节点名称定义或标识符和节点字段名称或标识符组成。如果在场景图的当前解析级别中未发现标识符, 则不应该激活其控制。然后, 应用可以检索字段值类型 (例如浮点、整数) 作为在场景图中的其定义的结果。该字段实质是在单一用户界面组件的情况下允许分组修改的字符串、例如同时的几个颜色字段的列表。因此所列字段必须是相同类型。

本发明的实施例可以进一步包括出现在应用用户界面上的名称。这通常不同于在前面段落中提到的标识符名称, 这是由于, 通常场景图中的标识符是在语法上受语义限制的 (没有空格或特定字符), 并且可以不呈现适当的描述 (即这些字符通常是由图形建模工具自动产生的)。

根据本发明的实施例, 复位或出厂值可以包括用于当在用户界面上激活“复位”或“出厂值”按钮时设置参数的值。该值可以不同于场景图本身中的值, 这是因为场景图存储特定时刻的值。而且, 可以包括定义对于参数有效的值的范围的最小值和最大值。最小值和最大值定义了对用户界面控制的限制。

本发明的实施例的应用还可以包括: 步长增量 (Step Increment), 其对应于通过例如按下“+”或“-”按钮而在用户界面上 (正或负地) 应用参数的逐步增量 (绝对值); 以及对其使能该控制的所定义的简档。即, 可以在此列出对其使能该控制的所选的简档标识符 (例如以上列出的那些或者其他应用定义的标识符)。

在本发明的一个实施例中，所有值被表示在字段类型数字域中。如果所选取的场景图不提供通用的字段类型来表示例如浮点值（Floating values）和布尔值（Booleans），则最大的域可用于定义那些参数。例如，可以使用浮点表示，并且可以将值计算（cast）为整数或布尔值。根据本发明的实施例，所有值可以是数字的数组，这是因为很多参数可以由不止一个元素或分量构成，例如具有 RGB（红色、绿色和蓝色）值的颜色。如果构成所控制的字段的元素/分量的数量与最小值、最大值、复位和增量值中所包含的元素/分量的数量不同，则应该仅使用与所控制的字段中的元素/分量的数量的匹配的、关于那些最小值、最大值、复位和增量值中元素的最小数量。例如，如果所控制的字段表示具有 3 个分量的 RGB 颜色并且“复位”值被定义为“[1.0 1.0 1.0 1.0]”，则系统将仅使用前三个元素，留下最后一个未被使用。

如果包含所控制的字段的节点在场景图的执行期间被毁坏，则也可以使链接到该节点的用户界面控制去激活（deactivate）或毁坏。本发明的实施例还允许对所控制的项（item）分组。这允许以更自然的方式来对关于用户界面的参数列表进行分组、例如“汽车”由“车轮”构成，或允许超越（override）用于向用户呈现信息的场景图自然分级结构。也可以命名该分组特征，并且该分组特征可以递归地包含其他组。

可以在任意现有的场景图上实现本发明的实施例，唯一的限制是那些对每个场景图模型固有的限制。例如，VRML2.0 不提供通用的字段类型，因此仅“SFFloat”可用作最大的域（domain）。

例如，本发明的 VRML2.0 节点实现可以表示如下：

```
ControlProfiles{  
    field MFString profiles []  
}
```

如果在应用层定义了简档，则可以丢弃“ControlProfiles”（“控制简档”）节点。

```
ControlGroup {  
    field SFString name "car"  
    field MFNode items []  
    field MFString profiles []  
}
```

“items”（“项”）字段可以包含“ControllItems”（“控制项”）列表（参

见以下)或其他“ControlGroup”(“控制组”)节点。“name”(“名称”)字段定义要呈现给用户的组名称,例如“汽车”。所呈现的“profiles”(“简档”)字段超越了关于“items”列表中的各个项的简档指定,允许简档容易地控制整个组。

```
ControlItem {
    field MFString identifiers []
    field SFString name ""
    field MFFloat resetValue []
    field MFFloat minValue []
    field MFFloat maxValue []
    field MFFloat stepIncrement []
    field MFString profiles []
}
```

在上述实施例中,“profiles”列表包含如由“ControlProfiles”节点或依赖于应用的值所定义的简档的列表。

在本发明的替换实施例中,VRML2.0实现内的PROTO结构概念可以表示如下:

```
CONTROLGROUP [
    field MFString profiles []
    field fieldtypename "name" IS [identifiers]
    [resetValue] [minValue] [maxValue]
    [stepIncrement] [profiles]
...
]
{
    # Internal Scenegraph section
}
```

在本发明的另一替换实施例中,本发明的X3D节点实现可以表示如下:

```
<ControlGroup name="" profiles=[]>
    <ControlItem identifiers=[] name="" resetValue=[]
    minValue=[] maxValue=[] stepIncrement=[]
    profiles=[]/>
</ControlGroup>
```

图1绘出根据本发明的一个实施例的使能场景图参数的控制的处理的流程图。图1的处理100以包括本发明的上述实施例的控制方法的创作或建模工具102开始。在步骤103,内容创建器输出主题场景图。如果场景图已经

使能控制，则处理继续到步骤 108。如果创作/建模工具未向场景图提供这种功能，则可以创建特定的模型以便随后添加控制。例如，当创作建模工具 102 不提供这种功能时，创作者将场景图 104 输出到模块 106，用于添加所需控制。然后，带有控制的场景图被使能 108，并被加载到应用块 110。然后，应用 112 加载场景图以将其再现 114，并且现在可以创建关于图形用户界面的控制 116（并使得其对用户可用）。

在本发明的实施例中，应用 112 可以直接嵌入所创建的控制模块 116，以添加用户控制。对于内容创建器检验正确的执行以及场景图的再现及其相应控制，这可以是有益的。

图 2 绘出根据本发明的实施例的场景图的图形表示和相应控制。图 2 的实施例示出通过支持本发明的方法的应用来再现场景图 200（在此情况下的简单立方体）及其控制（在此情况下的简单 RGB 颜色选择）。场景图包括白色背景 202、显示在应用再现窗口 208 上的淡蓝立方体 206 以及关于 VRML 节点实施例的解释的、由应用创建的控制窗口 204。使用 VRML2.0 场景图来写入图 2 中使用的场景图例子，其中简档被去激活。根据本发明的一个实施例，场景图的 VRML2.0 描述可以表示如下：

```
Shape {
  geometry Cube { }
  appearance Appearance {
    material DEF MAT Material {
    }
  }
}
ControlGroup {
  name "Cube"
  items ControlItem {
    identifiers "MAT.diffuseColor"
    name "Color"
    resetValue [ 0.0 0.0 0.0 ]
    minValue [ 0.0 0.0 0.0 ]
    maxValue [ 1.0 1.0 1.0 ]
    stepIncrement [ 0.1 0.1 0.1 ]
    profiles [""]
  }
}
```

如在以上例子中绘出的，应用 112 使用控制项来建立来自控制组“Cube”（“立方体”）的名称“Color”（“颜色”）与场景图字段“diffuseColor”（“散布颜色”）（“MAT”节点的一部分，其类型是 VRML2.0 规范中的 SFCOLOR）之间的绑定，因此在功能上支持所期望的控制（即用于红色、绿色和蓝色级别的 3 个可编辑字段）。这由图 2 中的控制窗口 204 示出，其中示出了各个 R、

G 和 B 控制。

图 3 绘出根据本发明的实施例的用于场景图中的参数的访问和控制的方法的流程图。图 3 的方法 300 开始于步骤 302，在步骤 302 中初始地重新定义场景图中的一组元素。然后，方法前进到步骤 304。

在步骤 304，一旦重新定义，就根据重新定义的元素来访问场景图中所选的信息。然后，方法前进到步骤 306。

在步骤 306，当已访问了所选信息时，提供用于控制所选信息的手段。

图 4 绘出根据本发明的实施例的图 3 的方法 300 的重新定义步骤 302 的流程图。为了重新定义如在图 3 的方法 300 中定义的元素，初始地在步骤 310 标识要向其提供控制的该组元素。在步骤 312，一旦标识，就将额外的控制信息链接到与场景图中的现有结构的相应的标识的一组元素。

图 5 绘出根据本发明的实施例的图 4 的方法的链接步骤 312 的流程图。在图 5 的实施例中，为了链接额外的信息，在步骤 320，初始地寻找场景图中的分量或节点的出现。在步骤 322，然后确定要控制的信息是否是该分量或节点的一部分。一旦做出确定，就在步骤 324 对来自场景图中使用的标准语法的信息数据类型做出另一确定。在步骤 325，随着做出的所有上述确定，可以将类型和对数据的指向（point to data）存储在用于接收用户事件的应用中。

即使未参考特定方式或仅使用一种方式做出以上描述，也可以按各种方式中的一种或多种来实现各个方面、实现和特征。例如，可以使用例如方法、装置、用于执行方法、程序或其他指令集的装置或处理设备、包括程序或指令集的装置以及计算机可读介质中的一个或多个来实现各个方面、实现和特征。

装置可以包括例如离散或集成的硬件、固件和软件。作为例子，装置可以包括例如处理器，其指的是通用的处理设备，包括例如微处理器、集成电路或可编程逻辑器件。作为另一例子，装置可以包括具有用于实行一个或多个处理的指令的一个或多个计算机可读介质。

计算机可读介质可以包括例如软件载体或诸如例如硬盘、紧致磁盘、随机存取存储器（“RAM”）或只读存储器（“ROM”）之类的其他存储器件。计算机可读介质还可以包括例如格式化的电磁波编码或传输指令。指令可以是例如在硬件、固件、软件中或者在电磁波中。可以在例如操作系统、单独的

应用或两者组合中得到指令。因此，处理器的特征可以在于例如被配置用于实行处理的器件和包括具有用于实行处理的指令的计算机可读介质的器件。

已经描述了用于提供对场景图内的参数的访问和控制的方法和系统的优选实施例（其旨在例示而非限制），注意，根据以上教导，可以由本领域技术人员做出修改和变更。因此，将理解，可以在如由所附权利要求所概括的本发明的范围和精神内、在所公开的本发明的具体实施例中做出修改。尽管前文专注于本发明的各种实施例，但是也可以在不脱离其基本范围的情况下设计本发明的其他和进一步的实施例。

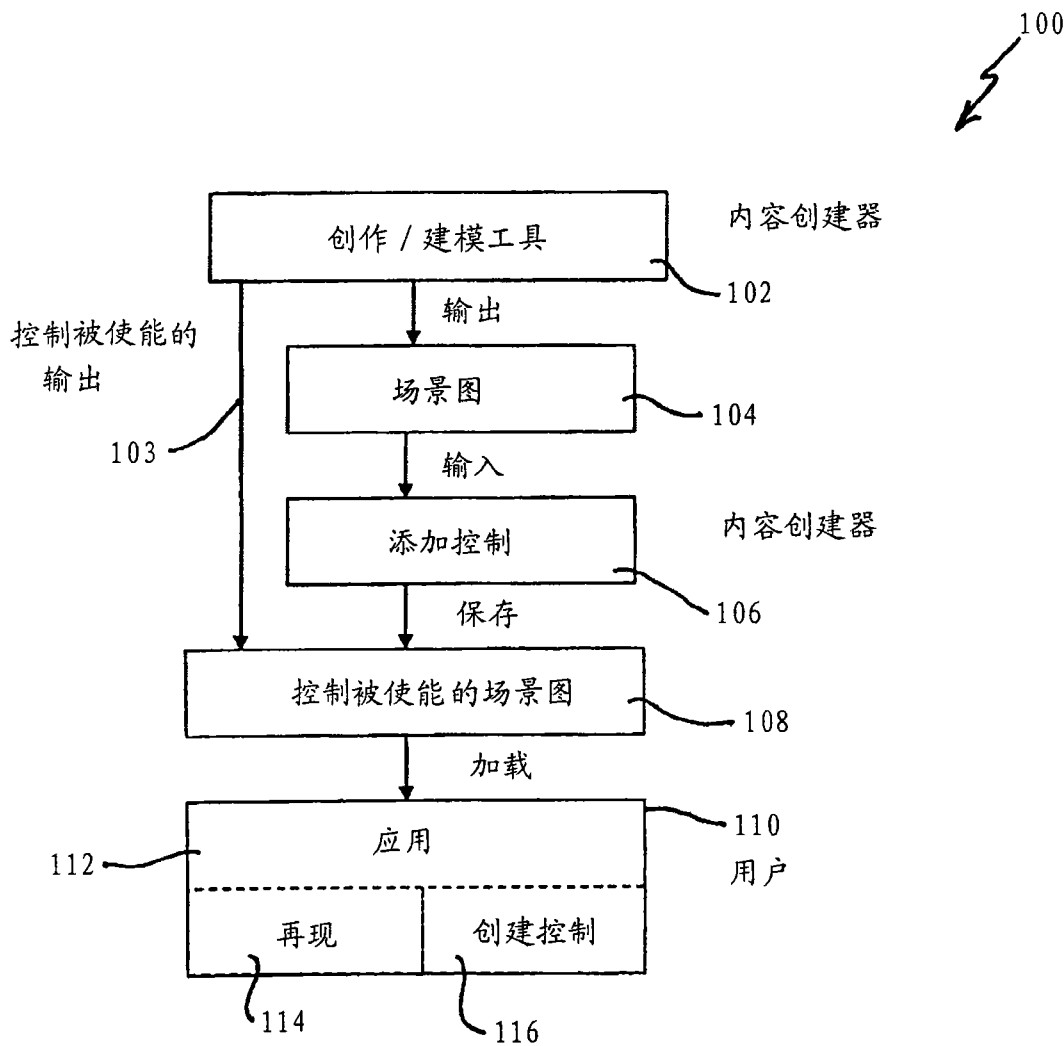


图 1

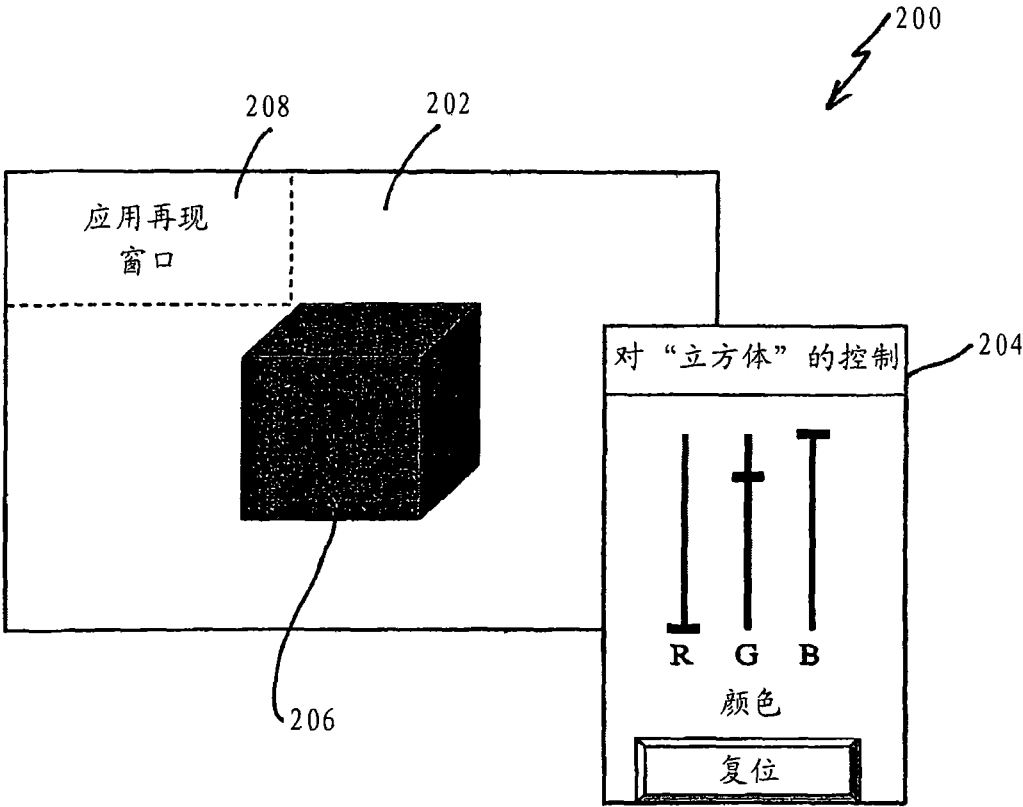


图 2

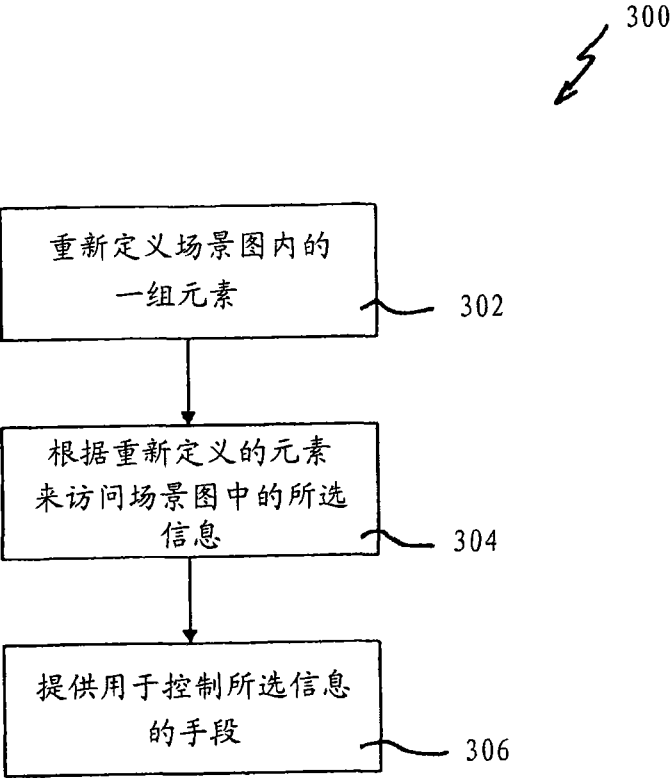


图 3

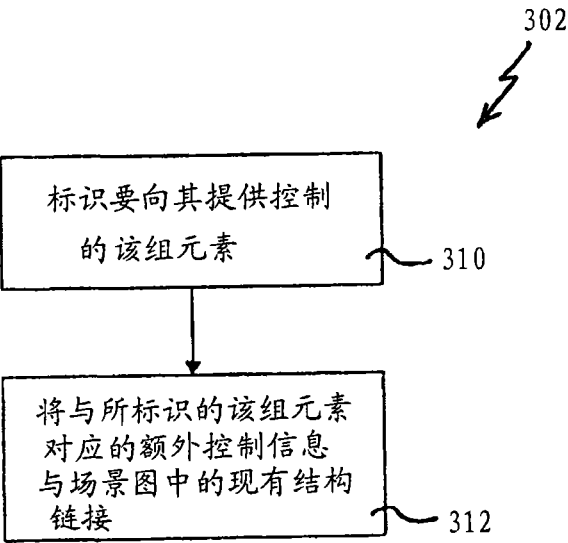


图 4

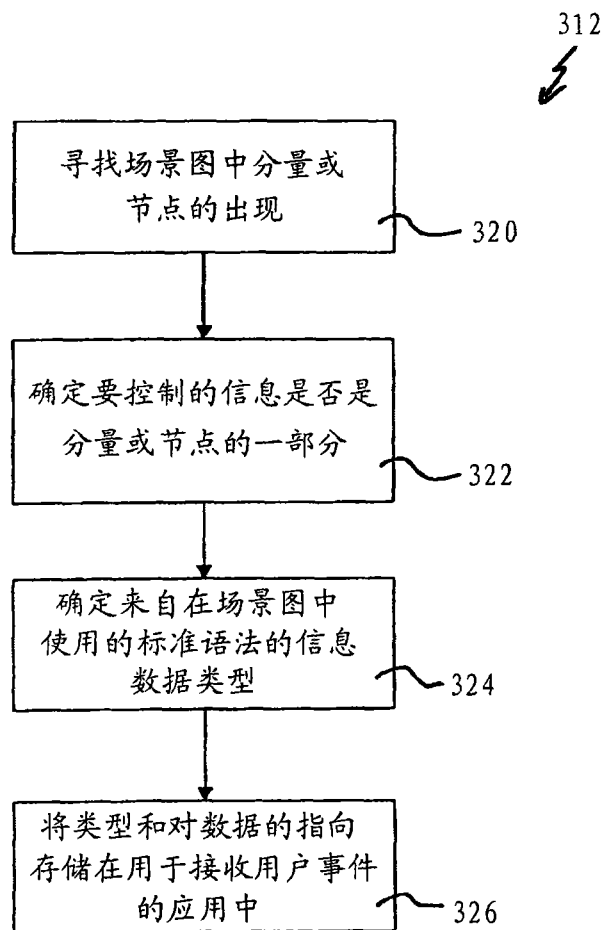


图 5