



[12] 发明专利说明书

专利号 ZL 97196774.1

[45] 授权公告日 2007 年 5 月 30 日

[11] 授权公告号 CN 1319026C

[22] 申请日 1997.8.1 [21] 申请号 97196774.1

[30] 优先权

[32] 1996.8.2 [33] US [31] 08/691,996

[86] 国际申请 PCT/US1997/013655 1997.8.1

[87] 国际公布 WO1998/008192 英 1998.2.26

[85] 进入国家阶段日期 1999.1.26

[73] 专利权人 微软公司

地址 美国华盛顿

[72] 发明人 迈克尔 F·科恩 贺立为

戴维·塞尔辛

[56] 参考文献

US 5517663A 1996.5.14

WO 9303453A1 1993.2.18

审查员 姚梦琦

[74] 专利代理机构 中原信达知识产权代理有限责
任公司

代理人 余 朦 穆德骏

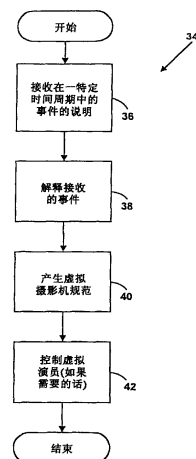
权利要求书 5 页 说明书 36 页 附图 17 页

[54] 发明名称

虚拟电影摄影方法和系统

[57] 摘要

通过自动地应用一般活动画面所用的电影摄影规则实现了在计算机产生的一个三维虚拟环境中的通信。从被解释的电影摄影事件(38)产生摄影机画面规范(40)。在虚拟现实和其它交互式应用中使用了一个称为虚拟电影摄影应用模块(VC)(50)的实时摄影机控制器,以改善固定视点镜头或顶棚安装的摄影机。在智能代理用户界面中使用 VC(50)以使用户能在一处于看来自然的摄影机位置利用代理看到他们自己。



1. 一种用于虚拟电影摄影的方法，该方法包括：

用实时应用模块实时地接收电影摄影事件；

利用电影摄影逻辑实时地解释所接收的电影摄影事件，电影摄影逻辑包括用于解释所接收的电影摄影事件的一个或多个有限态自动机；和从解释的电影摄影事件自动地产生一摄影机画面规范，其中摄影机画面规范被用于观看一虚拟场景。

2. 如权利要求 1 所述方法，其中摄影机画面规范包括从一组用于镜头的多个规范中选择的至少一个规范，所述多个规范包括：

第一规范：第一演员处于镜头中央的一侧，而第二演员处于镜头中央的另一侧；

第二规范：使用特写摄影机距离，并且第一演员处于镜头中央的一侧，而第二演员处于镜头中央的另一侧；

第三规范：将三个演员在镜头中作等边三角形安排；

第四规范：使得第一演员在镜头中处于第二演员的肩头之上；

第五规范：使用远摄画面，使得第一演员在镜头中处于第二演员的肩头之上；

第六规范：使得第一演员和第二演员在镜头中处于第三演员的肩头之上；

第七规范：使得第一演员和第二演员在镜头中处于第三演员的肩头之上，但只有第一演员和第二演员出现在镜头中；

第八规范：使得第一演员在镜头中处于第二演员的肩头之上，但只有第一演员出现在镜头中；

第九规范：使得只有第一演员的整个身体出现在镜头中；

第十规范：镜头与第一演员一同移动，使得当第一演员逼近第二演员时，镜头跟随第一演员；

第十一规范：当第一演员逼近第二演员时，镜头在原地绕枢轴转动以跟随第一演员；

第十二规范：当第一演员逼近第二演员时，镜头在原地绕枢轴转动以跟随第一演员，然后，镜头与第一演员一同移动，使得当第一演员逼近第二演员时，镜头跟随第一演员；

第十三规范：使镜头保持在固定位置、方向和视野；

第十四规范：比以前的摄影机的位置更靠后和更高；

第十五规范：在镜头中显示谈话中的多个演员的每一个；

第十六规范：使镜头中画面从演员头部一位置开始，镜头方向基于从演员头部至一目标的矢量；

第十七规范：使镜头中画面从演员头部一位置开始，镜头中摄影机的焦距随时间的继续而变长；

第十八规范：使镜头中画面从演员前方一位置开始，镜头方向跟随演员前方的方向；以及

第十九规范：保持相对于先前摄像机位置不变。

3. 如权利要求 1 所述方法，其中摄影机画面规范包括选择摄影机画面距离。

4. 如权利要求 3 所述方法，其中摄影机画面距离包括至少一个从一组含一个极端特写画面距离，一个特写画面距离，一个中等画面距离，一个全画面距离和一个远画面距离中选择的距离。

5. 如权利要求 1 所述方法，其中电影摄影逻辑包括用于选择影片镜头的影片试探法。

6. 如权利要求 5 所述方法，其中影片试探法包括至少一个从一组含不要越线，避免跳跃剪辑，使用固定拍摄，让演员引导，间断运动和保持连续运动中选择的试探法。

7. 如权利要求 1 所述方法，其中电影摄影逻辑包括摄影机模块和电影摄影表达方式。

8. 如权利要求 1 所述方法，还包括：调节一个电影摄影事件内的一个电影摄影角色以使它与摄影机画面规范一致。

9. 如权利要求 1 所述方法，还包括：生成虚拟电影摄影的摄影机模块，该生成步骤包括：

选择一个虚拟对象观看；

选择一个虚拟摄影机画面规范观看对象；和

将所选择的虚拟摄影机规范链接于虚拟对象。

10. 如权利要求 1 所述方法，还包括：

自动选择一个虚拟摄影机画面距离；和

实时地修改一个虚拟演员的排列和动作以符合于摄影机画面规范和摄影机画面距离。

11. 如权利要求 7 所述方法，还包括：生成一个或多个独立虚拟电影摄影表达方式，该生成步骤包括：

定义一个或多个独立虚拟电影摄影表达方式状态；

选择一个摄影机模块以关联每个定义的独立虚拟电影摄影表达方式状态；和

定义包括定义的独立虚拟电影摄影表达方式状态之间的过渡的表达方式动作，

其中表达方式动作被用于实时地显示一虚拟场景。

12. 如权利要求 11 所述方法，其中表达方式动作包括例外的寄存和引发。

13. 如权利要求 11 所述方法，还包括：

定义一个或多个复杂虚拟电影摄影表达方式状态；

为一个或多个定义的复杂虚拟电影摄影表达方式状态选择一个独

立虚拟电影摄影表达方式；

定义包括定义的复杂虚拟电影摄影表达方式状态之间的过渡的复杂表达方式动作；和

定义被测试以确定发生了哪一种过渡的条件，

其中复杂虚拟电影摄影表达方式被用于在计算机显示器上实时地显示一虚拟场景。

14. 如权利要求 11 所述方法，还包括：

选择独立虚拟电影摄影表达方式的第一组；

组合独立虚拟电影摄影表达方式的第一组以生成一复杂虚拟电影摄影表达方式，其中复杂虚拟电影摄影表达方式包括复杂虚拟电影摄影表达方式状态，摄影机模块和复杂虚拟电影摄影表达方式动作；

选择独立虚拟电影摄影表达方式和复杂虚拟电影摄影表达方式的第二组；和

组合独立虚拟电影摄影表达方式和复杂虚拟电影摄影表达方式的第二组以生成一个高级虚拟电影摄影表达方式，

其中高级虚拟电影摄影表达方式被用于在计算机显示器上实时地显示一虚拟场景。

15. 如权利要求 14 所述方法，其中高级虚拟电影摄影表达方式是一个动词虚拟电影摄影表达方式。

16. 如权利要求 7 所述方法，还包括：

选择一组动词虚拟电影摄影表达方式；

组合选择的动词虚拟电影摄影表达方式以生成多个虚拟电影场景；和

组合选择的各个虚拟电影场景以生成一部虚拟电影。

17. 如权利要求 7 所述方法，还包括：

选择一组虚拟电影摄影表达方式，其中虚拟电影摄影表达方式包括

一个或多个虚拟电影摄影表达方式状态，一个或多个摄影机模块和独立表达方式动作；

根据特定的影片导演风格在虚拟电影摄影表达方式内改变摄影机模块以生成一组定制的、修改的虚拟电影摄影表达方式；

根据特定的影片导演风格组合修改的虚拟电影摄影表达方式组以生成多个虚拟电影场景；和

组合选择的各个虚拟电影场景以生成一部虚拟电影。

18. 一种用于虚拟电影摄影的系统，该系统包括：

一个实时应用模块，用于在客户计算机上实时响应接收的用户输入而产生电影摄影事件；

一个虚拟电影摄影模块，用于实时解释电影摄影事件，并且基于电影摄影事件自动地确定摄影机位置，虚拟电影摄影模块包括用于解释电影摄影事件的一个或多个有限态自动机；和

一个重现器应用模块，用于响应虚拟电影摄影模块的命令而重现一虚拟环境。

19. 如权利要求 18 所述系统，其中虚拟电影摄影模块包括摄影机模块和虚拟电影摄影表达方式。

20. 如权利要求 18 所述系统，还包括：多个连接到网络的客户计算机，其中第一所述的客户计算机连接到网络并且至少两个客户计算机中的每一个客户计算机具有虚拟环境下的虚拟演员，每个由其相关的客户计算机分开控制。

虚拟电影摄影方法和系统

本发明涉及计算机动画制作。更具体地讲，本发明涉及产生在虚拟三维计算机动画环境中显示事件的规范。

国际互联网这类的计算机网络的迅速扩大，为计算机使用者提供了从远方地点获得文本、图形以及音频和视频片段形式的信息的便捷途径。计算机使用者可以利用一个连接到国际互联网或其它计算机网络(例如，内部网(intranet))的计算机与其它连接到计算机网络的计算机使用者实时地交互联系。例如，一个计算机使用者可以与一个远端计算机使用者一起参与动画计算机视频游戏。这些动画计算机视频游戏中的某一些使用一种计算机使用者在其中控制虚拟动画角色的计算机产生的虚拟三维(3D)环境。

虚拟 3D 闲谈环境也在不断地普及，在这种环境中，计算机使用者通过由计算机使用者控制并代表计算机使用者的 3D 虚拟动画演员(有时称为化身)相互交互联系。在这类闲谈环境中，例如，为每个计算机使用者提供了一个房间的 3D 显示，在这个房间中根据那些正在互相通信的使用者提供虚拟演员或化身。虚拟演员的排列和定位为计算机使用者提供了指示哪些计算机使用者正在相互通信的 3D 显示。在使用文本作为通信界面的常规闲谈环境中这种类型的图形指示是不可能的。

这种在一虚拟 3D 环境中通信的形式，尽管有着巨大的发展前景，但也存在着许多相关的问题。例如，使用者通常难于领悟和驾驭虚拟 3D 环境，在模拟环境中确定他们希望与之通信的计算机使用者的虚拟演员，和以适当方式安排他们的演员使得所有在一起谈话的使用者能够看到彼此的演员。

这类问题与一个世纪前影片制作开始以来电影摄影师所面临的问题是一样的。在这些年中，电影制作人发展了一些使动作能够全面和有效地传递的拍摄惯例或规则。尽管几乎没有对这些拍摄规则的清楚的说明，但它们被如此普遍地使用，以至为观众所接受和很好地理解。这些电影摄影的拍摄惯例或规则利用摄影机位置，场景结构和“镜头间(inter-shot)”一致性规则来传达电影摄影信息。例如，观众可以很好地理解这样一组场景，从一个高视点的画面开始，过渡到以一条公路和汽车为主的低视点画面，然后是汽车中人的特写镜头。

在计算机图形界中较少注意用来控制摄影机位置和场景结构的影片的电影摄影惯例或规则。许多计算机动画制作系统企图在有限的应用中将一些电影摄影原理运用到计算机图形，有限的应用包括：利用教学放映的脱机规划解释复杂任务的动画制作规划系统；响应自然语言命令的半自主演员的生成；和组编来自视频资料库的视频剪辑片段的短序列。但是，这些系统一般很少或根本不注意摄影机方位或镜头间一致性规则(例如，一个演员从一帧画面的左面离开场景，又从右面进入必然显出不一致性)。

已经说明过一些用于在执行交互任务时发现最佳摄影机方位的交互式动画制作系统。但是这些系统既不试图产生场景序列，它们也不在开发它们的规范中应用电影摄影规则。

计算机执行的3D虚拟应用的自动化电影摄影原理存在着常规影片制作中不存在的困难问题。虽然在各种讲义中提到各种电影摄影规则的非正式说明，一般对它们定义的确切程度并不足以以计算机能够执行的正式语言表达。此外，在计算机上实时执行自动图像或“摄影机”控制遇到了比人类导演所面对的更难克服的限制。人类导演一般是根据一个事先通过的脚本工作的，并且可以在以后对原始胶片进行脱机编辑。这对实时执行的交互式3D计算机应用是不可能的。

根据本发明，克服了自动化电影摄影原理的问题。本发明包括一种用于根据自动化电影摄影原理在虚拟 3D 环境中捕获或提供事件的虚拟电影摄影方法。该方法包括接收在一特定时间周期中(例如，一个计算机时钟分段)发生的事件说明。事件一般是诸如(主体，动词，对象(subject,verb,object))这类的选择形式。例如，一个(B，谈话(talk)，A)事件表示虚拟演员 B 正在对虚拟演员 A 讲话。解释接收的事件以产生一个用于观看虚拟演员的适当的摄影机规范。

该方法使用两个主要组成部分：摄影机模块和电影摄影表达方式。摄影机模块负责一个场景中特定摄影机的低级几何方位，和把虚拟演员位置上的精细变化制成每个摄影机镜头的最佳帧。电影摄影表达方式说明用于把摄影机镜头组合成序列(例如，动画序列)的电影摄影逻辑。将摄影机模块和电影摄影表达方式一起使用以产生虚拟影片和动画。

利用这种方法来实现一基于自动虚拟电影摄影的有限态自动机的实时摄影机控制器，称为虚拟电影摄影应用模块(VC)。在虚拟现实和其它交互式应用中使用这种 VC，以改善固定视点镜头或这类应用经常使用的安装在顶棚上的摄影机。VC 还通过允许用户利用一处于看来自然的摄影机位置的代理看到他们自己来帮助改善“智能代理”用户界面。

通过下面的详细说明可以对本发明的上述和其它特征和优点更清楚的了解，说明是通过参考附图进行的。

图 1 是一种用于实现本发明的示范实施例的计算机系统的方框图；

图 2 是说明本发明的一个实施例的方法的流程图；

图 3 是本发明的一个系统的方框图；

图 4 是说明“线”概念的方框图；

图 5 是说明电影摄影的摄影机位置的方框图；

图 6 是说明近景和全景摄影镜头的方框图；

图 7A 和 7B 是说明在一个摄影机镜头中三个演员的安排的方框图；

图 8 是说明用于三个谈话中的演员的摄影机镜头的方框图；

图 9 是说明顶点(apex)摄影机模块的方框图；

图 10 是说明特写顶点(closeapex)摄影机模块的方框图；

图 11 是说明顶点 3(apex3)摄影机模块的方框图；

图 12 是说明 external 摄影机模块的方框图；

图 13 是说明外侧 1 到 2(external 1to2)摄影机模块的方框图；

图 14 是说明包含(exclose2)摄影机模块的方框图；

图 15 是说明内侧(internal)摄影机模块的方框图；

图 16 是说明完整(full)摄影机模块的方框图；

图 17 是说明跟踪拍摄(tracking)摄影机模块的方框图；

图 18 是说明摇拍(panning)摄影机模块的方框图；

图 19 是说明追拍(follow)摄影机模块的方框图；

图 20 是说明固定(fixed)摄影机模块的方框图；

图 21 是说明升高(raised)摄影机模块的方框图；

图 22 是说明组(group)摄影机模块的方框图；

图 23 是说明主观(subjective)摄影机模块的方框图；

图 24 是说明“线”位置变化的方框图；

图 25 是说明用于一表达方式的有限态自动机方框图；

图 26 是说明一个具有一种包含另一个有限态自动机状态的有限态自动机的方框图；

图 27 是说明一个用于两个演员之间谈话的表达方式的方框图；

图 28 是说明一个用于三个演员之间谈话的表达方式的方框图；

图 29 是说明生成一个独立表达方式的方法的流程图；

图 30 是说明生成一个复杂表达方式的方法的流程图；

图 31 是说明生成一个高级表达方式的方法的流程图；

图 32 是说明一个等级表达方式结构的方框图；

图 33 是说明从表达方式生成虚拟电影的方法的方框图；

图 34 是说明一种表达方式座标显示系统的方框图；

图 35 是说明一种用于虚拟电影摄影应用的用户界面的方框图。

参考图 1，一种用于本发明示范实施例的操作环境是一个带有包括至少一个与一存储系统 16 结合的高速处理单元(CPU)14 的计算机 12 的计算机系统 10，一个输入设备 18，和一个输出设备 20。这些元件由一总线结构 22 相互连接。

图示的 CPU 14 是熟知的设计，并包括一个用于执行计算的 ALU 24，一批用于数据和指令的临时存储的寄存器 26，和一个用于控制系统 10 操作的控制单元 28。包括来自 Digital Equipment, Sun, MIPS, IBM, Motorola, NEC, Intel, Cyrix, AMD, Nexgen 及其它公司的各种处理器中的任何一种都同样可以用作 CPU 14。尽管示出了一个 CPU 14，作为选择，计算机系统 10 可以包括多个处理单元。

存储系统 16 包括主存储器 30 和二级存储器 32。示出的主存储器 30 是高速随机存取存储器(RAM)，和只读存储器(ROM)。主存储器 30 可以包括任何附加或替代高速存储器件或存储电路。二级存储器 32 采取长期存储的形式，例如 ROM，光盘或磁盘，有机存储器，或任何其它易失或非易失性海量存储系统。熟悉本领域的技术人员会知道，存储器 16 可以包括各种替代组件和/或替代组件的组合。

输入和输出设备 18,20 也是众所熟知的。输入设备 18 可以包括一个键盘，鼠标，指向设备，声频设备(例如，麦克风等)，或任何其它向计算机系统 10 提供输入的设备。输出设备 20 可以包括显示器，打印机，声频设备(例如，扬声器等)，或其它向计算机系统 10 提供输出的设备。输入/输出设备 18，20 也可以包括网络连接，调制解调器，或其它用于与别的计算机系统或设备进行通信的设备。

如同本领域的技术人员所熟知的，计算机系统 10 还包括一个操作系统和至少一个应用程序。操作系统是控制计算机系统的操作和资源分

配的一组软件。应用程序是利用通过操作系统分配的可用计算机资源执行用户所希望的任务的一组软件。两者都存在于所示的存储系统 16 中。

根据熟悉计算机编程的技术人员的实践，除非另有指示，以下参考计算机系统 10 所执行动作和操作的符号代表说明本发明。这种动作和操作有时被看作是计算机执行的。应当理解动作和符号代表的操作包括由 CPU14 操纵代表数据位的电信号，该操纵产生所得到的变换或简化电信号代表和在存储系统 16 中的存储单元维持数据位，由此而重新配置或改变计算机系统的操作，以及其它信号处理。其中保持数据位的存储单元是具有对应于数据位的特殊电、磁、光学或有机性质的物理单元。

也可以把数据位保存在包括磁盘和任何计算机 12 可读的其它易失性或非易失性海量存储系统的计算机可读介质上。计算机可读介质包括仅存在于计算机系统 10 中或分布在可能是本地的或远端的多个互联计算机系统 10 之间协同或互联的计算机可读介质。

如图 2 中所示，本发明的一个示范实施例包括一种用于在虚拟 3D 环境中提供计算机产生场景的虚拟电影摄影方法 34。这样一种虚拟 3D 环境代表一个 3D 图像空间，在这个空间中计算机产生虚拟对象或角色的显示图像并在显示屏上重现出来。电影摄影是“虚拟的”，在于图像不是用摄影机形成的，而是根据计算机产生的观看方向，放大率等形成的。

方法 34 包括接收在一特定时间周期(例如，一个计算机时钟分段)中发生的事件的说明，步骤 36。事件一般是由(主体，动词，对象)这样的选择形式表示的。例如，一个(B, talk, A)事件表示虚拟演员 B 将要 对虚拟演员 A 讲话。解释事件，步骤 38，以产生适当的虚拟摄影机规范，步骤 40。还产生表演提示，以把虚拟演员转移到一个镜头的更好帧中。

图 3 示出了用于本发明的一个示范实施例的系统 44。系统 44 包括一个实时应用模块 46(例如, 一个多用户闲谈应用, 一个计算机视频游戏, 一个视频会议应用, 一个虚拟现实应用, 等等), 该实时应用模块 46 向一虚拟电影摄影应用模块 50 提供事件的说明和包括静态几何信息, 材料性质(例如, 纹理和明暗度)及光源方位等在内的其它信息 48。虚拟电影摄影应用模块 50 执行虚拟电影摄影方法 34, 并把虚拟摄影机信息和虚拟演员控制信息传送 52 到重现器模块 54。虚拟电影摄影应用模块 50 也可以对来自实时应用模块 46 的信息(例如, 虚拟演员的位置)进行询问 56。重现器模块 54 包括图形和动画制作软件, 例如微软公司(Microsoft Corporation of Redmond, Washington)提供的 Direct3D。但是, 也可以使用其它重现器软件。重现器模块 54 从实时应用模块 46 接收动画制作参数 58(例如, 静态几何信息, 虚拟演员模型, 光照方位, 等等)。

在指定的时间间隔(例如, 一个时钟分段), 实时应用模块 46 还负责接收来自用户的输入, 适当地移动虚拟演员, 和把所得的动画制作信息(例如, 虚拟演员的位置) 60 送回到重现器模块 54。此外, 在指定的时间间隔, 虚拟电影摄影应用模块 50 对虚拟摄影机的规范作出决定, 虚拟环境就是从这个规范重现的。如果需要, 虚拟电影摄影应用模块 50 还对虚拟演员的排列和动作进行小的修改。

影片的电影摄影

一个活动画面或影片一般是多个场景的序列, 每个序列捕获一个总体上连续的情景或动作。而每个场景又是由一个或多个摄影机镜头构成的。一个摄影机镜头是一个由摄影机连续操作形成的活动画面片段。一般一部影片包括大量的独立的摄影机镜头, 每个镜头一般延续一两秒至数十秒。

图 4 示出了在大多数电影中使用的影片的电影摄影惯例或规则。具

体地讲，影片导演相对于一条“线” 62 规定摄影机方位，线 62 是一个连接两个相互交流的演员 X64 和 Y66 的假想的矢量。线 62 也指向一个演员运动的方向 68，或定向于该演员面对的方向。相对于线 62 规定摄影机方位 70。

如图 5 中所示，相对于线 62，从摄影机方位 A 72 拍摄演员 Y66 叫做“外侧倒摄方位(external reverse placement)”；演员 Y66 右侧的三分之二在焦点上，而左面的三分之一显示了演员 X64 头部的后侧。从摄影机方位 B74 拍摄演员 X64 称为“平行摄影机方位(parallel camera placement)”，这个方位仅显示演员 X64。从位置 C76 拍摄演员 X64，产生“内侧倒摄方位(internal reverse placement)”。一般在摄影机方位 C76，演员 X64 仅占据镜头左侧的左三分之二。从位置 D78 拍摄导致一种演员 X64 和 Y66 都显示出来的“顶点方位(apex placement)”拍摄。从位置 E80 拍摄，产生一个对演员 Y66 的“内侧倒摄方位”。从位置 F82 拍摄产生对演员 Y66 的“平行摄影方位”。最后，从 G84 拍摄称为“外侧倒摄方位”；镜头左侧的三分之二显示焦点上的演员 X64，而右侧三分之一显示演员 Y66 头部的后侧。但是，一个电影导演也可以使用更多或更少的摄影位置。

电影摄影师已经知道某些“截取高度(cutting height)”可以造成令人喜爱的画面构成，而其它的则产生令人不快的结果(例如，在演员踝部截取的图像是不好的)。一般在这个领域中使用的有五种有用的摄影机距离：(1)一种极端的特写图，在颈部截取；(2)一个特写图，在胸部以下或腰部截取；(3)中距离图，在两腿交叉处或膝部以下截取；(4)一个全视图，显示整个演员；和(5)远视图，提供演员的远距离的配景画面。

利用截取高度，对于一个特定的特写镜头看起来十分自然的演员位置，当从更靠后的地方观看时，可能会显得太靠近了。各个独立的镜头也需要演员方位的细微差别，以使他们在屏幕上显得自然。例如，如图 6 中所示，演员 X88 和另一个演员 Y90 相对于线 62 的特写图 86 看起来

十分自然。但是，全视图 92 显得演员 X88 与 Y90 比想象的靠得更近，因此，全视图 92 是不自然的。

同样，包含多个演员的镜头经常需要移动演员的位置，以便适当地安排他们。如图 7A 中所示，相对于线 62，第一个演员 X94 和第二个演员 Y96 部分地掩蔽或遮挡了第三演员 Z98。移动第三演员 Z98 的位置会恰当地安排镜头以使第三演员 Z98 包括进来。如图 7B 中所示，第一演员 X94 和第二演员 Y96 保留在原来的位置，而第三演员 Z98 的位置已经移动。但是，也可以移动第一演员 X94 和第二演员 Y96 的位置，而使第三演员 Z98 留在她原来的位置上，以达到同样的效果。

影片的电影摄影试探法

电影制作人已经发表了许多用于选择好镜头的试探法(即，规则)，并且已经非正式地规定了置于连续镜头上以产生好的场景的限制。比较重要的试探法说包括：“不要越线(Don't cross the Line)”；“避免跳跃剪辑(Avoid jump cuts)”；“使用固定拍摄(Use establishing shots)”；“让演员引导(Let the actor lead)”；“间断运动(Bread movement)”；和“保持连续运动(Maintain continuous movement)”。但是，可以使用更多或较少的试探法。

“不要越线”规则保证了运动中演员的连续镜头保持视在运动方向。如图 5 中所示，一旦从线 62 的一侧拍摄一个初始镜头，也必须从这一侧拍摄后续的镜头，除非使用一中性的固定拍摄来表示向另一侧的过渡。演员 Y66 向图 5 的右侧移动，对于在线 62 另一侧的摄影机来说，表现为向左移动。

一个场景内的摄影机镜头的改变(即，一个切换)应当是在两个镜头之间演员的大小、视图或数目中的显著差异，以“避免跳跃剪辑”。不能满足这些条件的剪辑称为跳跃剪辑，并且一般会产生跳动、无条理的效果。

“使用固定拍摄”规则在移动到近镜头之前建立背景场景。如果在场景中存在新的发展，重新建立变更的背景场景。

“让演员引导”允许演员开始全部运动，摄影机跟随。摄影机停留在演员稍微前面一点的地方。

至于“间断运动”规则，是把显示运动的一个场景分成至少两个镜头。一般每个镜头被剪辑成演员看起来像是通过半个屏幕区移动。在切换中完成摄影机-主体间距离的改变。

“保持连续运动”规则说的是剪辑之间的运动对象的视在速度应当是恒定的。并且不要颠倒运动的方向。

摄影机镜头序列

电影摄影师使用把特定的动作捕获为镜头序列的程式。例如，在三个演员之间的对话中，拍片人可以以一个三个人的固定拍摄开始，然后转到两个对话演员的外侧倒摄镜头，偶尔点缀几个第三个演员的反应镜头。当电影制作人对一种程式的喜爱超过另一种程式时，除讨论的情况外，例如像 D. Arijon 的《电影语言基础》(Grammar of the Film Language, Hasting House, New York, 1976)之类的书中提供了非正式的程式汇编。

图 8 示出了一种表现三个演员之间谈话的常用程式。这个从 D.Arijon 的书中的图 6.29 改编的程式提供了一种用于表现三个演员之间谈话的方法。第一个镜头是一个越过演员 A 102(圆头)的肩部向演员 B 104(八边形头)和演员 C 106(三角形头)的外侧倒摄镜头 100。方框 108 中示出了外侧倒摄镜头 100 的画面(演员 B 104'和 C 106'以及演员 A 102'的头的后部)。

第二镜头 110 和第三镜头 112 是演员 B 104 和 C 106 的外侧倒摄镜

头。方框 116 和 118 分别示出了这些镜头的画面。外侧倒摄镜头 110 显示出，在方框 116 中左侧三分之二是演员 C 106'''(三角头)的头的后部和右侧三分之一是演员 B 104'''(六边形头)。外侧倒摄镜头 112 在方框 118 中示出了右侧三分之二是演员 B 104'''的头，左侧三分之一是演员 C 106'''。第四镜头 120 是演员 A 102''''的内侧倒摄反应镜头。方框 122 中示出了镜头 120 的画面。由于在演员 A 102''''的“背后”没有其它演员，所以仅显示了演员 A 102''''的头的正面。由于这是一个内侧倒摄反应镜头，所以没有显示头的后部。

在这本书中，Arijon 规定，在演员 B 104 和 C 106 对演员 A 102 说话时，利用这种方案的典型顺序的编辑次序应当是第一镜头 100(方框 108)与第四镜头 120(方框 122)之间的交替出现。当演员 B 104 和 C 106 开始互相说话时，顺序转移到第二镜头 110(方框 116)与第三镜头 112(方框 118)之间的交替出现，偶尔地插入反应镜头 120(方框 122)。周期性地引入镜头 100(例如，每隔 15 秒)以重新建立整个组。

尽管有各种用来把镜头按顺序组合在一起的程式，但每个电影导演倾向于依赖少量的程式。任何一个独立的导演所使用的特殊程式，为该导演的影片带来特定的品味或风格。在本发明的一个示范实施例中，导演的风格是由在虚拟电影摄影应用模块 50(此后称为 VC)中编码的特定程式确定的。

体系结构

上述电影摄影专业知识在 VC 50 中的计算机软件中编码为动作和由计算机执行的操作的符号代表。VC 50 包含两个部分：摄影机模块和表达方式。摄影机模块提供了利用上述的和图 5 中所示的不同摄影机方位形成计算机生成图像的虚拟摄影机。摄影机模块也可以稍微修改虚拟演员的位置以构成更好的镜头和影响表演的虚拟演员的比例。表达方式说明了用于把来自摄影机模块的镜头如同前面对影片中说明过的那样组合成序列的逻辑。

摄影机模块

每个摄影机模块把被称为主要演员的多个演员作为输入；确切的演员数量取决于特定的摄影机模块。每个摄影机模块把虚拟摄影机放在距演员的一个特定距离，以适合上面讨论过的五种截取高度。但是，也可以使用更多或更少的截取高度。虚拟摄影机位置也使演员处于镜头中的一个特定位置上。此外，摄影机模块可以给演员稍微地再定位，以改善镜头。最后，摄影机方位是自动选择的，以遵守上述的不越过线 62 的规则。

在本发明的一个示范实施例中，如下面将指出的，提供了 19 种不同摄影机模块。但是，也可以使用更多或更少的摄影机模块。

如图 9 中所示，摄影机模块 `apex(actor1,actor2)`124 取两个演员 A 126 和 B 128(对应于 `actor1` 和 `actor2`)作为输入，并且设置虚拟摄影机 130 的位置使得相对于线 62 第一演员 A 126 在镜头中央的一侧，而第二演员 B 128 在中央的另一侧。摄影机距离是用圆圈示意性表示的两个演员 A 126 与 B 128 之间距离的函数。

如图 10 中所示，摄影机模块 `closeapex(actor1,actor2)`132 提供了 `apex` 摄影机方位。但是，`closeapex` 摄影机模块与 `apex()`摄影机模块的不同在于后者总是使用虚拟摄影机 134 相对于线 62 的特写摄影机距离。为了构成更好的镜头，这个摄影机模块 132 可以移动使演员 A 136 和 B 138 更靠拢一些。

如图 11 中所示，摄影机模块 `apex3(actor1,actor2,actor3)`140 将演员 A 142，B 144 和 C 146 作等边三角形安排。摄影机模块 `apex3()`把虚拟摄影机 148 放在与 `apex()`124 和 `closeapex()`132 相同的相对位置上，但是它将总是选择对当前镜头有益的一侧的线 62。

如图 12 中所示，摄影机模块 `150external(actor1,actor2)`取两个演员 A 152 和 B 154 作为输入，并且安排虚拟摄影机 156 的位置使得越过第二演员 B 154 的肩头看第一演员 A 152，第一演员 A 152 占据镜头的三分之二，第二演员 B 154 占据镜头的其余三分之一。与此相同，摄影机模块 `external(actor2,actor1)`设置虚拟摄影机 156'的位置使得越过第一演员 A 152 的肩头看第二演员 B 154，第二演员 B 154 占据镜头的三分之二，第一演员 A 152 占据镜头的其余三分之一。图 8 的方框 116,118 中示出了这些 `external()`摄影机模块画面。

`2shot(actor1,actor2)`摄影机模块(未示出)与 `external` 类似，但是使用了两个演员的远摄画面。

如图 13 中所示，摄影机模块 `external1to2(actor1,actor2,actor3)`158 提供了一个演员 A 160 与两个其他演员 B 162 和 C 164 之间的一种外摄影机方位。`External1to2()`摄影机模块设置虚拟摄影机 166 的位置，使得越过演员 A 160 的肩头看两个演员 B 162 和 C 164，演员 B 162 和 C 164 占据镜头的三分之二，而演员 A 160 占据镜头的其余三分之一。图 8 的方框 108 中示出了这个画面。这个摄影机模块 166 也可以修改演员的位置以构成更好的镜头。

如图 14 中所示，摄影机模块 `exclose2(actor1,actor2,actor3)`168 调节处于与摄影机模块 `external1to2()`158 中摄影机 166 相同位置的虚拟摄影机 170 的焦距和方向，使得三个演员中只有两个出现在镜头中。例如，演员 B 174 和 C 176 出现在镜头中，而演员 A 172 不在镜头中出现。

如图 15 中所示，摄影机模块 `internal(actor1,[actor2])`可以包括一个或两个演员。如果规定两个演员 A 180 和 B 182，那么摄影机 184 沿 `external()`摄影机模块 150 相同的视线放置，但更向内靠并且具有更窄的视野，使得只能单独看到第一演员 A 180，占据着三分之二的屏幕。如果指定一个演员 A 180，那么线 62 采取演员 A 180 面对的方向，摄影

机方位 184' 相对于这条线以相同的方式选择。

如图 16 中所示, `full(actor,theta)`摄影机模块 168 将一个演员 188 放在镜头的中央。适当地调节摄影机到演员 188 的距离, 使得演员 188 的整个身体显示出来。摄影机 190 的方向与演员 188 方向之间的角度是 `theta 192`。

在一个演员运动时, 使用以下三个有关摄影机模块 `tracking()`, `panning()`和 `follow()`。它们与前面刚刚说明过的摄影机模块的区别在于, 它们定义了一个运动的摄影机, 该运动摄影机动态地改变位置和/或方向, 以相对于一个第二演员保持所希望演员的位置靠近镜头中央。在上述的摄影机模块中摄影机位置是固定的。摄影机模块 `tracking()`, `panning()`和 `follow()`使用了最小和最大距离限制, 以确定把摄影机放在离线 62 多远的地方。

如图 17 中所示, `tracking(actor1,actor2,mindist,maxdist)`摄影机模块 194 沿垂直于线 62 的线 198 设置摄影机 196。然后, 摄影 196' 与演员 A 200' 一同移动, 随着演员 A 逼近演员 B 202 保持与演员 A 200' 同样的方向 198'。

如图 18 中所示, `panning(actor1,actor2,mindist,maxdist)`摄影机模块 204 将摄影机 206 设置在线 62 以外演员 A 208 的前方, 然后在演员 A 逼近演员 B 212 时在位置 210 绕枢轴转动以跟随演员 A 208' 的移动。

如图 19 中所示, `follow(actor1,actor2,mindist,maxdist)`摄影机模块 214 组合 `tracking()`194 和 `panning()`204 摄影机模块的操作。首先它像一个摇镜头摄影机 216 一样操作, 但当演员 A 218 通过一个预定点 220 时, 摄影机 222 开始从背后“跟随”(即, 跟踪)演员 A 218 到 222', 而不是在演员 A 逼近演员 B 224 时让演员 A 218' 离开进入到这段距离中。

其余的摄影机模块执行杂项功能。如图 20 中所示，`fixed(cameraspec)`摄影机模块 226 用来为摄影机 230 规定一个特殊的固定位置，方向和视野 228。用它来提供一个场景 232 的全景镜头。

如图 21 中所示，`raised()`摄影机模块 234 提供了一个与以前摄影机位置有关的方位。新摄影机位置 236 比过去的摄影机在 L1,H1 的位置 238 更靠后 L2 和更高 H2，但是朝向场景 244 的方向与以前的摄影机 238 相同。

如图 22 中所示，`group(conversation)`摄影机模块 246 覆盖了谈话中的所有演员 248，250，252。这个来自摄影机 254 的 `group()`镜头是由实时应用模块 46 发出的图形信息 256(例如，有边界框)确定的。

如图 23 中所示，`subjective(actor,target)`摄影机模块 258 使摄影机 260 与演员的头部 262 上的位置同步。从演员的头 262 开始，确定了一个基于从演员头 262 至一目标 266(例如，场景中另一个演员或一个对象)的矢量的方向。

以下三个摄影机模块由于容易确定它们中每一个的功能，所以没有用独立的附图示出。`Subjectzoom(actor)`摄影机模块与 `subjective()`摄影机模块 258 类似，只是摄影 258 的焦距随时间的继续和演员绕场景运动而变得更长。

`Pov(actor)`摄影机模块模仿一个演员的前方。这个摄影机模块跟随演员前方的位置和方向。`Null()`摄影机模块使摄影机停留在它先前位置。这个摄影机模块使得一个摄影机的方向保持保持不变。

相对于线选择虚拟摄影机方位

如上所述和如图 14 中所示，线 62 是相对于镜头中两个演员 64,66 定义的。因此，线 62 本身将不可避免地从一个镜头到另一个镜头地改

变。电影摄影的规则要求当线 62 保持恒定时，摄影机 70 应当保留在线 62 的同一侧。相对于线 62,一个给定规范(例如， actor1,actor2)所用的摄影机模块可以说明相应于线 62 两侧的对称位置的两个距离中的一个。如果线 62 从一个镜头至下一个镜头没有改变，那么特定距离的选择是简单的：选择在线 62 同一侧的摄影机位置 70。

当线 62 改变时，例如，当镜头中两个演员中的一个改变位置时，选择就不容易确定了。参考图 24，和根据本发明的一个示范实施例，适当地选择特定距离，使得相对于新线 270 选定的摄影机方向 268 最接近以前的摄影机 70 的方向。上述的摄影机模块具有一个以上的摄影机方向的选择，并且选择最接近以前的摄影机方向的新线。但是，在当前线改变时也可以使用其它摄影机方向。

用摄影机模块影响动作

如早先指出的，摄影机模块能够通过影响镜头中演员的位置和表演尺度精细地改善镜头。由于实时应用 46 主要负责控制演员，VC 50 作出的改变必需足够精细而不破坏镜头间的连续性。

例如，如果象图 6 的方框 86 中所示，两个主要演员之间的距离大于某个最小距离，那么 `closeapex()`摄影机模块 132 移动他们更靠近一些。如图 7B 中所示，`apex3()`摄影机模块 140 调节三个主要演员的位置，使得没有演员被镜头中任何其他演员挡住。一些摄影机模块可以把演员从镜头中一同清除，以避免一个演员仅仅是半路出现在屏幕上，或在屏幕上挡住了另一个主要演员的情况。例如，`internal()`摄影机模块 178 如图 15 中所示将第二演员 B 182 从镜头中清除了，十分象拍摄真的内侧倒摄镜头的方式。

摄影机模块也影响一个演员的表演尺度。例如，在点头时演员头部移动的程度，或走路时演员腿部摆动的程度。通过设置一个与每个演员相关的参数，对镜头中的所有演员(不只是主要演员)进行改变，重现器

54 可以使用这个参数来调节演员的运动尺度。可以用一个简单的函数 $L(\text{scalequation})$ 设置这个参数。当屏幕上演员的尺寸减小时函数单调增加，并表示为：

$$L = \arctan(\text{distance} - D_n) \cdot k + 1.0 ,$$

其中 D_n 是到演员具有正常的表演水平的距离(例如, 1.0), k 把 L 调节到 0.5 至 1.5 的范围内。

利用摄影机模块检测遮挡

除了定位虚拟摄影机之外, 摄影机模块还有一个额外的责任。它们也负责检测在什么时候一个或多个主要演员在场景中被遮挡(例如, 图 7A 中的演员 Z98)。在发生遮挡的时候, 摄影机模块增加一个遮挡计数器的计数, 或如果被遮挡演员解除了遮挡时, 把计数器重新设置为零。这个计数器是表达方式用来决定是否改变到一个不同镜头。例如, 如果检测到遮挡, 一个一般要将一个镜头保持一个预定的时间间隔(例如, 20 个时间分段)的表达式可能决定更快地改变镜头。

使用摄影机模块

参考图 3, 在一些预定的时间间隔(即, 一个动画序列的每一帧)实时应用 46 向 VC 50 发送发生在该时间间隔中并且对希望的演员是重要的事件的说明。事件是(主体, 动词, 对象)形式的。例如, 一个(B, 谈话, A)事件表示演员 B 正在向演员 A 讲话, 和一个(A, 移动, 酒吧)事件表示演员 A 准备向一个酒吧移动。主体是一个演员, 而对象可以是一个演员, 一个当前谈话, 一个固定对象(例如, 酒吧), 或不存在。在本示范实施例中, 动词可以是如下面将说明的集合(空闲(idle), 谈话(talk), 反应(react), 注视(lookat), 结合(join), 发信号(singal), 移动(move))中的一个。但是, 也可以使用更多或更少的动词, 并且也可以使用其它事件形式。

VC 50 使用当前事件加上一个动画的现存状态(例如, 当前镜头已经延续了多长时间), 以产生一个向重现器模块 54 传送的适当的虚拟摄影机规范。为了产生虚拟摄影机规范, VC 50 可以向实时应用 46 查询 56 额外的信息, 例如各虚拟演员的特定位置和有边界框。有边界框是包围演员全部或一部分的一个框, 这对熟悉本领域的人员是公知的。VC 50 也可以在虚拟演员的位置和运动中进行精细的改变, 称为表演提示。虚拟演员位置上的任何改变也被传送 52 到重现器 54。

重现器 54 利用当前的环境说明, 实时应用 46 传送的演员动画参数 58, 和 VC 50 传送的摄影机规范和表演提示 52 重现场景以便在动画序列中产生一个场景。

表达方式

除了摄影机模块之外, VC 50 还使用表达方式。一个单独的表达方式编码专门知识以捕获一种特定类型的情景, 例如, 两个虚拟演员之间的谈话, 或单一虚拟演员从一点到另一点的运动。表达方式负责决定哪种镜头类型是适合的, 和在什么条件下一个镜头应当过渡到另一个镜头。表达方式还决定什么时候情景已经改变到足以移动到表达方式的专门知识域之外, 例如, 当第三人进入两人的谈话时。

如图 25 中所示, VC 50 中的一个表达方式体现为一个分级有限态自动机(FSM)272。FSM 272 内每种状态 274, 276 调用一个或多个摄影机模块。因此, 每个状态对应于产生的动画中的一个独立的镜头。每个状态还包括一个条件表, 当这些条件中的一个被满足时, 将致使 FSM 272 沿弧线 278-284 中的一个退出到另一个状态。例如, 条件 C 致使 FSM 272 沿弧线 280 从状态 1 274 退出到状态 2 276。结果, 每当横过 FSM 272 中的一个弧线时就隐含地产生一个剪辑。FSM 是分成等级的, 因为每种状态本身可能是一个完整的 FSM, 具有特定的进入和退出状态, 当进入该状态时执行那个 FSM。

图 26 是一个分级 FSM 的实例，其中一个状态的本身就是一个 FSM。表达方式 A 288 具有两个节点：节点 1 290；和节点 2 292。两个节点以弧线 294-300 连接。在这种情况下，标为 2 292 的节点称为父节点，因为如表达方式 B 中所示，它是一个带有四个子状态 304-310 和弧线 312-326 的 FSM 302。应当注意，不管是否包括任何的子状态，都像任何其它状态一样处理父状态 292。

表达方式是用以下的语法定义的：

```
DEFING_IDIOM_IN_ACTION(<name>)
<actions>
END_IDIOM_IN_ACTION
```

```
DEFINE_IDIOM_OUT_ACTION(<name>)
<actions>
END_IDIOM_OUT_ACTION
```

其中<name>是一个表达方式的名称(例如， 2Talk，其显示两个虚拟演员谈话)，<actions>是当进入一个表达方式(IN_ACTION)和退出这个表达方式(OUT_ACTION)之前时为这个表达方式发生的动作。一个动作的例子是 WHEN(talking(A,B)GOTO(1))，表示当虚拟演员 A 和 B 正在谈话时，VC 50 前进到这个表达方式的有限态自动机中的状态 1。动作是以用于实现表达方式的专门编程语言写出的。

一个表达方式的有限态自动机中的每个状态的动作是用下面的语法定义的：

```
DEFINE_STATE_ACTIONS(<name>)
<actions>
END_STATE_ACTIONS
```

其中<name>是一个动作的名称(例如， COMMON，是通用于每个状态

的动作的名称， 1 是状态 1 的名称)， <actions>是为该状态而发生的动作。

在一个表达方式的有限态自动机中的一个状态可以有 in-actions 和 out-actions 。 In-actions 用以下的语法定义：

```
DEFINE_STATE_IN_ACTIONS(<state>)  
<actions>  
END_STATE_IN_ACTION
```

其中<state>是对其将执行 in-actions 的状态的名称，<actions>是 in-action 动作。当一个表达方式的有限态自动机中的一个特定状态在进入这个状态时需要专门的处理时，使用 in-actions 。对于一个表达方式的有限态自动机中的所有或大多数状态不需要 in-actions ， 并可能是空集。 Out-actions 用以下语法定义：

```
DEFINE_STATE_OUT_ACTIONS(<state>)  
<actions>  
END_STATE_OUT_ACTION
```

其中<state>是对其将执行 out-actions 的状态的名称， <actions>是 out-action 动作。当一个表达方式的有限态自动机中的一个特定状态在退出这个状态时需要专门的处理时，使用 out-actions 。对于一个表达方式的有限态自动机中的所有或大多数状态不需要 out-action ， 并可能是空集。

上述摄影机模块(例如， external()150)是由摄影机建立模块定义的。摄影机建立模块具有下述语法：

```
DEFINE_SETUP_CAMERA(<idiom>)
```

```
<camera actions>  
END_SETUP_CAMERA_MODULES
```

其中<idiom>是一个表达方式的名称，<camera actions>建立表达方式的摄影机模块的动作。在本发明的一个实施例中，使用了两种类型的摄影机建立模块：产生模块和链接模块。

MAKE_MODULE(<module id>,<type>,<parameter list>)产生一个带有规定参数的指定类型的新摄影机建立模块，并给予它一个专用的识别模块号码。LINK_MODULE(<state>,<module id>,<name>)将指定的摄影机模块与指定的状态联系在一起。

在本发明的一个实施例中，表达方式代码是用 C++编程语言编写为编程宏的。定义在表达方式内的动作和其它语句是用 C/C++编程语法编写的。但是，也可以使用其它编程语言。用全大写字母写出的关键词(例如，MAKE_MODULE)是编程宏。定义表达方式所用的编程宏的列表在附录 A 中示出。

当编译包含表达方式宏的源代码时，C 预处理器(一般将 C++体现为对 C 编程语言的扩展)把宏扩展到 C/C++源代码。每个表达方式宏的定义包括对实际完成表达方式宏的希望的动作的 C++程序的一次或多次的调用。例如，扩展宏 DEFINE_SETUP_CAMERA_MODULES()以调用一个 C++程序 SetupCameraModules()来完成摄影机模块建立。

在本发明的一个实施例中，有数个可以在状态动作中使用的全局变量，它们包括：Occluded，一个或多个主要演员被遮挡的连续时间单位的数量；T，在当前状态中的时间分段或单位的数量；IdiomT，迄今在这个表达方式中消耗的时间单元的总数量；D[A,B]，两个演员之间的距离(用以下将说明的“头直径”单位测量的)，和 forwardedge[x]，rearedge[x]，centerline[x]，相对于屏幕座标的虚拟演员 x 的有边界框的

边缘。但是，也可以使用更多或更少的全局变量。

还有许多在本发明的一个实施例中使用的预定义控制结构：
STAY，表示把同样的状态再保持一个时间单位；GOTO(x)，表示过渡到状态 x；RETURN，表示返回到父状态，和CALL(idiom,<Parameter List>)，表示通过向指定表达方式传递指定的参数表而执行之。但是，也可以使用更多或更少的预定义的控制结构。

用实例说明在构造表达方式中涉及的概念。在第一实例中，说明了一个称为 2Talk 的，用于描述两个虚拟演员之间的谈话的表达方式。在第二实例中，在建立一个称为 3Talk 的用于描述三个虚拟演员之间的谈话的更复杂的表达方式中把 2Talk 用作一个原语。

2Talk 表达方式实例

2Talk 表达方式实例对拍摄两个谈话和相互反应的虚拟演员的方法编码。它仅使用两个演员的外侧倒摄镜头(72, 84)(图 5)。图 27 示出了 2Talk 有限态自动机 328。2Talk 过程用两个正在谈话的虚拟演员 A 和 B 作为参数。它具有四种状态 330-336。第一状态 330 使用显示 A 对 B 讲话的外摄影机模块 150(图 12)。第二状态 332 用于 B 对 A 讲话时的相反情景。第三和第四状态 334, 336 使用外摄影机模块 150 方位以捕获每个演员的反应镜头。

当 2Talk 表达方式 328 被启动时，它跟随起始于称为“进入点” 342 的小圆圈的两个初始弧线 338, 340 中的一个。被使用的弧线(338 或 340)是由下面的源代码确定的：

```
DEFINE_IDIOM_IN_ACTION(2Talk)
  WHEN(talking(A,B))
DO(GOTO(1); )
  WHEN(talking(B,A))
```

```
DO(GOTO(2); )
END_IDIOM_IN_ACTION
```

这个表达方式测试是 A 正在对 B 讲话还是 B 正在对 A 讲话，并立即过渡(338, 340)到 2Talk 表达方式中的适当的状态，在本例中，分别是状态 1 或 2(330, 332)。

当进入一种状态时，它执行一个用于该状态的 in-actions 集合。in-actions 经常是空集，就象 2Talk 表达方式实例中所有状态的情况。一旦进入一种状态，该状态的摄影机模块(例如，external()150)被调用定位虚拟摄影机。然后，该状态在一预定时间间隔(例如，每个时钟分段)执行一系列的动作。这些动作包括如上所述的 C++过程的执行。此外，这些动作可以被用于影响到其它状态的条件过渡(344-358)。最后，当退出该状态时，它执行一个 out-actions 集合，在 2Talk 表达方式的情况下也是空集。

在 2Talk 表达方式中，摄影机建立模块是如下定义的：

```
DEFINE_SETUP_CAMERA_MODULES(2Talk)
  MAKE_MODULE(1, external, (A, B))
  MAKE_MODULE(2, external, (B, A))
  LINK_MODULE(1, 1, "A talks")
  LINK_MODULE(2, 2, "B talks")
  LINK_MODULE(3, 1, "A reacts")
  LINK_MODULE(4, 2, "B reacts")
END_SETUP_CAMERA_MODULES
```

例如，MAKE_MODULE(1, external, (A, B))生成具有两个虚拟演员 A 和 B 的 external()150 类型的摄影机模块，并赋予它识别号码 1。LINK_MODULE(1, 1, "A talks")表示每当进入状态 1，就要使用一个

如同 MAKE_MODULE(1, external(A, B))定义的越过演员 B 的肩头的演员 A 的外侧倒摄镜头(即, "A talks")。LINK_MODULE(3, 1, "A reacts")表示在状态 3, 演员 A 将对 MAKE_MODULE(1, external(A, B))定义的与演员 B 的谈话作出反应。

可以在一所有状态共用的程序块中规定要在每个状态中执行的第一动作编码。这原来是一种避免在表达方式的每个状态中重新指定相同的(条件, 弧线)对的一种简化机制。但是, 也可以使用没有所有状态共用的程序块的表达方式。

在 2Talk 表达方式中共用动作是:

```

DEFINE_STATE_ACTIONS(COMMON)
    WHEN(T<10)
DO(STAY ; )
    WHEN(! talking(A, B)&&! talking(B, A))
END_STATE_ACTIONS

```

但是, 也可以使用更多或更少的共用动作。一个共用程序块是用与任何其它非共用状态相同的语法定义的。

第一语句(WHEN(T<10))启动一个检查, 看在这个状态中到现在使用的总时间 T 是否少于 10 个时间单位(例如, 10 个时钟分段)。如果使用的少于 10 个时间单位, 那么当前状态保持不变。一个以后将说明的 EXCEPTION(例外)机制优先于对逝去时间的测试, 并且可以抢占镜头。如果镜头已经延续了至少十个分段, 但是 A 和 B 不再谈话, 那么表达方式返回到调用它的表达方式。变量 T 是一个任何状态都可以存取的全局变量。

共用程序块中的动作编码利用一个称为 talking(A, B)的绝对域过

程，当且仅当事件的当前列表包括(A, talk, B)(即，虚拟演员 A 正在对演员 B 讲话)时，这个绝对域过程返回真。在上面所示的编码中，惊叹号“!”是 C/C++的“非”运算符，双&号“&&”是 C/C++的“与”运算符。动作语句是顺序评估的。因此，编码中前面的语句比列在后面的语句优先。

2Talk 表达方式 328 的状态 1 330 被用于描述演员 A 对 B 讲话。除了共用动作之外，在状态 1 330 时每个时间单位 T 执行的动作列表是：

```

DEFINE_STATE_ACTION(1)
    WHEN(talking(B, A))
    DO(GOTO(2); )
    WHEN(T>30)
    DO(GOTO(4); )
END_STATE_ACTIONS

```

但是，也可以使用更多或更少的动作。如果 B 现在正在对 A 讲话，那么需要过渡 344 到状态 2 332，以捕获这种情景。如果一个演员在相同的镜头中已经超过了 30 个时间单位 T，那么应当过渡 356 到状态 4 336，以从其它演员得到反应镜头。

表示演员 B 对演员 A 讲话情况的状态 2 332 是完全对称的：除了 A 和 B 位置颠倒了(例如， talking(A, B))以及用状态 1 330 和 3 334 替代状态 2 332 和 4 336(例如， DO(GOTO (1))和 DO(GOTO(3))之外，代码与上面所示的状态 1 330 的完全相同。

```

DEFIE_STATE_ACTIONS(2)
    WHEN(talking(A, B))
    DO(GOTO(1); )
    WHEN(T>30)

```

```
DO(GOTO(3); )
END_STATE_ACTIONS
```

状态 3 334 的动作代码在下面示出：但是，对状态 3 334 也可以使用更多或更少的动作。

```
DEFINE_STATE_ACTION(3)
  WHEN(talking(A , B))
    DO(GOTO(1); )
  WHEN(talking(B , A) || T>15)
    DO(GOTO(2); )
END_STATE_ACTIONS
```

如果演员 A 现在正对 B 说话，那么需要过渡 352 到状态 1 330，以捕获这个情景。如果演员 B 现在正对演员 A 说话，或“||”如果一个演员在同一镜头中已经超过了 15 个时间单位 T，则应当过渡 350 到状态 2 332。

注意，状态 3 334 可以进行一个过渡 352 返回到状态 1 330，它可以使用像这里状态 3 334 中使用的同样的摄影机模块。在这种情况下，两个镜头合并为一个单一的镜头而没有任何剪辑。

最后，状态 4 336 对称于状态 3 334。

```
DEFINE_STATE_ACTION(4)
  WHEN(talking(A , B))
    DO(GOTO(1); )
  WHEN(talking(B , A) || T>15)
    DO(GOTO(2); )
END_STATE_ACTIONS
```


由于 out-actions 对 2Talk 表达方式 328 是空集，2Talk 表达方式已经被完全说明。可以使用更多或更少的状态，或在每个状态内可以使用更多、更少或替代动作，以定义 2Talk 表达方式。现在可以把 2Talk 表达方式 328 用作一个更高级别的表达方式(例如，处理三个演员之间的谈话)的子程序。

3Talk 表达方式实例

3Talk 表达方式实例是从 2Talk 表达方式 328 建立的。图 28 中示出了用于处理三个虚拟演员之间谈话的 3Talk 表达方式 360 的有限态自动机。这个表达方式 360 实现了 Arijon 描述的和图 8 中所示的三个演员的电影处理。3Talk FSM 具有和 2Talk 相同类型的组成部分：它具有状态 362-368，和代表状态之间过渡的弧线 370-390。此外，这个 FSM 使用下面将讨论的 EXCEPTION 机制。

3Talk 表达方式具有四种状态 362-368。标号 1 的第一状态 362 是对应于图 8 中第一摄影机位置 100 的所有三个演员的固定拍摄。标有 2AND3 的第二状态 364 是调用 2Talk 表达方式 328 的父状态，并对应于图 8 中的摄影机 2 110 和 3 112。标有 4A 366 和 4B 368 的最后两种状态捕获第一演员的反应镜头；这两种状态对应于图 8 的摄影机镜头 4 122。

全部四种状态都具有与 2Talk 表达方式 328 中所述的类似的动作。两个状态 4A 366 和 4B 368 作为独立的状态实现，因为它们在 3Talk 表达方式中的功能不同，尽管它们都是从相同的虚拟摄影机拍摄场景。状态 4A 366 在开始序列中或是在一个新固定拍摄之后使用，使得所有三个演员的镜头能够与演员 A 的反应镜头交替使用。相反，状态 4B 368 仅在演员 B 和 C 之间的双向交谈成为主流时使用，以得到演员 A 的偶尔反应和迅速返回到 B 和 C 之间的双向交谈。

与前面考虑的状态不同的一种状态是标有 2AHD3 364 的状态。首

先，不像以前那些状态，状态 2AND3 364 确实具有以下宏代码表示的 in-actions：

```

DEFINE_STATE_IN_ACTION(2and3)
REGISTER_EXCEPTION(left_conversation,A,LEFT_CONVERSATION);
REGISTER_EXCEPTION(too_long, 100, TOO_LONG);
REGISTER_EXCEPTION(reacts, A, GET_REACTION);
CALL(2Talk, (B, C));
END_STATE_IN_ACTION

```

这些 in_actions 寄存了多个 EXCEPTION，这些 EXCEPTION 在被引发时将致使一个子表达方式退出并使控制返回到父状态。

为寄存一个 EXCEPTION，使用了一个 REGISTER_EXCEPTION() 宏。每个 REGISTER_EXCEPTION (<procedure>, <parameters> <test>, <exception name>)命令采用三个参数：<procedure>，调用测试是否应当引发 exception 的过程名称；<parameters>，传递给该函数的一个任意参数集合；和<exception name>，是一个列举的类型。例如，

```
REGISTER_EXCEPTION(too_long, 100, TOO_LONG);
```

寄存用于调用带有参数 100(即，对一个 TOO_LONG 时间周期测试的时间单位 T 的数量)过程 too_long()的 EXCEPTION TOO_LONG。

状态 2AND3 364 的最后 in-action 动作(即，CALL(2Talk,(B,C))调用 2Talk 表达方式 328,向它传送演员 B 和 C 作为参数。在子表达方式的每个状态中的动作被执行之前，隐含地测试所有寄存的例外(exception)。

当 2Talk 328 在它的一个动作中执行了一个 RETURN，或当引发了一个 EXCEPTION 时，2Talk 328 将返回。在该点，控制被返回父状态，并且它的动作被执行。状态 2AND3 364 的动作是：

```

DEFINE_STATE_ACTIONS(2and3)
  WHEN(EXCEPTION_RAISED(LEFT_CONVERSATION))
  DO(GOTO(1);)
  WHEN(EXCEPTION_RAISED(TOO_LONG))
  DO(GOTO(1);)
  OTHERWISE
  DO(GOTO(4b);)
END_STATE_ACTION

```

在这种情况下，如果 LEFT_CONVERSATION 或 TOO_LONG EXCEPTION 已经被引发，那么产生一个过渡 378 返回到状态 1 362，以获得另一个固定拍摄。否则，产生一个到状态 1 362 的过渡 378，以获得一个反应镜头。

除了 in-actions 以外，状态 2AND3 具有 out-actions。刚好在进行到新状态的过渡之前评价的状态 2and3 的 out-actions 被用于清除由 in-actions 建立的 EXCEPTION。用下面的宏代码表示 out-actions：

```

DEFINE_STATE_OUT_ACTION(2and3)
  DELETE_EXCEPTION(LEFT_CONVERSATION);
  DELETE_EXCEPTION(TOO_LONG);
  DELETE_EXCEPTION(GET_REACTION);
END_STATE_OUT_ACTION

```

这些 out-actions 清除用于寄存在 in-actions 中的并且可能在状态 2and3 的状态动作中执行的 LEFT_CONVERSATION，TOO_LONG，

和 GET_REACTION 的 EXCEPTIONS。

表达方式的等级结构

在本发明的一个实施例中，表达方式的等级结构的产生是从各独立表达方式开始的。如图 29 中的流程所示，使用方法 392 产生一个独立的表达方式。通过定义一个表达方式中的状态 394 和选择用于观察表达方式的定义状态的摄影机模块 396 产生一个独立表达方式。然后，定义包括从一个定义的状态到另一个的过渡的动作 398。这个方法 392 被用于生成 2Talk 表达方式 328。2Talk 表达方式 328 是通过定义 394 四个代表图 8 中所示的四个摄影机镜头(100,110,112,120)的状态 330-336 而生成的。通过选择 396 观察所有四个状态的 external()摄影机模块 150 而观察这四个状态 330-336。然后，为四种状态定义 398 造成从一种状态到另一种状态的过渡的动作。

如图 30 中的流程所示，一旦用方法 392 生成独立表达方式后，使用另一种方法 400 生成一个复杂表达方式。复杂表达方式是用与生成独立表达方式相似的方式生成的。定义复杂表达方式的状态 402。为一个或多个状态，选择一个独立表达方式以定义该状态的动作 404。为没有被选择的独立表达方式的(如果有的话)状态选择将用于观察表达方式的定义状态的摄影机模块 406。定义包括在定义的状态之间的过渡的动作 408。这个方法 400 被用于生成 3Talk 表达方式 360。如上所述，3Talk 是一个允许三个虚拟演员谈话的表达方式。由于已经生成了一个用于两个虚拟演员谈话的独立表达方式(即，2Talk 328)，因而在需要观察仅有两个虚拟演员之间的谈话的状态中使用 2Talk 表达方式。

如图 31 中的流程所示，还有一种方法 410 被用于生成表达方式等级结构。这种方法 410 包括选择两个或更多的独立表达方式 412，组合选择的独立表达方式以生成一复杂表达方式 414。选择一个或多个复杂表达方式和一个或多个独立表达方式 416。组合选择的复杂和独立表达方式以生成一动词表达方式 418。动词表达方式在表达方式等级结构中

是高级表达方式。例如，动词“谈话”是由独立的 2Talk 表达方式和组合的 3Talk 表达方式组合而成的。“谈话”动词表达方式使用 external()150 摄影机镜头允许两个或三个虚拟演员之间的谈话。

图 32 示出了“converse”动词的表达方式等级结构 420 的一部分。“谈话”动词 422 是 3Talk 表达方式 424 与 2Talk 表达方式 426 的组合。这个 3Talk 表达方式(与上述的 3Talk 表达方式 360 不同)使用 external()和 internal()摄影机模块 428,430,并且调用 2Talk 表达方式 426。2Talk 表达方式 426 仅使用 external()摄影机模块 428。其它表达方式动词具有类似的等级结构，其包括最低级的摄影机模块，摄影机模块之上的独立表达方式，独立表达方式之上的复杂表达方式，和复杂表达方式之上的动词表达方式。

使用表达方式等级结构的独立导演风格

用类似的方式可以从低级独立表达方式生成其它动词表达方式或其它高级表达方式。例如，“move”动词表达方式是一个独立表达方式 1Move 和一个复杂表达方式 2Move 的组合。独立表达方式 1Move 允许一个虚拟演员移动，并且使用摄影机模块 tracking()194， panning()204 和 follow()214。复杂表达方式 2Move 是从独立表达方式 1Move 生成的，并且允许两个虚拟演员移动。可以利用方法 392，400 和 410 生成实际上无限数量的表达方式。在本发明的一个实施例中，动词表达方式可以是集合(空闲，谈话，反应，注视，参加，发信号，移动)中的一个。但是，也可以使用更多或更少的动词。

当生成了动词表达方式或其它高级表达方式集或库时，一个导演可以容易地控制虚拟演员以“拍摄”一部虚拟电影。虚拟电影应当用实际上与用于真实影片的技术和摄影机角度相同的技术和摄影机角度“拍摄”。也可以容易地定制动词表达方式以适合一独立导演的风格。例如，导演 A 可以使用如上所述的“converse”动词表达方式。但是，导演 B 根据她个人导演风格可能喜欢在 2Talk 独立表达方式 328 中使用一个

internal()摄影机模块 178，而不是 external()摄影机模块 150。导演 B 可以定制 2Talk 表达方式以使用 internal()摄影机模块 178 适合于她个人喜好。

如图 33 中流程所示，用于虚拟导演的的方法 432 使用上述的表达方式和摄影机模块“拍摄”虚拟电影。VC 根据事件(主体，动词，对象)和运行时状态自动地从动词表达方式集(例如，一个库)选择表达方式 434。组合选定的动词表达方式以生成虚拟场景 436。组合虚拟场景以生成虚拟电影 438。导演根据她本人的风格使用 VC 摄影机模块和表达方式等级结构自由地生成虚拟电影。

虚拟演员的屏幕运动

在用于表现具有虚拟演员的动画制作的视频屏幕上捕捉动作存在着特殊的问题。具体地讲，可能希望不仅在实时系统启动事件时，而且也在演员到达屏幕上的某一位置(例如，屏幕的边缘)时，结束一个镜头。使用上述的全局变量 `forwardedge[x]`，`rearedge[x]`，`centerline[x]`便于这类的测试。

如图 34 中所示，这些变量是在相对于每个演员 442 的方向和位置建立的屏幕坐标系 440 中测量的。演员正面对的屏幕边缘定义为在+1 444，而演员后面的边缘定义为在-1 446。屏幕的中心线在零 448。因此，例如，通过测试 `forwardedge[x]`是否大于 1，一个表达方式中的一种状态可以确定演员 442是否刚好到达屏幕的边缘。通过测试`rearedge[x]`是否小于-1，一种状态也可以测试出演员是否已经完全走出屏幕。也可以用其它方法和坐标系开发表达方式，以捕获屏幕上的运动。

聚会应用

已经把方法 34，392，400，410，432 和系统 44 应用于一个模拟“聚会”环境。图 35 示出了一个用于表现一个聚会参加者显示出的聚会模拟器的用户界面 450。聚会在一虚拟的房间中举行，房间中有两

个各带有虚拟酒吧服务员和两个虚拟吧凳的吧台，一个雕像，和其它虚拟顾客。顾客们可以走动，环顾，彼此交谈，走向他们喝饮料或对酒吧服务员说话的吧台，或他们只是什么都不做。

一个使用者控制一个高度抽象的主角虚拟演员 452 的动作。其余演员 454 可以作为软件代理自动控制，或由聚会的其它参加者控制。具体地讲，使用者可以调用(动词，对象)对，将其转译成(主体，动词，对象)三项组，其中主角演员 452 是主体。当前动词包括：谈话，反应(其造成演员点头)，走到，饮用(其造成演员走向最近的空吧凳)，注视(其造成演员转过他们的头，注视选定的对象几秒，然后转回他们的头)，和空闲。一个动词的每次调用致使对应的按钮按下片刻之后主角演员动作改变。聚会参加者在开始他们自己的谈话之前“有礼貌”地等待说话中的演员停止说话。

一个附加的界面按钮 456 使得独自站立或谈话中的演员可以“投票表决”是接受还是拒绝一个发出要加入谈话信号的演员。发信号动词是在一个演员已经趋近到走向目标的短距离之内时隐含地产生的。其它用户界面按钮 458,460 允许控制虚拟演员附加的功能性。

在每个相对于最大帧频确定的并且在实例中是 5Hz 的时间单位，聚会模拟器发送对主角的私人 VC 有益的(主体，动词，对象)三项组列表。有益的三项组是那些涉及作为主体或对象的主角(或在与主角相同的谈话中的其它演员)的三项组。

聚会模拟器负责演员的所有低级动作。这包括：走动，在谈话时嘴的运动，以及在谈话中或朝向注视的目标时转动头部以面对当前的说话者。

在本发明的一个实施例中，提供了聚会模拟器，重现器 54,VC50 和实时应用 46 以在同一基于奔腾处理器的计算机系统 10 上同时运行。

在本发明的另一个实施例中，聚会模拟器和系统 44 的组成部分是作为由一计算机网络(例如，国际互联网络，内部网)连接的独立的远端模块提供的。界面 450 的源代码是以微软公司(Microsoft Corporation of Redmond, Washington)的 Visual C++和 Visual Basic 的组合实现的。但是，也可以使用其它编程语言来提供界面 450。重现器 54 使用来自微软公司的 Direct3D(Rendermorphics)产生每一帧。但是，也可以使用其它重现软件。整个系统是以大约每秒 5 时间单位的速度运行的，大部分时间是消耗在系统的重现器 54 中。

已经证实 VC 50 在不用与聚会事件或谈话有关的信息以前在聚会模拟实时设定中是十分健全的。在演员动作的高级控制与以 VC 50 编码的表达方式等级控制之间有很好的匹配。

系统 44 的使用是很直观的。在本示范实施例中使用者不必利用自动化的摄影机控制来控制摄影机。镜头和切换提供了信息丰富的事件叙述。越过使用者肩头观看的观众也能够容易地跟随动作，而不必知道使用者发出的特定命令。

以下是 VC 50 中声明的电影摄影原则：摄影机角度和移动，在摄影机模块中捕获；不要越线，摄影机模块中适当情况的选择；避免跳跃剪辑，在改变表达方式状态时表达方式构造加强摄影机模块有意义的变化(或无变化)；使用固定拍摄，表达方式的等级结构；让演员引导，表达方式中动作调用弧线过渡；间断运动，由全局 T 值调节的表达方式过渡弧线；保持连续运动，摄影机模块中情况的选择；镜头成帧，摄影机模块方位，表演提示以再定位演员，和由特定导演设计的风格、特定表达方式和表达方式等级结构。

通过把真实制片人开发的专门知识编码成等级结构的有限态自动机，VC 50 自动地生成用于独立镜头和动作展开时这些动作的序列的摄影机控制。结果显示为对聚会模拟器的实时设置中的预先未知偶然事

件是十分健全的。此外，VC 50 处理有效处理几何细节的低级摄影机模块；允许表达方式的域相关等级结构以捕获简单、健全和可标度的导演专门知识。VC 50 也为重现器 54 提供表演提示。提示包括精细地去除演员以清除虚拟摄影机画面；精致地重新安排演员；和根据演员的屏幕尺寸修改表演尺度。

VC 50 被使用于虚拟现实和其它交互式应用，以改善这类应用一般使用的固定视点镜头或安装在顶棚上的摄影机。尽管基本上是参考一种虚拟 3D 闲谈环境进行说明的，但 VC 50 也通过允许使用者利用一位于看来自然的摄影机位置的代理看到他们自己而帮助改善“智能代理”用户界面的下次产生。

在一种虚拟 3D 闲谈环境中，计算机使用者通过由计算机使用者控制和代表计算机使用者的动画 3D 虚拟演员(有时称为替身)相互交流。虚拟演员重现在一 3D 房间的显示中，在该 3D 房间中根据那些正在进行相互通信的使用者安排虚拟演员。以类似的方式，VC 50 可以提供一“智能代理”用户界面，在该界面中相对于在 3D 显示空间中的相互关系安排和定位智能代理和任何潜在的计算机应用或多个计算机应用。

智能代理一般是一种在各种相互关联的条件中使用的高度交互式用户界面设备(例如，一种字符)，例如向一使用者提供软件应用帮助或援助。在一个实施例中，VC 50 可以给智能代理适当定位，使得它不会遮挡智能代理正在提供援助或能够重现的应用屏幕。在另一个实施例中，系统的使用人被赋予一个 3D 显示空间中的虚拟位置，这个虚拟位置也被 VC 50 使用以形成摄影机画面和结果的显示。

应当懂得，这里所说明的程序，处理和方法并不涉及或限于任何特定类型的计算机装置，除非另有指示。根据这里所述的技术，可以使用各种类型的一般用途或专门用途的计算机装置或用它们进行操作。

鉴于可以使用我们的发明的各种各样的实施例，应当理解说明过的实施例只是示范性的，不应当作为对我们发明范围的限制。而是，我们要求凡是落入下面的权利要求和其等同物的范围和精神内的所有这种实施例都是我们的发明。

图 1

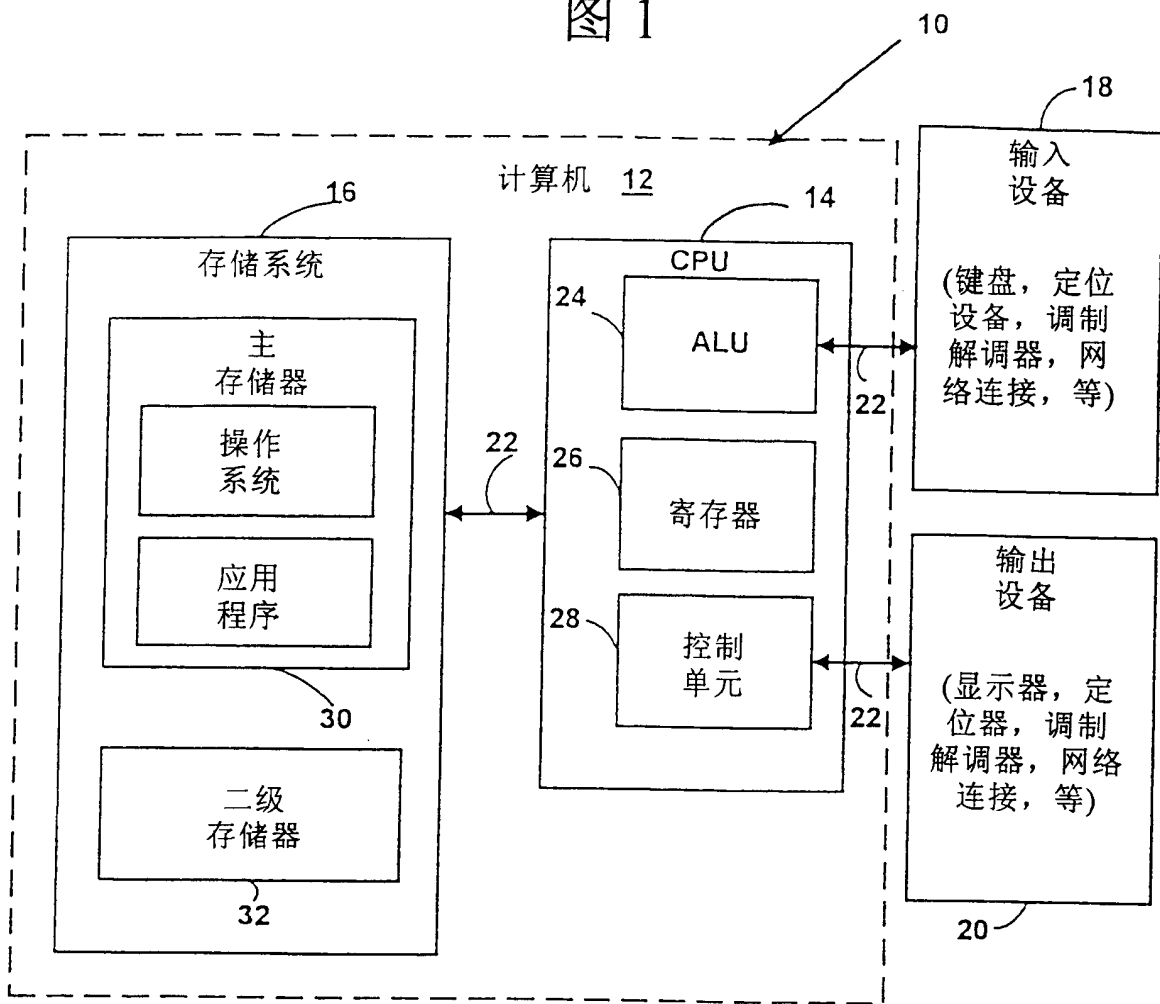
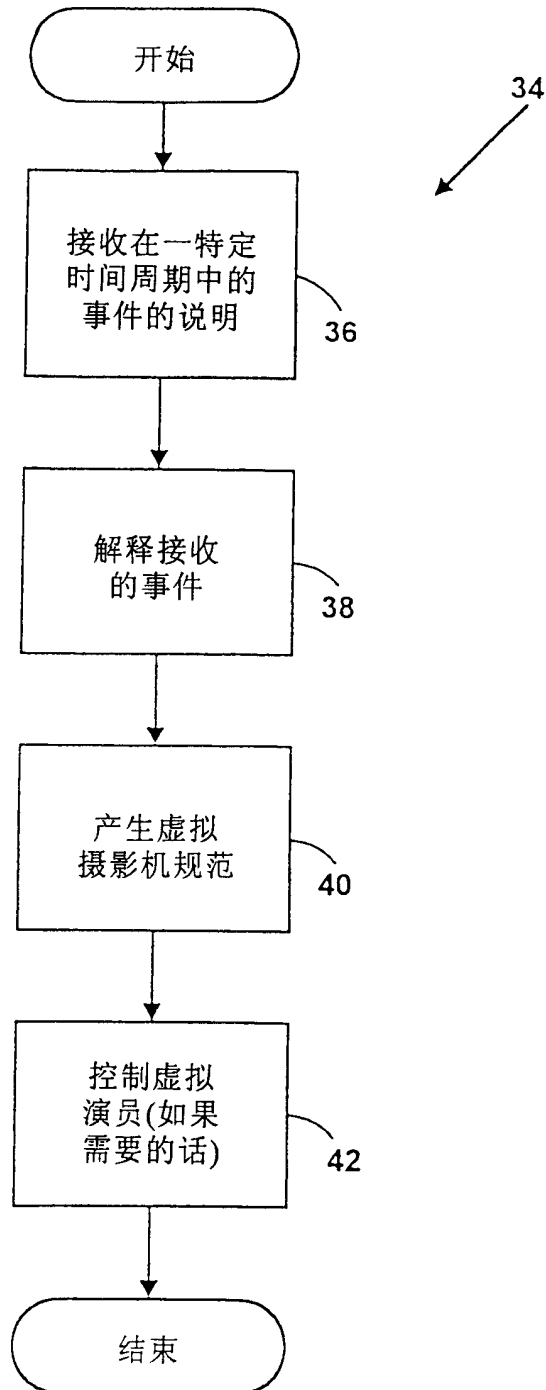
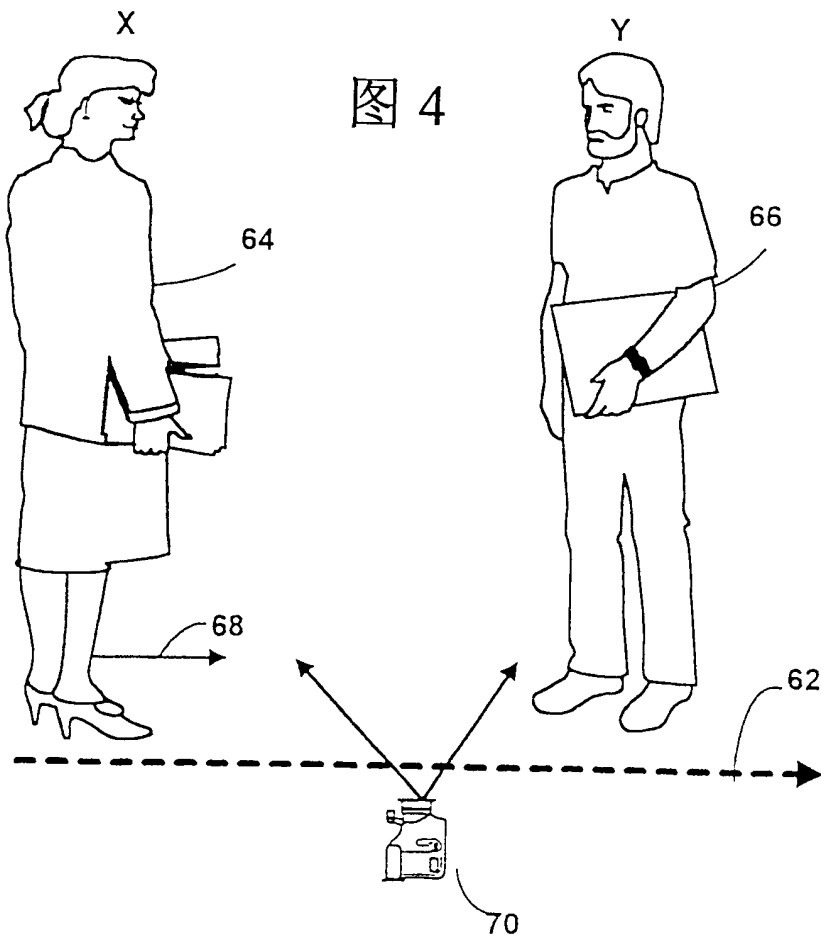
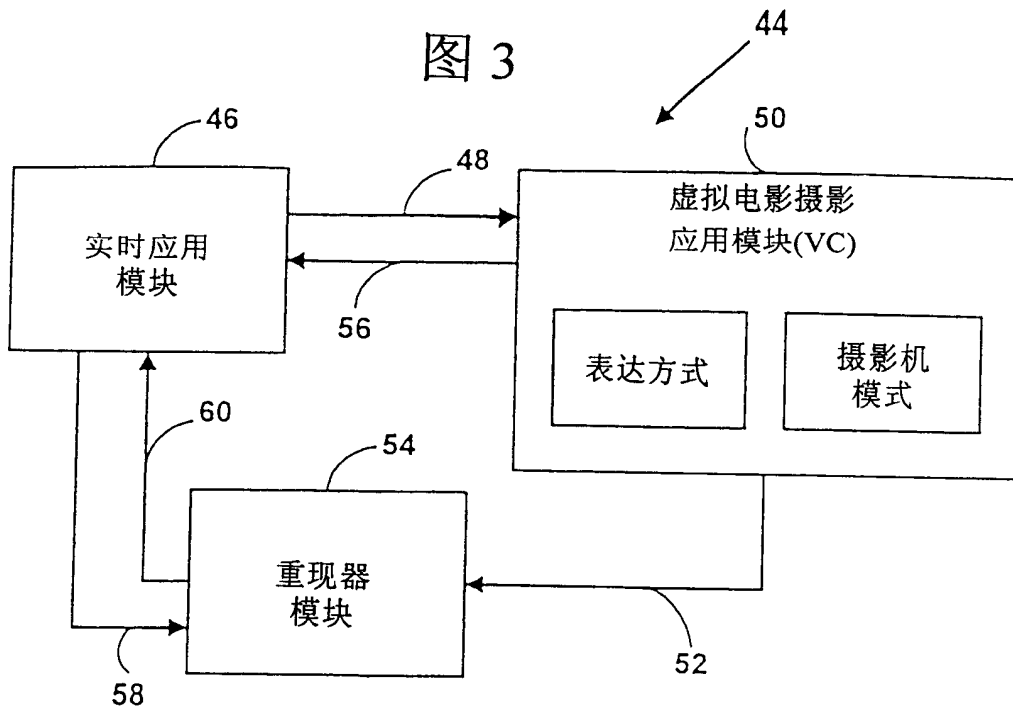


图 2





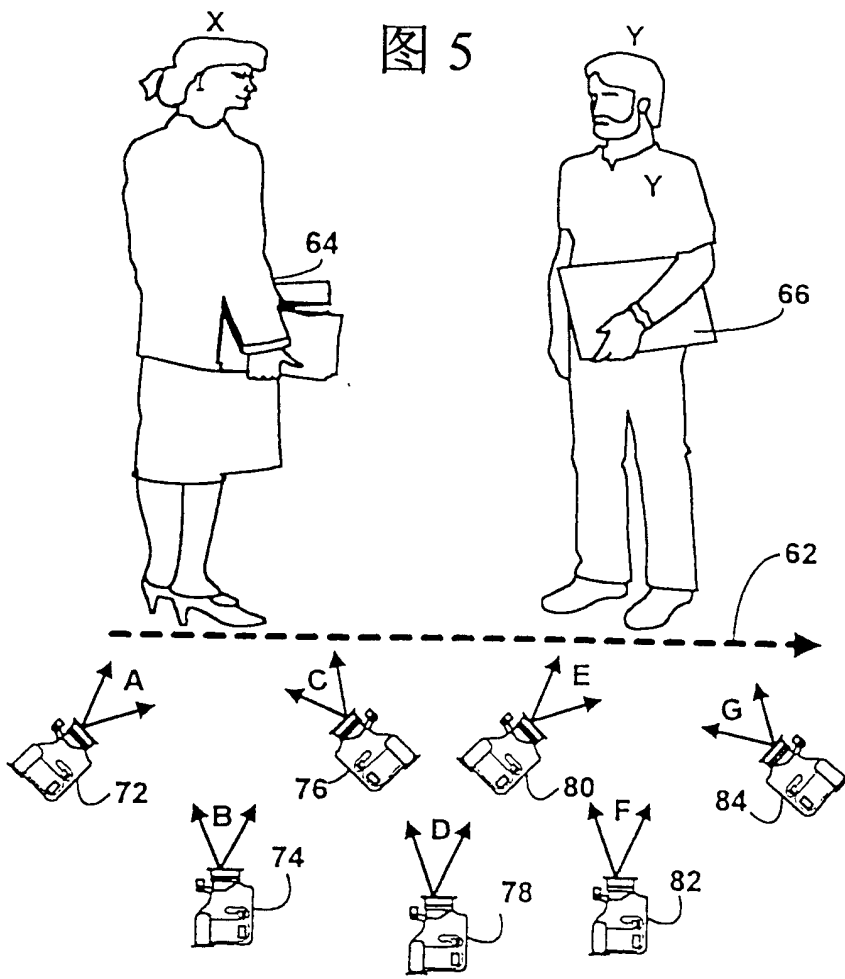


图 6

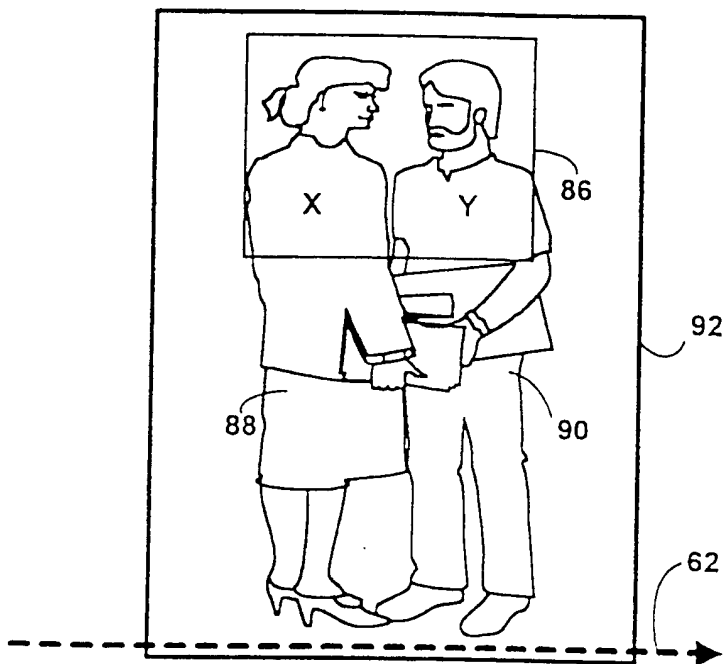


图 7A

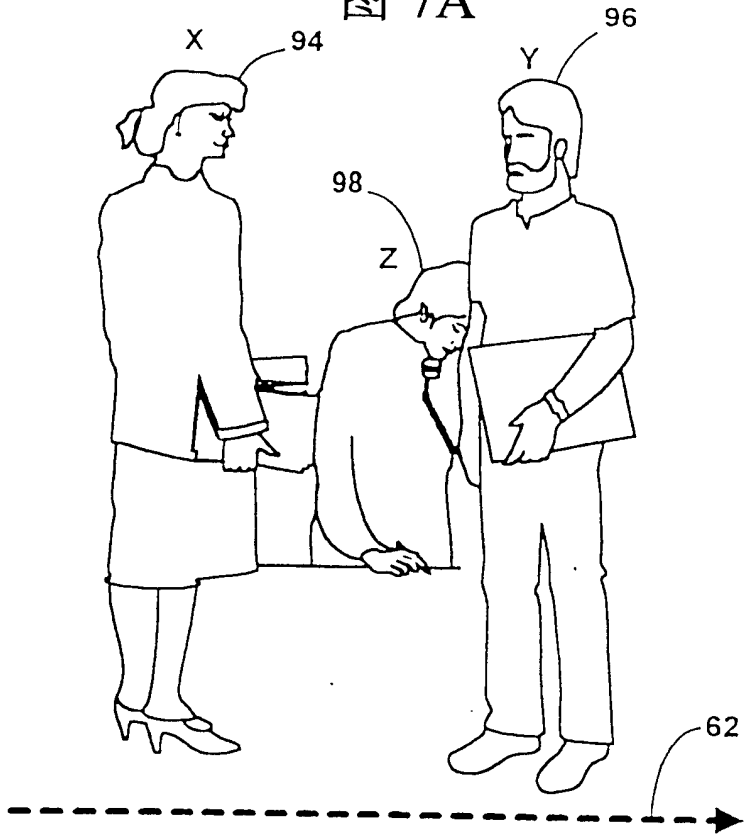


图 7B

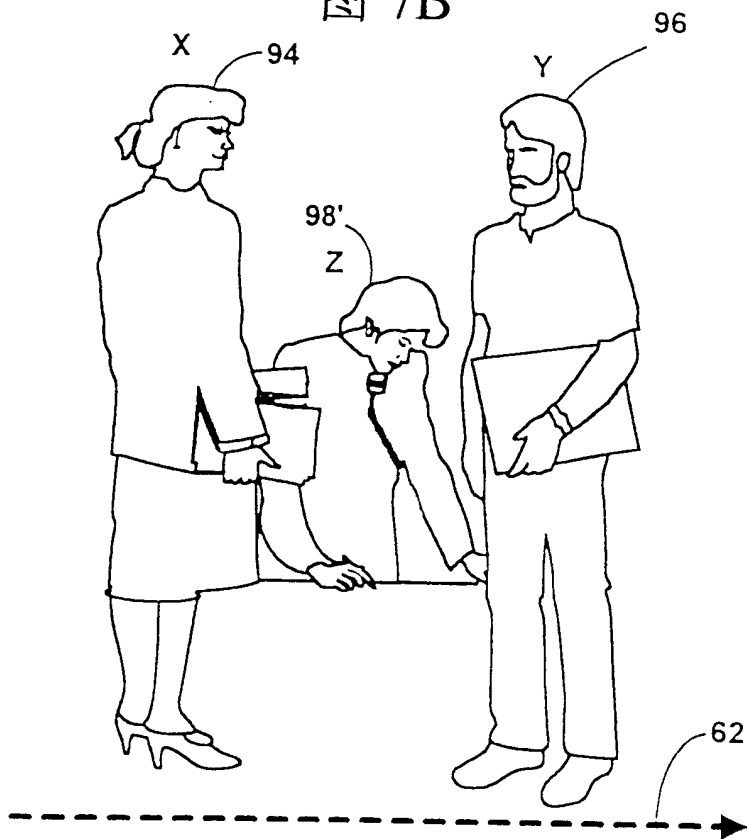


图 8

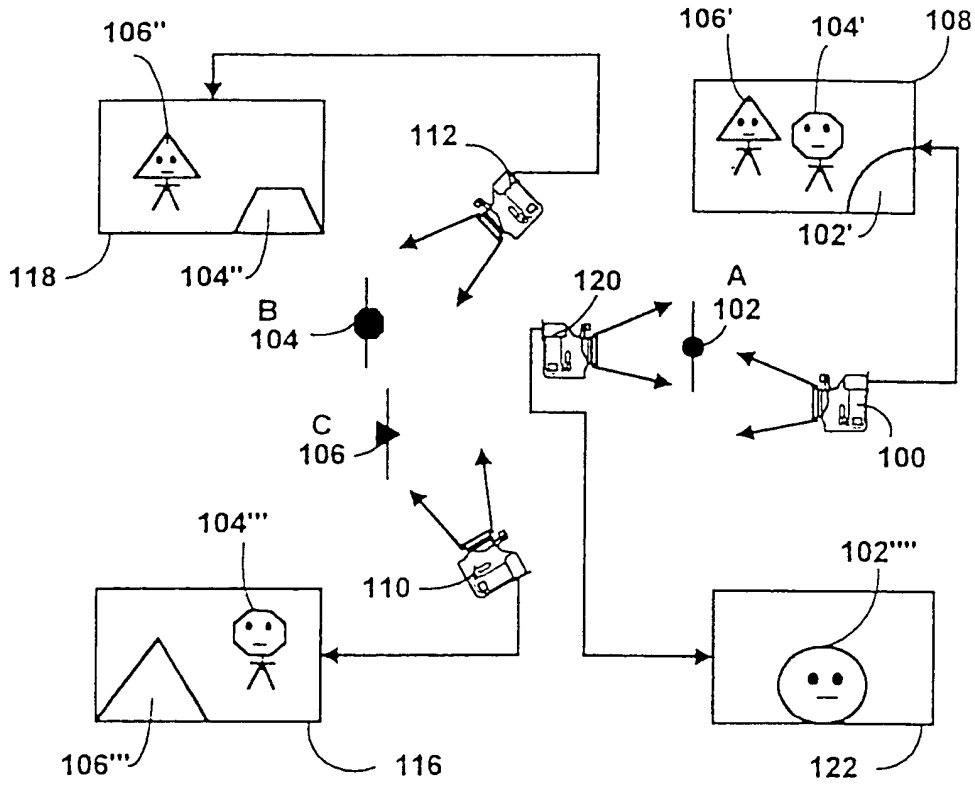


图 9

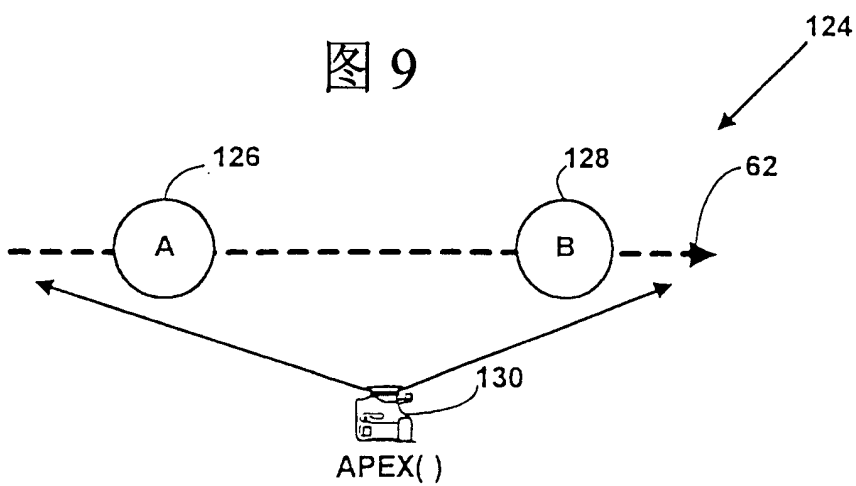


图 10

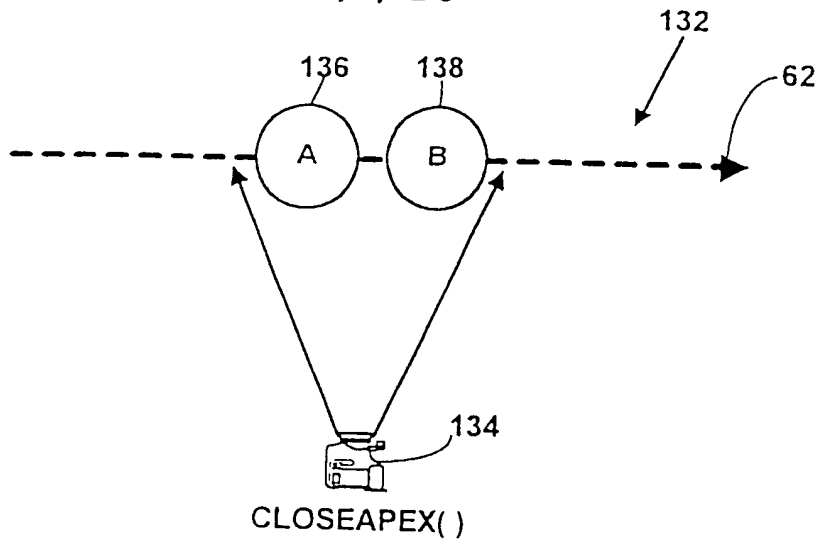


图 11

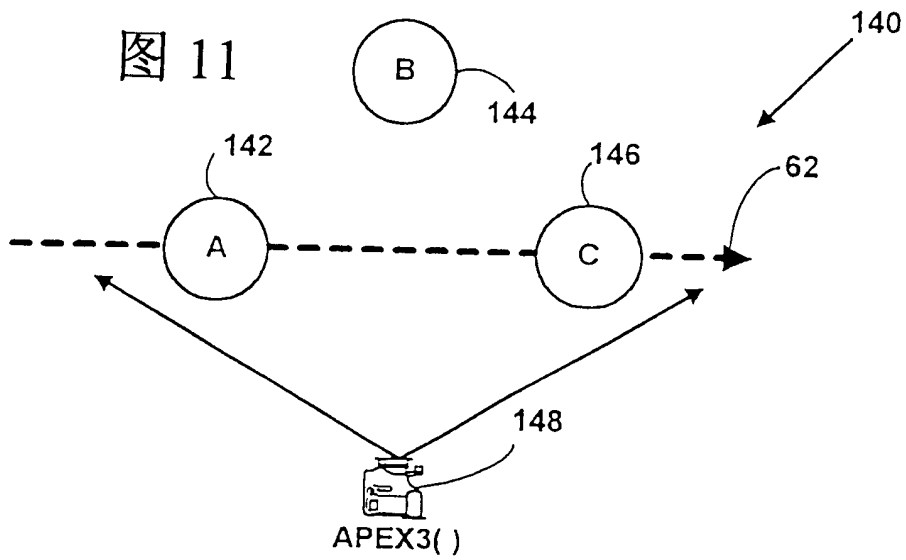


图 12

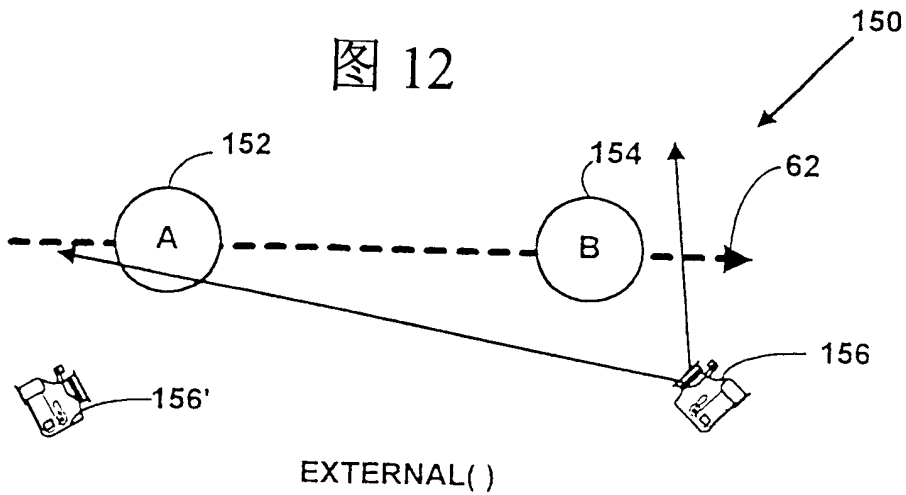


图 13

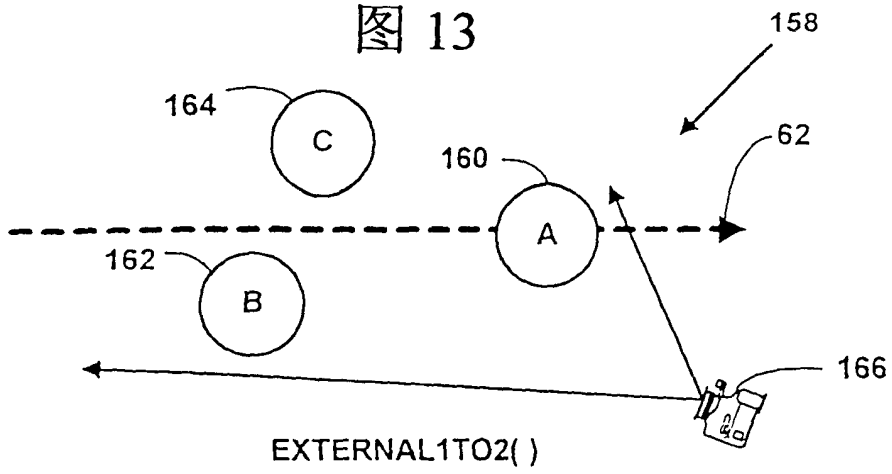


图 14

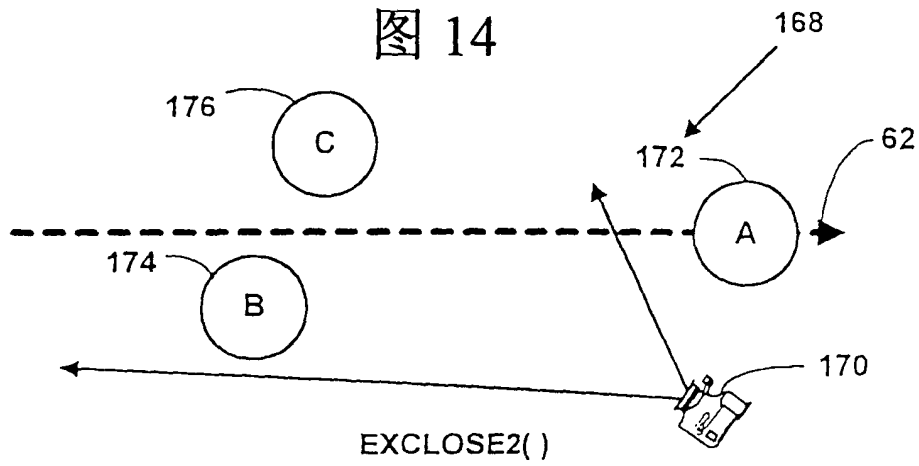


图 15

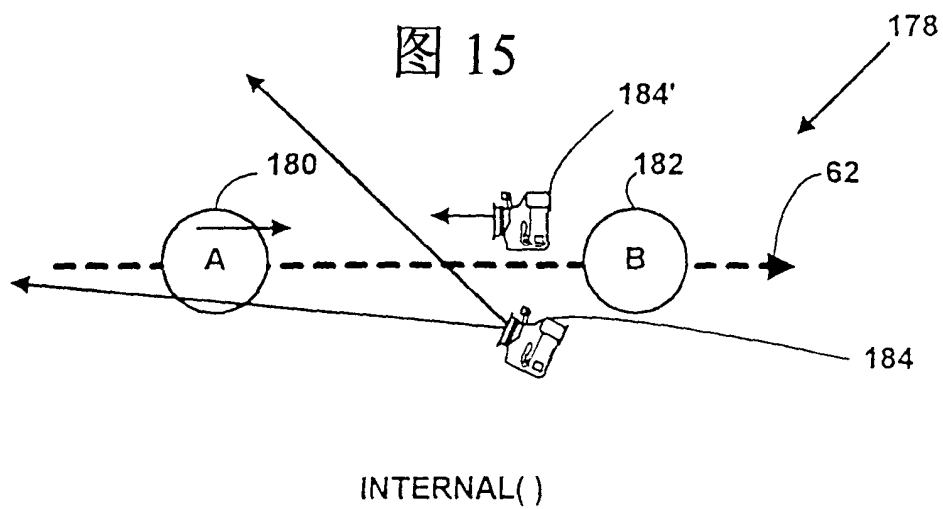


图 16

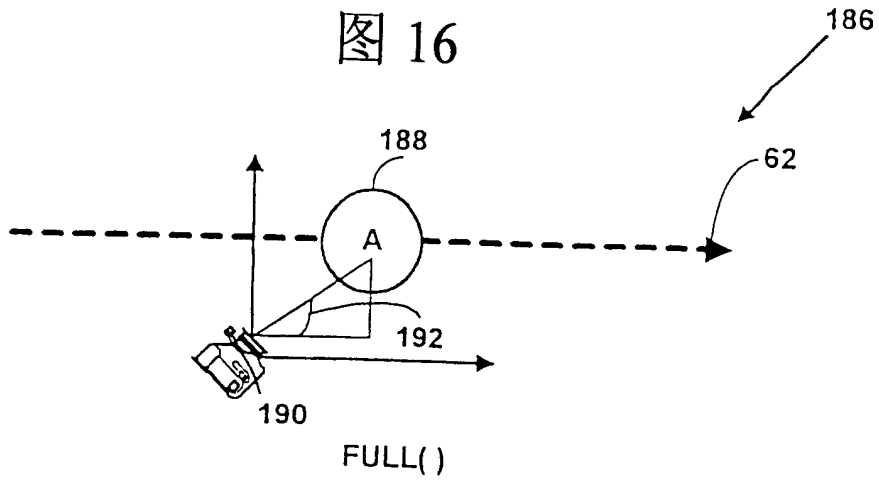


图 17

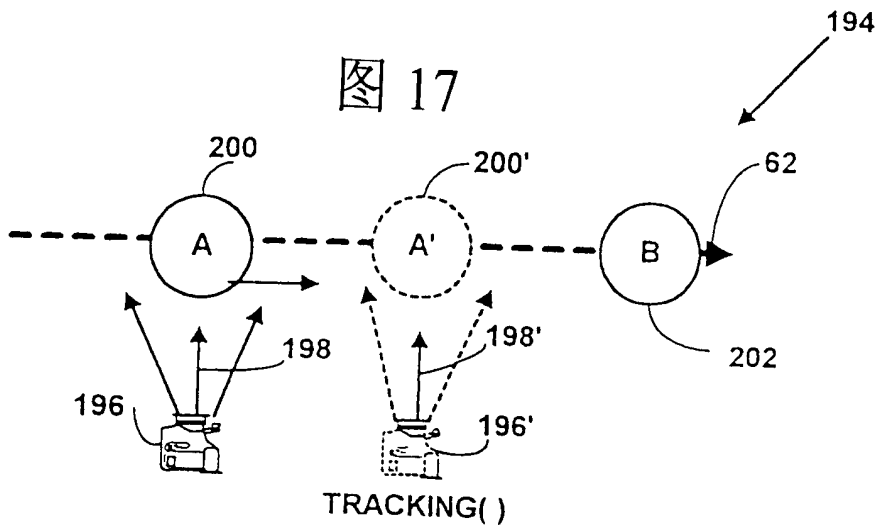
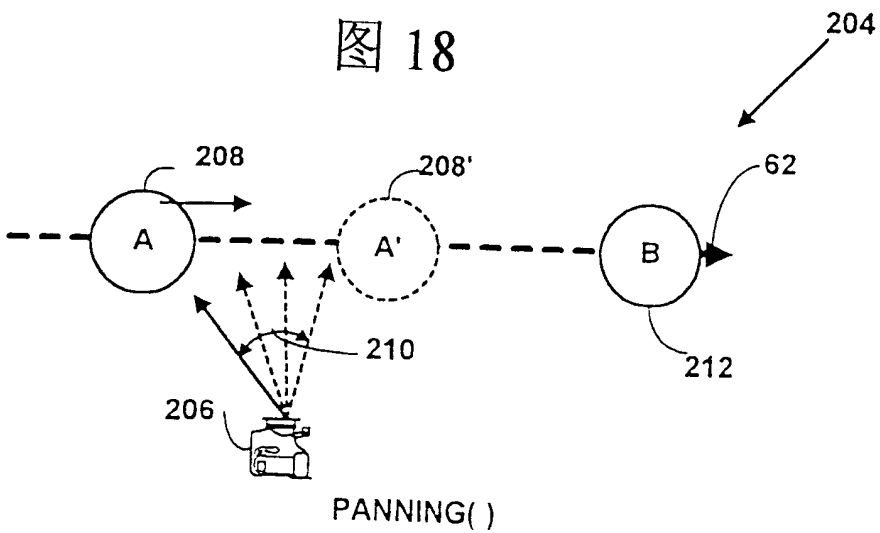
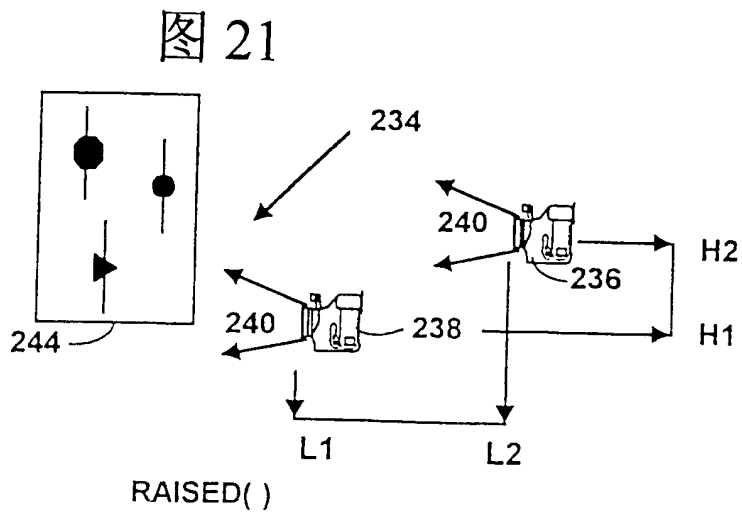
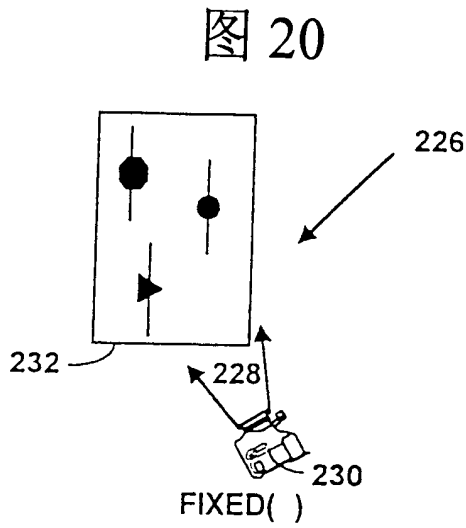
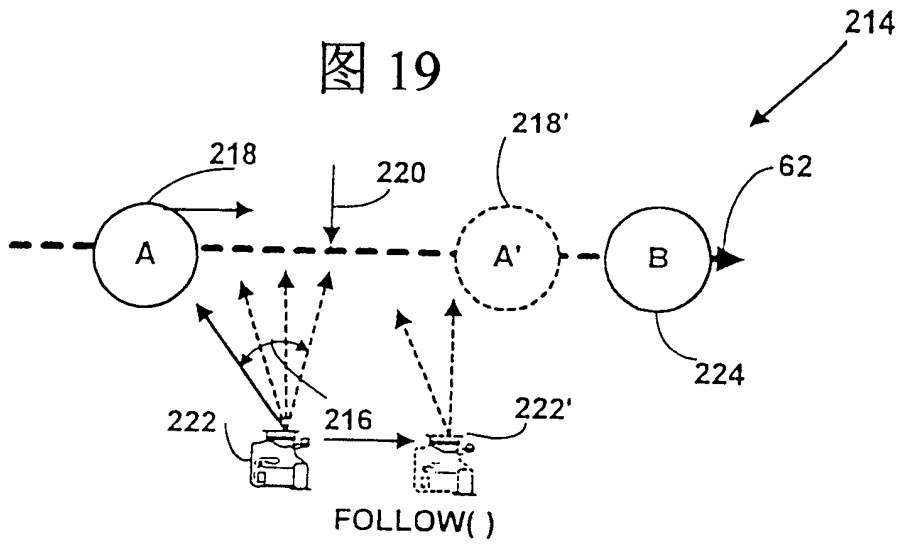


图 18





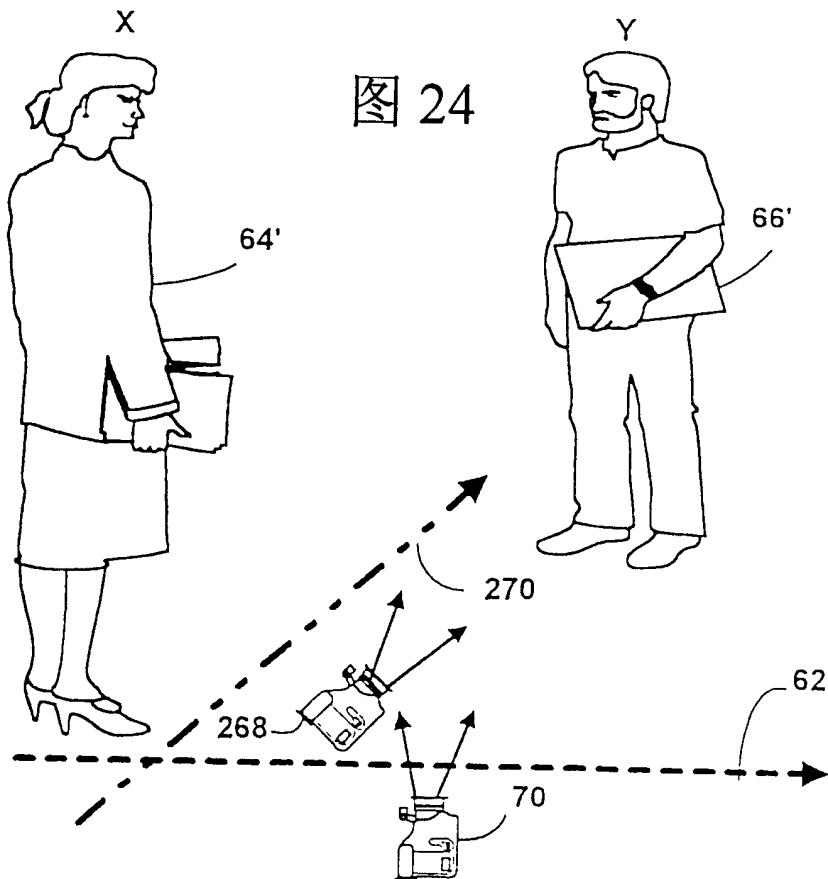
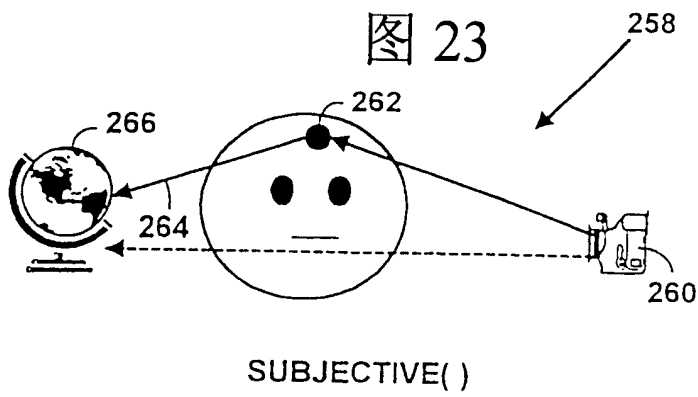
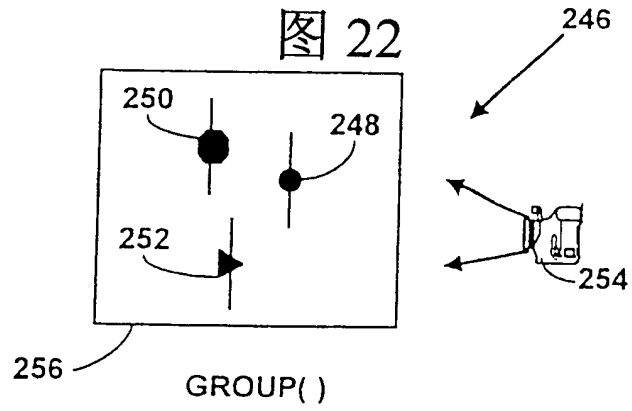


图 25

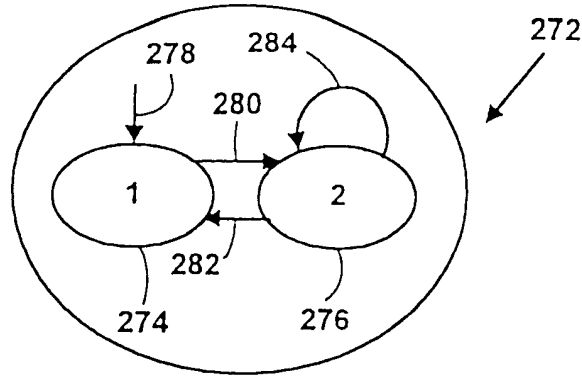


图 26

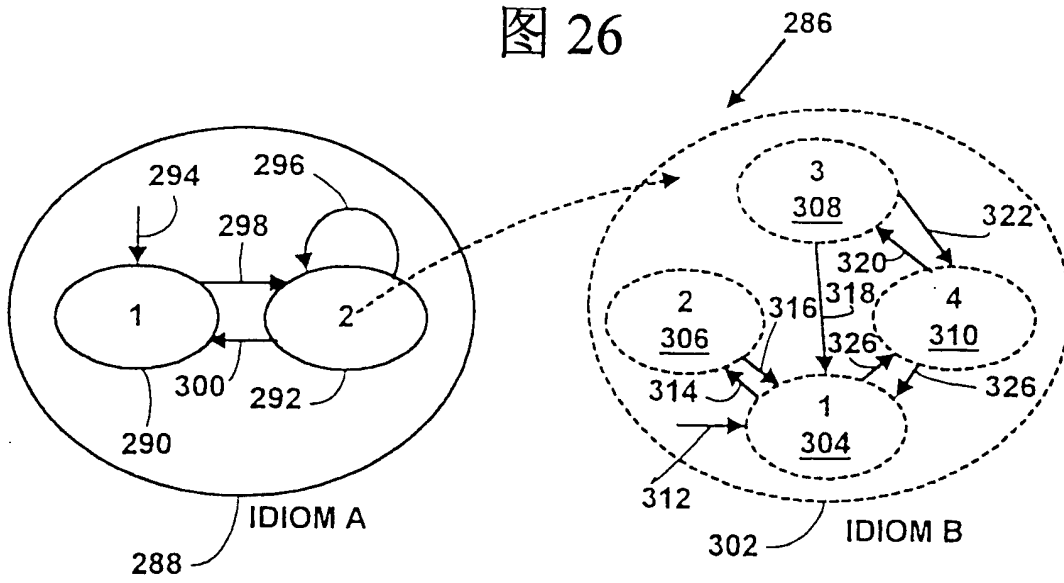


图 27
2TALK 表达方式

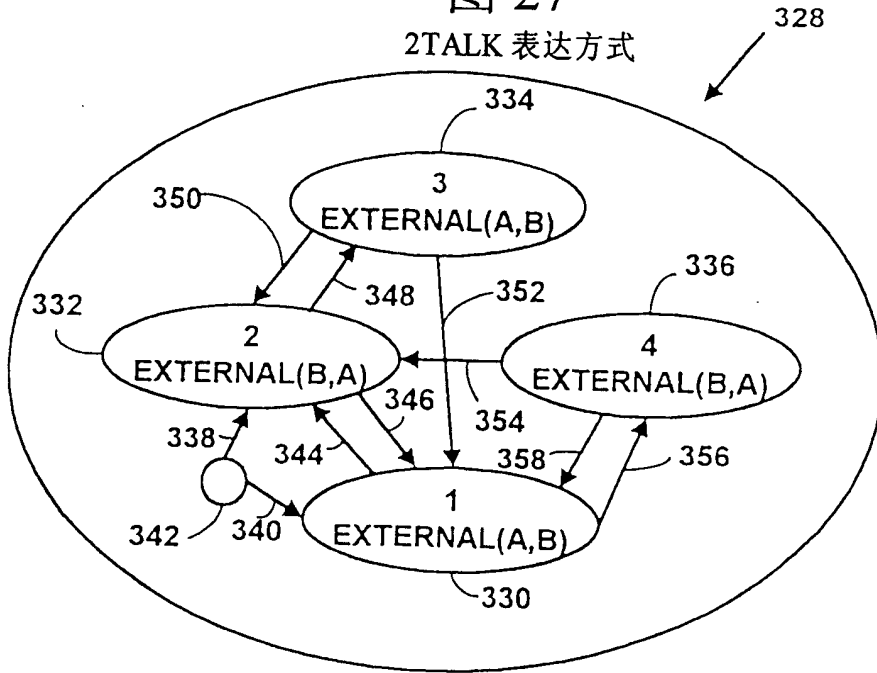


图 28
3TALK 表达方式

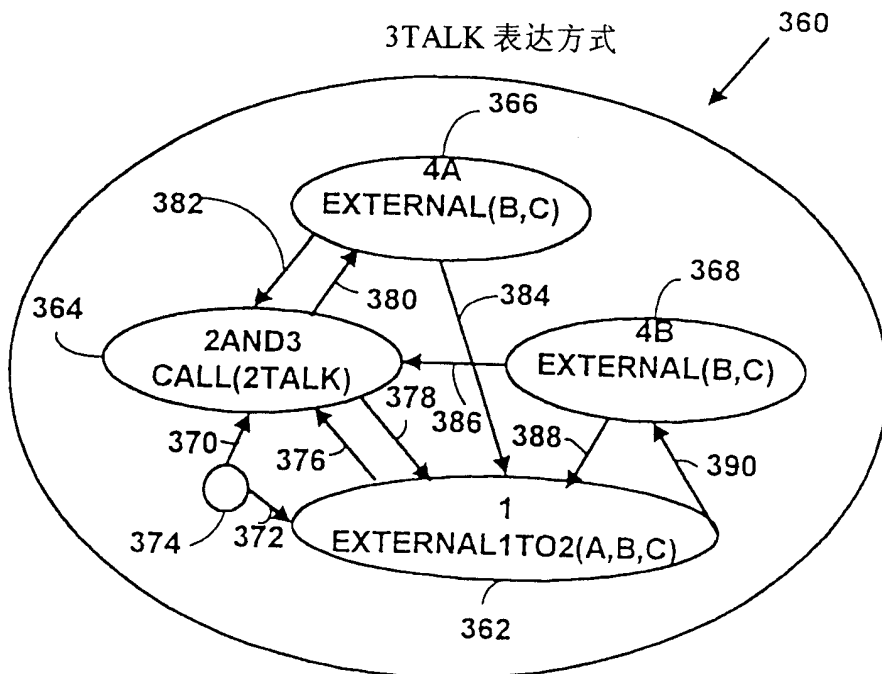


图 29

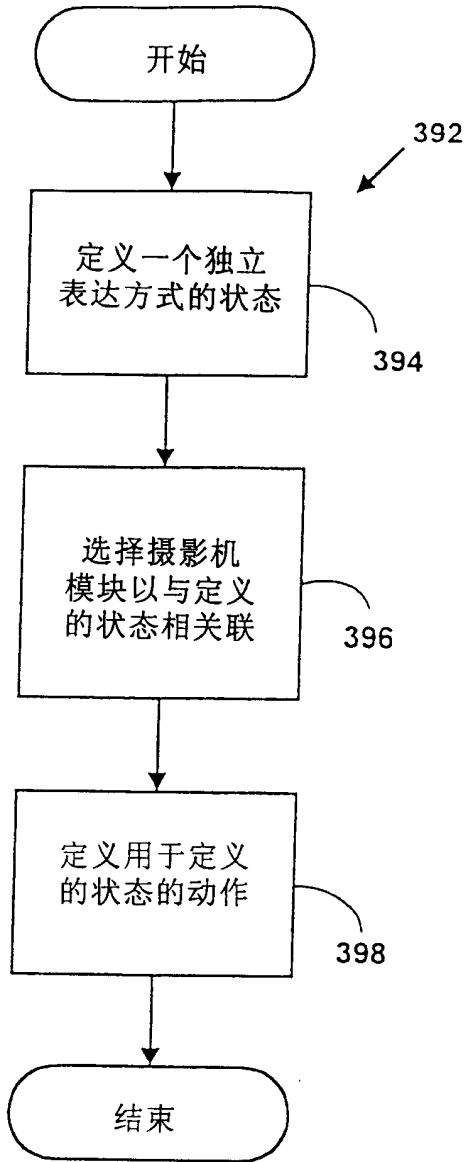


图 30

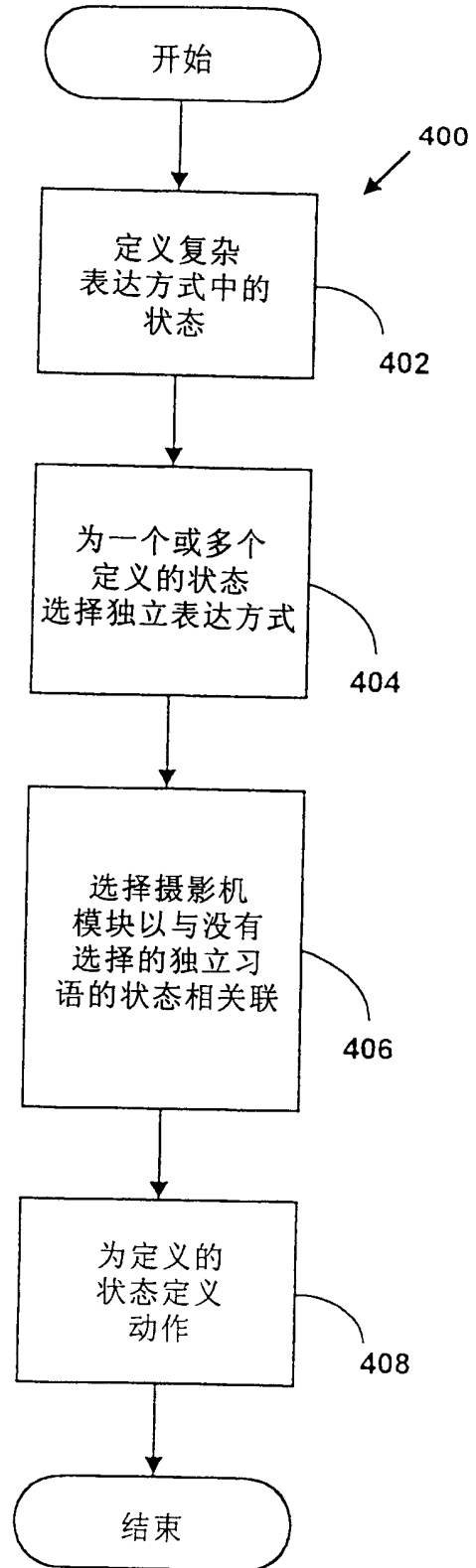


图 31

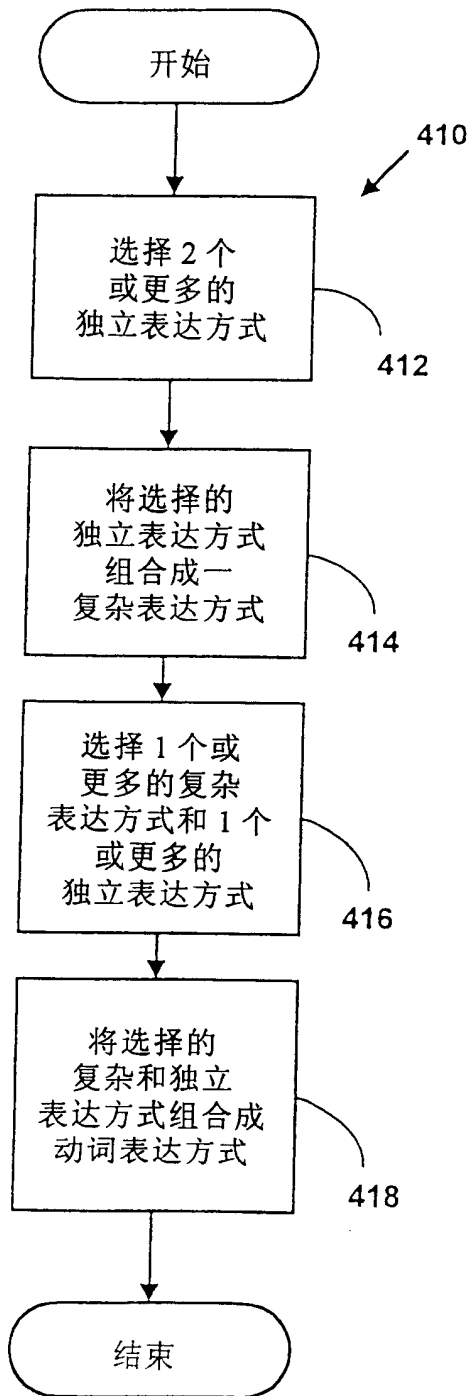


图 32

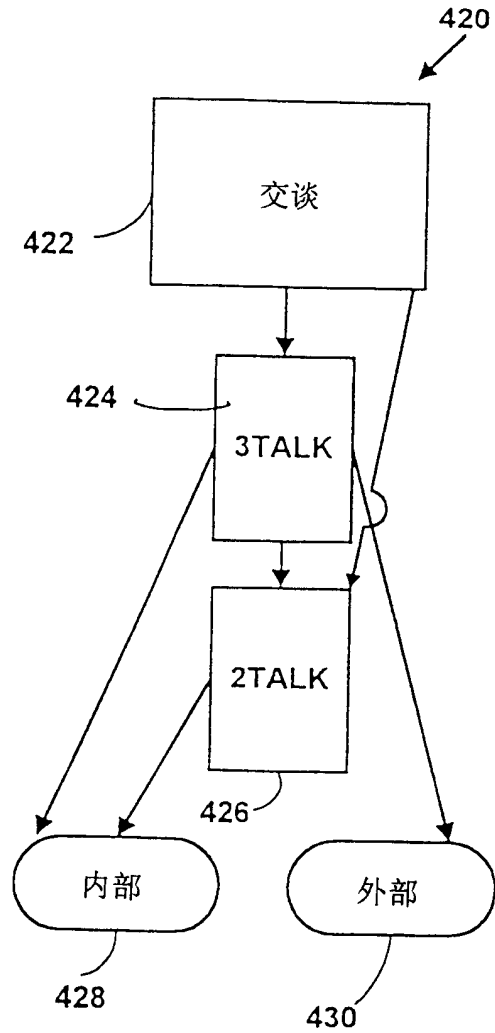


图 33

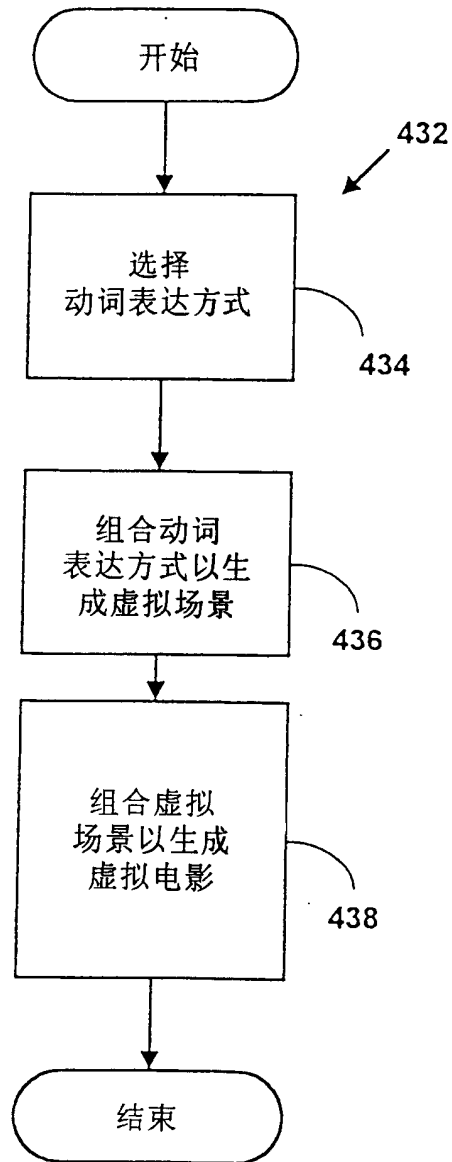


图 34

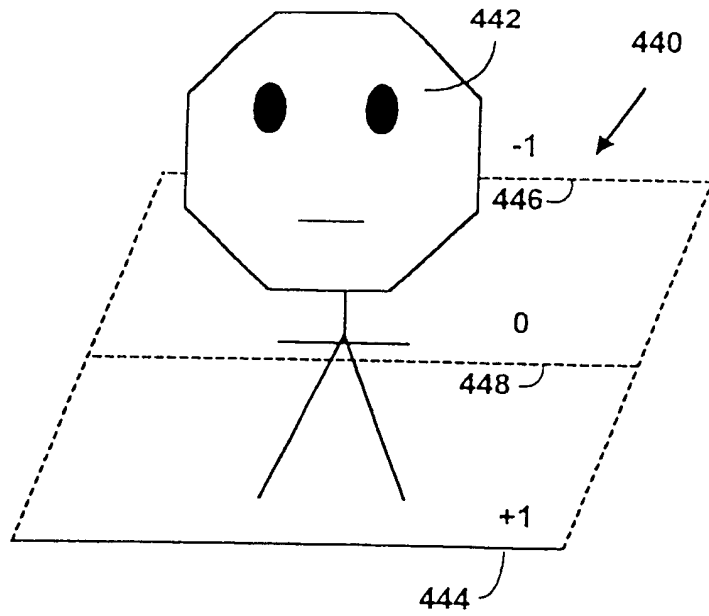


图 35

