

三维交互任务的描述和结构设计*

田 丰, 戴国忠, 陈由迪, 程 成

(中国科学院 软件研究所 人机交互技术与智能信息处理实验室, 北京 100080)

E-mail: tf@iel_mail.iscas.ac.cn

http://iel.iscas.ac.cn

摘要: 主要讨论了三维交互中交互任务的归纳描述和结构设计问题. 首先归纳出三维交互中的基本交互任务和普遍交互任务, 对二者之间的联系进行了描述, 提出了一个三维交互任务的产生框架, 最终建立起三维交互中通用的核心模块. 三维交互核心模块描述了三维交互中通用的底层结构, 模块中定义了三维交互中通用的一系列交互原语和交互任务, 实现了原语的产生和任务的整合工作. 在开发三维用户界面构造系统(3D UI toolkit 或 3D UIMS)或具体的三维/虚拟现实应用时, 可以直接用它来建立系统的底层交互结构, 而程序员将只关注于交互风格的建立等高层工作.

关键词: 三维交互; 交互原语; 交互任务

中图法分类号: TP311 文献标识码: A

随着计算机交互设备和交互技术的不断发展, 二维图形用户界面的局限性越来越明显地体现出来. 首先从界面的信息表示能力来看, 二维图形技术的一个重要的缺点是不能用一种自然的方法表示复杂的多维关系^[1]. 在表示多个关系时, 不得不使用多窗口环境来表示应用信息. 多窗口的划分使得用户需要更大的认知努力以建立起一个完整和一致的认知模型. 其次, 从交互方式而言, 对于一些领域内的应用(如虚拟装配环境等)就无法用传统的交互方式来进行自然、合理的表达. 当需要同物体进行三维交互时, 只能通过不同输入方式的组合来实现. 这种方法一方面很不自然, 另一方面大大增加了用户的交互难度, 同时也加重了交互任务的整合工作. 与之相比, 三维用户界面具有很大的优点: 首先, 在一定的应用领域内信息的三维表示较为直观, 与现实世界中的对象较为接近, 人们较容易感知这种表示并上升为一种理性认识; 其次, 虚拟场景中的交互方式通常采用模拟现实世界中人与物体的交互方式. 这种交互方式对于交互语义的表达是自然和清晰的, 一方面用户易于理解和掌握, 另一方面交互系统容易定义和实现.

三维用户界面相对于二维用户界面需要全新的软件工具和交互技术. 它的发展还远未成熟, 只有当对相关的 3D 交互技术和 3D 交互组件(3D Widget)以及信息的三维表示范型研究成熟后, 才能开发出成功的、能被广泛应用的三维界面^[2]. 对于一个新兴的信息系统而言, 进行底层的信息描述是实现整体系统的基础. 正如 Mark Green 所说, “三维用户界面的构造目前之所以具有相当大的难度, 其原因正在于对底层交互任务的理解上. 只有当这些底层的交互任务被定义和描述清楚之后, 才能合理而有效地进行三维交互技术、交互风格的研究和三维用户界面的构造”^[3].

目前国际上对三维交互任务主要是从基本交互任务(basic interaction tasks)^[4]和 VR/3D 中的普遍交互任务(universal interaction task)两个角度进行研究^[5]. 基本交互任务是交互系统中人机交互的最小组成单位. 用户在

* 收稿日期: 2001-05-28; 修改日期: 2001-07-16

基金项目: 国家自然科学基金资助项目(60033020)

作者简介: 田丰(1976 -), 男, 陕西绥德人, 博士生, 主要研究领域为人机交互技术, 虚拟现实; 戴国忠(1944 -), 男, 江苏无锡人, 主任研究员, 博士生导师, 主要研究领域为人机交互技术, 计算机图形学; 陈由迪(1933 -), 男, 浙江宁波人, 研究员, 主要研究领域为辅助设计, 计算机图形学; 程成(1966 -), 男, 安徽黟县人, 博士生, 主要研究领域为虚拟环境人机交互.

执行一个基本交互任务时,产生了对交互系统有意义的最小的信息.在二维图形用户界面中,Foley 对基本的交互任务的总结和归纳已经很完善了.Position Task, Select Interaction Task, Text Interaction Task, Quantify Interaction Task 构成了一个完整的基本交互任务集^[4].同时 Foley 指出,在三维图形界面中 Position Task 和 Selection Task 由于涉及到第三维的信息将会变得复杂,同时交互任务中应该加上 Rotation Task.与 Foley 的思想类似,GKS-3D^[6]和 PHIGS^[7]对逻辑输入设备所产生的逻辑输入值进行了分析和归纳.在 GKS-3D 中,逻辑输入值被分为 6 大类:Locator,Stroke,Valuator,Choice,Pick 和 String,这些逻辑输入值相当于 Foley 所描述的基本交互任务产生的交互原语.普遍交互任务从 3D/VR 应用需求的角度来研究三维交互任务,它是指从应用中归纳出的对应用而言有意义的三维交互任务.Jeff Pierce 在进行新的交互技术研究之前,列出了 Navigation, Moving Object, Accessing Storage 三种 3D Desktop 环境下常用的交互任务^[8].Bowman 指出,对于 3D/VR 交互系统而言,存在 4 个普遍的交互任务:Navigation, Manipulation, Selection/Pick, System Control^[5].

分别从基本交互任务或普遍交互任务两个角度来独立地进行交互任务的分析和归纳都有其局限性.基本交互任务的思想是从用户的角度来考虑交互过程的,用来描述信息从用户到交互系统的传递,将信息基本上化为若干个基本交互任务的组合.但是,如果从应用领域来考虑(比如对于 3D 以及 VR 应用),基本交互任务还需要进行一定的整合才能转化为对于应用有意义的最小信息单位.普遍交互任务虽然从应用的角度来分析和归纳交互任务,便于进行应用中交互模块的构造,但它与用户所产生的交互信息之间缺少必要的联系.本文通过基本交互任务和普遍交互任务的结合来研究 3D UI 中交互任务的归纳描述和结构设计问题.首先通过不同的方法归纳出 3D UI 中的基本交互任务和普遍交互任务,同时对二者之间的联系给出了描述,从而提出一个三维交互任务的产生框架,最终建立起三维交互中通用的核心模块.三维交互核心模块实现了用户的输入信息向交互任务的转变过程.在模块中定义了三维交互中通用的一系列原语和任务,实现了原语的产生和任务的整合工作.

1 3D 交互任务的分析、归纳和描述

可以看出,普遍交互任务的抽象层次是在基本交互任务的层次之上的,二者的划分各有优点.基本交互任务的思想便于我们对各种交互设备所产生的交互信息进行归纳和总结,定义三维交互中通用的交互原语.普遍的交互任务便于我们对 3D/VR 交互系统进行构造.我们在这里采用基本交互任务同普遍交互任务相结合的方法.首先归纳出较为完整的三维基本交互任务集合以及三维普遍交互任务集合,然后建立起二者之间的联系,并由此构造出 3DUI 中通用的交互框架.我们这里对基本交互任务和普遍交互任务的归纳采用两种方法的结合.一方面,我们在对二维用户界面中的基本交互任务进行扩展的基础上,分析三维交互应用中交互设备所产生的交互信息,以此得出基本三维交互任务.同时,我们通过对目前各类主要的 3D/VR 应用中所需要的交互任务的分析,归纳出普遍的三维交互任务.

首先,我们进行基本三维交互任务的总结.在此之前,我们先来看一下二维图形用户界面中的基本交互任务:定位、选择、字符交互和数量值交互.显然,在三维交互中这几种基本交互任务都应包含于其中,但复杂性应有所增加.对于定位而言,在三维交互中不仅需要深度定位的参数,同时还要添加空间的方向定位,我们在这里用一个六自由度定位的交互任务来统一表示三维交互中的定位任务.对于在三维空间执行选择任务时同样需要添加深度的信息,同时针对在三维交互中的需要,在二维图形交互中被忽略的释放任务在这里需要作为一个独立的交互任务提出.字符交互和数量值交互相对于二维图形交互没有太大变化.另外需要考虑的是,在三维交互中一些新的交互设备所产生的信息有可能增加基本的三维交互任务.有的三维交互设备,如数据手套、轨迹球以及语音输入设备等,其所能表达的三维交互任务可以用前面描述的几种基本的三维交互任务来描述.但对于笔这种交互设备而言,笔画(stroke)是用户所产生的基本交互信息,空间中的笔画同样带有深度和三维方向的信息.显然,空间中的一笔是无法用前面所述的基本交互任务来表达的,它也应作为一种基本的交互任务.当前,随着计算机硬件技术的不断发展,交互设备层出不穷.我们这里不可能完全分析所有交互设备所产生的交互信息,但对于 3D/VR 应用而言,我们可以分析各种主流的输入设备,键盘、鼠标、数据手套、轨迹球、语音输入设备笔等,从中归纳出较为完整的基本交互任务.由此,我们总结出三维交互中较为完整的 6 种基本任务:六自由度定位、选择、释放、字符交互、笔划交互和数量值交互.这 6 种基本交互任务用来表示在三维交互中有意义的

最小信息,它们彼此独立,构成了一个基本交互任务集.通过不同的基本交互任务组合成复杂的交互任务,最终可以构成三维交互中所需要的交互任务.

接着我们来定义三维交互原语.三维交互原语与基本三维交互任务相对应,它代表了用户通过交互设备到计算机的一个词法输入,表示一个独立的、最小的、不可分割的操作.通过三维交互原语的分析 and 整合可以得出 3D/VR 交互系统中常用的交互任务.交互原语可以定义为一个四元组: $IP(TASK, PARA, TEMP, DEVICE)$. $TASK$ 用来表示 IP 所对应的基本交互任务(比如,是六自由度定位原语还是字符原语), $PARA$ 用来表示原语的参数, $TEMP$ 为原语的产生时间, $DEVICE$ 是对用于产生原语的设备描述.虽然同一类型的交互任务可以用不同的交互设备产生,但我们这里定义的三维交互原语带有输入设备的信息.在传统的虚拟设备的概念中,交互原语的形成是独立于交互设备的,不同的交互设备可以产生同一种交互原语.然而当界面和交互设备复杂时,用来描述交互设备的信息将对整个界面的构造产生重要的影响.比如,在包含有笔输入设备的桌面环境下,如何区分和处理笔和鼠标的指点和勾画交互任务将是一个非常重要的问题.而在虚拟环境下,多种交互设备的参与将使这个问题变得更加复杂.这里采用 Machinlay 所定义的描述方式,它是一个六元组^[9]: $\langle M, In, S, R, Out, W \rangle$. M 用来表示输入设备的可操作属性, In 用来表示输入设备所感知的用户输入领域的范围, S 是设备当前所处的状态, R 是用来将用户的输入信息转换为设备的输出值的函数, Out 用来表示输入设备输出信息的范围, W 用来表示设备的工作属性.

以下给出了 6 大类交互原语的描述,其中 $Device$ 参数的分类和描述参照 Machinlay 的分类方法.

六自由度定位交互原语: $\langle Locate, \langle x, y, z, a, b, c \rangle, ms, \langle M, In, S, R, Out, W \rangle \rangle$.

选择原语: $\langle Select, \langle objectName \rangle, ms, \langle M, In, S, R, Out, W \rangle \rangle$.

释放原语: $\langle Release, \langle objectName \rangle, ms, \langle M, In, S, R, Out, W \rangle \rangle$.

字符交互原语: $\langle Text, \langle textContent \rangle, ms, \langle M, In, S, R, Out, W \rangle \rangle$.

笔画交互原语: $\langle Stroke, \langle p_1, p_2, \dots, p_n \rangle, ms, \langle M, In, S, R, Out, W \rangle \rangle$.

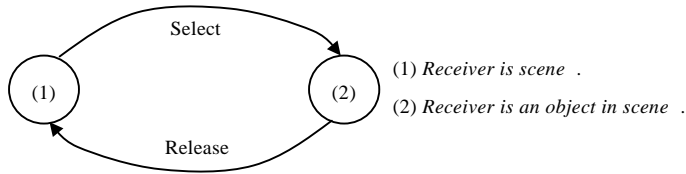
数量值交互原语: $\langle Quantify, \langle value \rangle, ms, \langle M, In, S, R, Out, W \rangle \rangle$.

其中 ms 表示用毫秒级时间单位来表示交互原语的时序.

另一方面,我们采用对目前各类主要的 3D/VR 应用中所需要的交互任务进行分析的方法来归纳出普遍的三维交互任务.由于 3D/VR 应用种类繁多,我们通过分层的方法来描述常见的 3D/VR 应用.这样归纳出来的普遍交互任务虽然无法概括所有的 3D/VR 应用所需求的三维交互任务,但它们包含了常见的 3D/VR 应用中所需的三维交互.根据分层的方法,我们将 3D/VR 应用分为 3 层:底层为三维建模、CAD 应用,中间层为场景仿真、三维可视化应用,高层为虚拟环境下的游戏、虚拟现实应用.我们对这些类型的三维交互应用进行分析,从应用的任务需求角度归纳出 3D/VR 应用中较为完整的普遍任务集.其中包括 6 种普遍的交互任务:场景浏览、实体选择、实体释放、实体操纵、场景数据访问和实体数据访问.

在归纳了基本三维交互任务和普遍三维交互任务之后,我们来描述二者之间的关系.3DUI 等新一代的用户界面同传统的 GUI 相比,一个主要的特点就是新一代的用户界面支持对连续的输入信息的描述和处理^[10].在 3DUI 中有两个连续的普遍交互任务:场景浏览和对象操纵. Jacob 采用基于限制的数据流图和状态图相结合的方法对这两种连续任务进行了描述^[11].但其描述是在假定不同的交互设备所产生的六自由度信息分别传给场景和场景中的实体时对任务离散的描述,并没有从三维交互全局的角度来考虑(比如,六自由度定位信息如果传给场景将变为视点的定位,如果传给场景中的实体对象将变为对实体的定位).我们可以根据前面所定义的交互原语,利用基于限制的数据流图和状态图相结合的方法对整个三维交互任务框架进行描述.

在传统的 GUI 中,用户的交互信息传给了屏幕上的各种交互组件和应用对象(Widget 和交互式图形对象).而在 3D/VR 交互系统中,用户的交互信息不仅可以传给场景中的交互组件和三维实体(3D Widget 和交互式图形对象),还可以直接传给场景本身.在 3DUI 中如何确定用户的交互信息是传给场景还是场景中的实体是非常重要的,是进行任务整合的基础.在总结出基本的交互任务之后,我们通过状态图来描述在交互过程中交互信息的接收者的变化情况,如图 1 所示.



场景接收交互信息, 场景中的实体接收交互信息.

Fig.1 STD of interaction information receiver
图 1 描述交互信息的接收者变化的状态图

通过基于限制的数据流图和状态图的结合,我们将三维交互中的连续交互任务做全局的描述.我们首先假定 6 种基本的交互任务所对应的交互原语都可以通过产生连续输入值的设备发送给系统.可以看出,图 2 的上半部分是用来描述 6 大类交互原语(通过原语的 TASK 元素来分类)的连续输入的基于限制的数据流图(Select 和 Release 只能是离散的交互原语.在这里,我们将它作为连续输入的一个特例,与其他交互原语放在一起,以便进行整体的描述).下半部分是用来控制限制值的状态图.如果此时通过输入设备产生的是六自由度定位类的交互原语,当 SCENE 状态为真时,限制 *primitivetoscene* 为真,Locate 类原语传给了场景对象,此时的普遍交互任务为 Navigation.当 OBJECT 状态为真时,限制 *primitivetoobject* 为真,Locate 类原语传给了场景中的实体,此时的普遍交互任务变成了 Manipulation.Text,Quantify,Stroke 这三类基本的交互原语通过限制可以转化为场景数据访问和对象数据访问这两种普遍的交互任务.

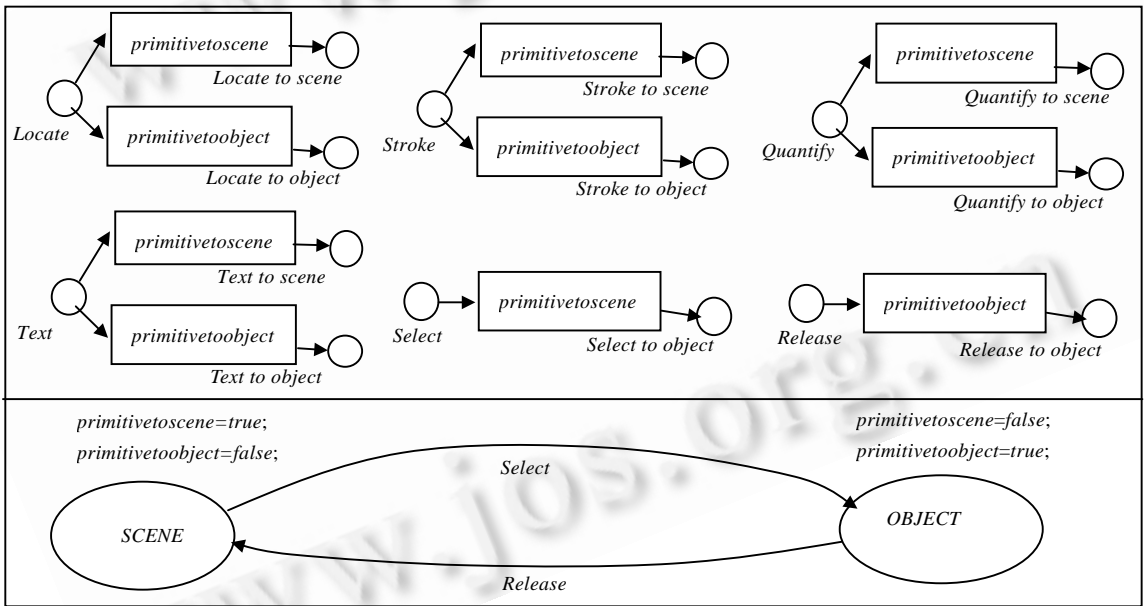
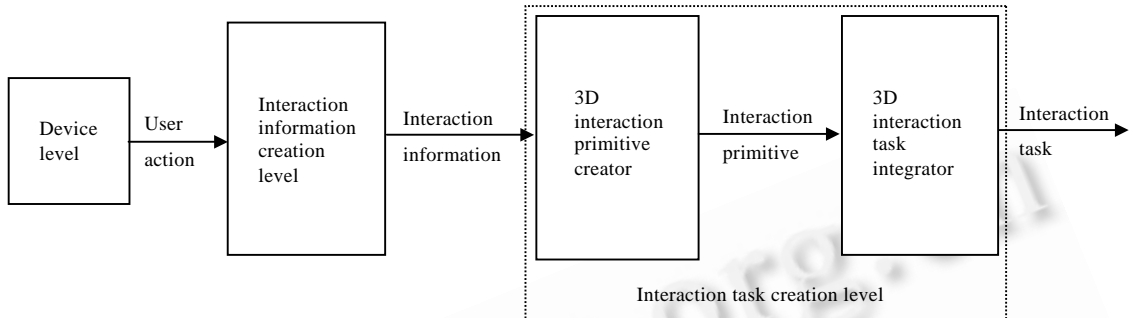


Fig.2 Specification of 3D interaction tasks
图 2 三维交互任务的描述

状态的变换需要在状态 SCENE 下接收 Select 原语或 OBJECT 下接收 Release 原语来实现.对于其他类型的交互原语,都是通过 *primitivetoscene* 和 *primitivetoobject* 这两个限制进行控制的.此外需要注意的是,Select 类型的交互原语在 *primitivetoscene* 为真的情况下传给场景中的实体,而在 *primitivetoobject* 为真的情况下并不传给场景或场景中的实体;Release 类型的交互原语在 *primitivetoobject* 为真的情况下传给场景中处于被拾取状态的实体,而在 *primitivetoscene* 为真的情况下,场景和场景中的实体并不接收.图 2 进行了三维交互的中连续交互任务的描述.对于传统的离散型交互任务而言,可以作为连续交互任务的特殊情况(在某个时间段内的连续交互任务),用图 2 同样可以进行描述.

2 三维交互任务的产生框架及三维交互核心模块

通过对三维交互一般形式的分析和描述,我们可以总结、归纳出一个通用的三维交互任务的产生框架,如图3所示.产生框架主要分为3个层次:硬件层、交互信息产生层和交互任务构造层.



硬件层, 交互动作, 交互信息产生层, 交互信息, 3D 交互原语构造器, 交互原语, 3D 交互任务整合器, 交互任务, 交互任务构造层.

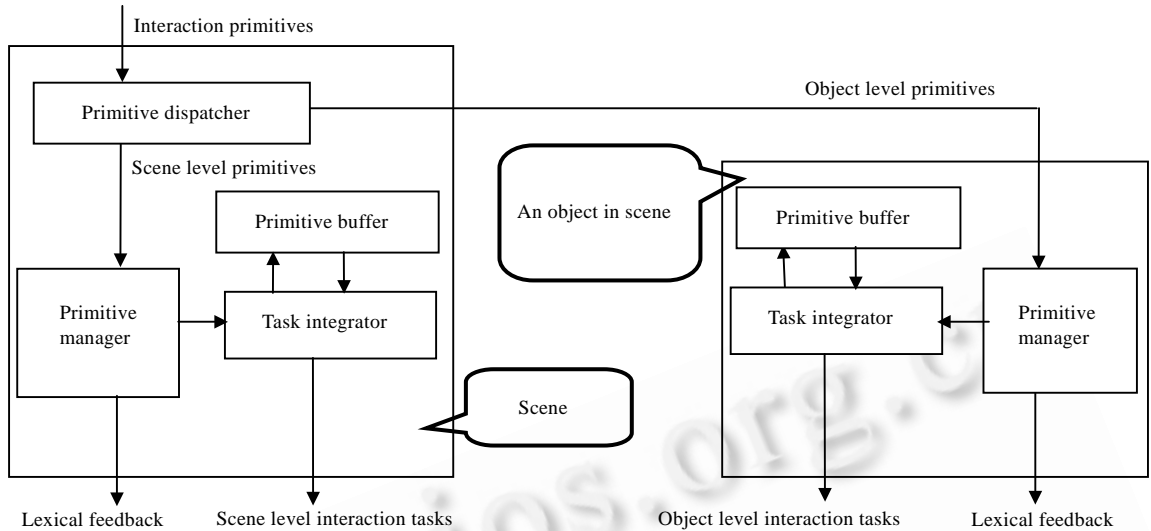
Fig.3 Framework of 3D interaction task creation

图3 三维交互任务的产生框架

硬件层由用户和交互设备组成.用户为了达到交互的目的,通过三维交互设备产生了一定的交互动作.这些交互动作是依赖于所使用的交互设备的,交互设备不同决定了交互动作的不同.比如,为了执行对象的选择这样的交互任务,用户可以通过鼠标产生一个按键的动作,也可以通过数据手套产生一个抓取的动作.同时,用户的交互动作和硬件设备的选择也与所采用的交互技术相关.在不同的交互技术中,同一种交互设备在执行同一个交互任务时可以产生不同的交互动作.交互信息产生层用来捕捉交互动作的执行过程中所产生的交互信息,并将其转化为计算机可表示的数据.所产生的数据是与硬件设备相关的.在执行同一交互任务时,用户使用不同的交互设备产生了不同的交互动作,相应地转化为不同表示形式的交互信息,但是这些交互信息所表示的内容是一致的.交互任务构造层用来将多种多样的交互信息构造成 3D/VR 应用中的普遍交互任务.它的功能是由三维交互核心模块来完成的.

三维交互核心模块是三维交互中通用的模块,与所使用的交互设备以及所采用的交互技术无关.它包括 3D 交互原语构造器和 3D 交互任务整合器两个部分.3D 交互原语构造器定义了基本的 3D 交互原语,实现了交互信息向交互原语的转换.由于交互信息是与硬件相关的,随着硬件的不断发展,所产生的交互信息也在不断变化.因此,我们将 3D 交互原语构造器设计为可扩充的结构.在硬件和交互信息增加时,我们可以产生新的构造子模块来分析这些新的交互信息,产生交互原语,而交互原语是不会发生变化的.3D 交互任务整合器定义了交互原语向交互任务的整合,交互任务的整合过程如图 4 所示.我们首先将 3D/VR 应用的普遍交互任务分为场景级任务和实体级任务两大类(分别指由场景或场景中的三维实体接收交互原语所产生的交互任务).场景级任务包括:场景浏览、实体选择、实体释放和场景数据访问;实体级任务包括实体操纵和实体数据访问.由此我们将任务的整合分为两大部分:场景级任务的整合和实体级任务的整合.我们把整合过程分别放在场景对象和场景中的某个实体对象中进行.由 3D 交互原语产生器所产生的交互原语首先发给场景对象.通过场景对象中的原语分发装置进行判断,根据判断结果来决定是由场景对象接收原语还是发送给场景中的实体对象.原语分发装置中的分析模块是根据图 2 的描述进行构造的.在场景对象和实体对象中的原语控制装置用来控制任务的整合,同时进行相应的词法反馈.任务整合装置可以进行离散交互任务和连续交互任务的整合.原语缓存装置用来临时存储交互原语以便进行组合交互任务的整合.

三维交互核心模块构成了 3D/VR 应用中的交互核心部分.程序员在开发 3DUI 构造系统(3DUI toolkit 或 3D UIMS)或具体的 3D/VR 应用时,可以直接用它来建立系统的底层,而程序员的工作将只关注于 3DUI 的交互技术的选择,交互风格的建立等高层活动.



交互原语, 原语分发装置, 场景级原语, 实体级原语, 原语控制装置, 原语缓存装置, 任务整合装置, 词法反馈, 场景级交互任务, 实体级交互任务, 场景对象, 实体对象.

Fig.4 Process of task integration

图4 任务整合过程

3 结束语

本文讨论的内容是三维交互方面的研究的基础工作,也是要构造一个自然、和谐、实用的三维用户界面首先必须解决的部分.本文对基本三维交互原语和普遍三维交互原语进行了归纳和描述,并以此为依据提出了三维交互产生框架和三维交互核心模块.三维交互核心模块作为三维交互中最为核心和抽象的部分,描述了三维交互中通用的低层结构.以本文所描述的工作为基础,在三维核心交互模块上,我们可以进行各种三维交互技术和三维交互风格的研究和评估.

致谢 中国科学院软件研究所人机交互技术与智能信息处理实验室的栾尚敏博士后对本文的完成提出了很多有益的建议,在此表示感谢.

References:

- [1] Koike, H. The role of another spatial dimension in software visualization. *ACM Transactions on Information System*, 1993,11(3): 266~286.
- [2] Myers, B., Hudson, S.E., Pausch, R. Past, present and future of user interface software tools. *ACM Transactions on Computer-Human Interaction*, 2000,7(1):3~28.
- [3] Green, M., Shaw, C.D. Virtual reality and highly interactive three dimensional user interfaces: a technical outline. *Technique Report*, Alberta: Department of Computing Science, University of Alberta Edmonton, 1999.
- [4] Foley, J.D., Van Dam, A., Feiner, S.K., *et al.* *Computer Graphics: principles and practice*. 2nd ed., Reading, MA: Addison-Wesley, 1990.
- [5] Bowman, D.A., Kruijff, E., LaViola, J.J., *et al.* Mine, Ivan Poupyrev. 3D user interface design: fundamental techniques, theory, and practice. In: Akeley, K., ed. *SIGGRAPH 2000 Course Notes 36*. Addison Wesley, 2000. <http://www.siggraph.org>.
- [6] *Computer graphics—graphical kernel system for three dimensions (GKS-3D) functional description*. ISO/IEC 8805, New York: American National Standards Institute, 1988.
- [7] *Computer graphics and image processing—programmer's hierarchical interactive graphics system (PHIGS), Part 1: functional description*. ISO/IEC 9592-1, New York: American National Standards Institute, 1995.

- [8] Pierce, J.S., Conway, M., Van Dantzich, M., *et al.* Toolspaces and glances: storing, accessing, and retrieving objects in 3D desktop applications. In: Hodgins, J., Foley, J., eds. *Proceedings of the Interactive 3D Graphics*. New York: ACM Press, 1999, 163~168.
- [9] Mackinlay, J.D., Card, S.K., Robertson, G.G. A semantic analysis of the design space of input devices. *Human Computer Interaction*, 1990,5(2-3):145~190.
- [10] Jacob, R.J.K., Deligiannidis, L., Morrison, S. A software model and specification language for non-wimp user interfaces. *ACM Transactions on Computer-Human Interaction*, 1999,6(1):1~46.
- [11] Jacob, R.J.K. A visual language for non-wimp user interfaces. In: Citrin, W., Burnett, M., eds. *Proceedings of the IEEE Symposium on Visual Languages*. Los Alamitos, CA: IEEE Computer Society Press, 1996. 231~238.

Specification and Structure Design of 3D Interaction Tasks*

TIAN Feng, DAI Guo-zhong, CHEN You-di, CHENG Cheng

(Laboratory of Human-Computer Interaction and Intelligent Information Processing, Institute of Software, The Chinese Academy of Sciences, Beijing 100080, China)

E-mail: tf@iel_mail.iscas.ac.cn

<http://iel.iscas.ac.cn>

Abstract: This paper aims at 3D interaction task analysis and construction, summarizes interaction tasks in 3D interaction field, presents the integration process from basic interaction tasks to universal interaction tasks in 3D/VR application, and finally constructs a core toolkit which can be widely used in 3D interaction field. This core toolkit specifies the low level structure of 3D interaction. A set of interaction primitives, interaction tasks and the process about primitive creation and task integration are defined in it. It can be used as the low level platform in 3D UI toolkit, 3D UIMS or 3D/VR application development. Therefore programmer can just pay attention to the high level work such as interaction style construction.

Key words: 3D interaction; interaction task; interaction primitive

* Received May 28, 2001; accepted July 16, 2001

Supported by the National Natural Science Foundation of China under Grant No.60033020