

基于经验感知的自适应用户界面模型

樊银亭^{1),2)} 滕东兴¹⁾ 杨海燕¹⁾ 马翠霞¹⁾ 戴国忠¹⁾ 王宏安¹⁾

¹⁾(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 100190)

²⁾(中国科学院研究生院 北京 100049)

摘 要 针对传统自适应界面缺乏自主学习用户交互历史、难以根据用户经验有效预测用户意图的现状,基于认知心理学相关理论,该文提出了基于经验感知的自适应用户界面模型,从界面静态组成元素、动态交互行为和自适应策略三个方面建立了该模型的统一描述,然后研究了自适应界面模型的实现架构、关键技术和建模方法.最后开发了一个个性化的虚拟家居定制原型系统,并进行了实验评估.实验结果表明,该模型能够根据用户交互历史和上下文环境,准确预测用户意图,实时调整界面布局和交互行为以主动地适应用户.

关键词 用户体验;自适应用户界面;上下文感知;界面模型;人机交互
中图法分类号 TP391 **DOI号**: 10.3724/SP.J.1016.2011.02211

An Adaptive User Interface Model Based on Experience Awareness

FAN Yin-Ting^{1),2)} TENG Dong-Xing¹⁾ YANG Hai-Yan¹⁾ MA Cui-Xia¹⁾

DAI Guo-Zhong¹⁾ WANG Hong-An¹⁾

¹⁾(Intelligence Engineering Laboratory, Institute of Software, Chinese Academy of Science, Beijing 100190)

²⁾(Graduate University of Chinese Academy of Sciences, Beijing 100049)

Abstract Traditional adaptive interface lacks the ability of self-learning user interaction history and can't predict user's intent effectively with their experiences. Based on the theory of cognitive psychology, this paper proposes an adaptive user interface model based on experience awareness. Firstly, the model is described by static interface elements, dynamic interactive actions as well as adaptive strategies. Secondly, we study the implementation architectures, key techniques and modeling method of adaptive interface model. We built a personal virtual furniture customizing system and verified it by experiments. The experiments demonstrate that the model can obtain the user's intent fairly well, adjust interface layout and interactive actions in real time to adapt to them.

Keywords user experiences; adaptive user interface; context-aware; interface model; human-computer interaction

1 引言

用户界面从最初的手工作业阶段,经过作业控

制语言以及命令语言阶段,发展到现在成熟和广泛普及的 GUI 阶段.每一个阶段都给计算机的使用带来了突破性进展.随着 GUI 的深入发展和普适计算的兴起,计算机硬件的日趋丰富,软件已经摆脱传统

收稿日期:2011-08-29;最终修改稿收到日期:2011-09-19. 本课题得到国家自然科学基金(60703078,61173057)、国家“九七三”重点基础研究发展规划项目基金(2011CB302205)、国家“八六三”高技术研究发展计划项目基金(2011AA040501)资助. 樊银亭,男,1975年生,博士研究生,讲师,主要研究方向为人机交互技术、自适应用户界面. E-mail: fytzcx@sohu.com. 滕东兴,男,1973年生,博士,副研究员,主要研究方向为可视分析技术、用户界面技术. 杨海燕,女,1980年生,博士,助理研究员,主要研究方向为人机交互技术、草图用户界面. 马翠霞,女,1975年生,博士,副研究员,主要研究方向为人机交互技术、草图计算、上下文感知交互技术. 戴国忠,男,1944年生,研究员,博士生导师,主要研究领域为人机交互、计算机图形学. 王宏安,男,1963年生,博士,研究员,博士生导师,主要研究领域为实时智能、用户界面.

的固定的输入输出模式,交互式系统日益多样和复杂,传统的界面由于缺乏主动满足多种用户需求的能力成为软件可用性提高的瓶颈.由于用户需求因个体特征、设备特征和环境状况的不同而千差万别,因此研究一种能处理多种上下文环境满足用户个体需要的自适应用户界面(Adaptive User Interface, AUI)^[1]显得尤为重要.现有面向不同领域的自适应用户界面应用^[2-3]对不具备专业计算机知识、也没有兴趣学习这些专业知识的最终用户^[4]来说,降低了对界面的认知差距和学习负担,使得人与机器之间的交互更加自然高效.自适应界面的研究目标是使用户更加方便灵活地使用软件,通过采用智能化的技术,使用户在使用软件时,让计算机通过学习或其他方式得到特定用户的认知方式、行为习惯、偏好等,使得用户界面适应用户的个性化特征.自适应用户界面是简化人机交互的好方法,它的目的是根据用户的个性化需求和应用环境主动裁减定制系统的显示和交互行为^[5].自适应用户界面是面向最终用户的界面的重要组成部分和具体体现,因此也是今后用户界面的发展方向和发展趋势.

从面向最终用户的角度来说,目前自适应界面的研究和应用在特定领域中的静态自适应方面(如界面定制)取得了较大进展^[6],但是对于界面动态自适应最终用户的研究仍显不足.具体表现在:

(1) 缺乏对用户的认知活动进行深入研究,难以从有利于最终用户交互的角度设计自适应界面;

(2) 当前界面自适应研究主要体现在界面静态元素的个性化呈现方面,缺乏对用户动态交互行为自适应的探索;

(3) 没有充分利用人们已经习成的交互方式和交互行为(经验),造成现有系统在学习用户体验方面缺乏主动性和智能性,从而影响了交互任务的高效执行;

(4) 缺少基于上下文环境感知的预测方法,正确感知、预测用户意图的准确率不高^[7],反而“干扰”用户任务的有序执行,降低了用户的执行效率.

基于经验感知的自适应界面的提出,为上述问题提供了一种可行的解决方案.它将成为未来最终用户界面的主要发展方向.借助用户的交互历史,它能主动学习用户已经取得的经验,结合上下文环境预测用户意图,为用户提供个性化的信息服务.

针对上述问题,本文提出一个基于用户体验感知的自适应用户界面模型(Adaptive User Interface Model Based on Experience Awareness, AUIM-BEA).首先根据用户的个性特征、用户角色、使用经

验,建立用户模型;然后在此基础上从静态界面组成、动态交互行为和自适应策略三方面对自适应界面模型进行统一描述,最后提出自适应界面的体系架构、关键技术和建模方法,并给出一个应用实例.实例表明,基于经验感知的自适应界面能根据用户交互历史和上下文环境,准确预测用户意图,实时调整界面布局和交互行为以主动适应用户.

本文第2节介绍自适应界面相关研究,并分析当前研究的不足之处,提出研究基于经验感知的自适应界面模型的必要性;第3节从认知心理学方面研究自适应用户界面的关键模型-用户模型,并做了详细描述和定义;第4节研究自适应界面模型主要组成部分,从静态组成元素、动态交互行为和自适应策略三方面对该模型进行统一描述;第5节描述了实现该模型的软件体系架构;第6节讨论了实现自适应机制的关键技术:交互动作序列模式构建和交互动作预测;第7节是提供该模型的建模过程;第8、9节给出一个自适应虚拟家居定制系统实例并进行了评价、分析;最后一节是对本文的总结和展望.

2 相关研究

自适应界面已经成为人机交互领域研究的热点,ACM CHI、UIST、IUI等相关国际会议都将自适应用户界面作为重要研究内容,AAAI早在2000年春季专门召开了题为“Adaptive User Interfaces”的研讨会.研究集中在特定领域的个性化界面定制和推荐等方面,如超媒体学习系统中的教师与学生之间的自动反馈^[8]、医疗领域中的针对不同用户(护士、医生)提供不同的界面和导航^[9]、残障人的辅助上网^[10]、市场中的个性化的客户关系^[11]等.此外,也有一些特定领域自适应界面应用评估方法^[12]的研究.从上述研究可以看出,自适应用户界面因其智能性、主动性的需求,使其相对于传统图形用户界面,更侧重在理解用户意图和预测未来行为方面.此外,自适应用户界面是普适计算环境中的重要界面形式,交互设备及应用场景的多样化给自适应用户界面的研究带来新的挑战.利用统一的模型框架解决自适应用户界面面临的上述问题成为研究的关键.

Takeo Igarashi开发了一个专门绘制3D图形的智能用户界面,系统根据用户显式提供的线索和意图,预测当前用户的目标任务并提供方案供用户选择^[13].Pu提供了一种“向前看”的原则和一组推荐策略,能够帮助用户做出更准确的方案选择^[14].香港浸会大学提出的自适应用户界面通过跟踪、识

别和记录用户在交互过程中的行为序列来识别用户的意图,属于面向用户的自适应用户界面^[15]. Yoo 等人^[16]提出一个推荐系统,该系统将用户划分成不同组,并在每组成员解决实际问题时进行协调过滤. 与传统推荐系统相比,南京大学孙正兴教授针对笔交互环境下的在线草图识别,研究了基于 SVM 的用户适应性解决方法^[17],解决了在线草图识别和用户适应性问题. Duarte 等人^[18]提出了用以指导开发多通道自适应用户界面的概念框架,并用行为矩阵来表示多通道用户界面的自适应规则. 徐礼爽等人^[19]提出了一个自适应笔式用户界面开发的概念框架,用来指导自适应笔式用户界面开发中的具体应用. 另外,也出现一些有关自适应的个性化系统,例如 Thompson 等人^[20]于 2003 年开发的个性化会话建议系统,戴姆勒-克莱斯勒研究与技术中心开发的自适应路线导航系统和自适应定位系统^[21]等. 除上述几类研究外,还有一些面向可视化的自适应用户界面研究^[22-23]. 这些研究着眼于在绘制上增强信息呈现效果,如鱼眼技术等,但缺乏对交互的有力支持.

当前,自适应用户界面研究主要还存在两方面不足:一方面,没有充分利用人们已有的与系统交互的历史经验,缺乏主动学习用户经验以动态适应用户,系统在界面自适应上呈现静态行为;另一方面,目前自适应用户界面的研究多集中在特定领域的应用,对自适应界面模型和实现框架研究不足,对于交互过程中自适应用户界面的作用机制和模型表达缺乏全面系统的理论研究,因而不能很好地指导自适应用户界面的设计与开发,限制了自适应用户界面

在更多领域的应用推广.

3 用户模型

3.1 模型描述

人机交互的过程是人 与计算机互相适应的过程,它包括两个层面的含义:(1)通过人工智能方法,使机器、环境适合于人;(2)通过最佳训练方法,使人适应于机器和环境. 后者往往会给用户造成认知和学习负担.

认知心理学家从信息加工角度把人看作信息加工系统,这种加工依赖于控制系统的当前目标. 人为了达到既定目标,通常会根据所处环境、个体认知和个性倾向性(即需要、动机和价值观)的不同,采取不同的行动. 在人、计算机、环境以及目标与行动之间形成一个人机交互空间. 从用户的角度来说,如果用户界面呈现的刺激/信息能适应用户的认知心理和个性特征,就有助于用户快速完成任务,达成目标. 从计算机系统角度来看,如果系统能不断根据来自用户的刺激/信息,为用户提供反馈,则能提高完成目标的效率. 这种交互过程在自适应用户界面系统中体现为探测-适应的过程. 首先,如果系统能够从用户的交互行为探测中获得足够的信息,就能感知用户的行为意图;其次,因为用户在与计算机交互过程中也不断学习,用户则不断提高使用系统的能力. 因此,结合这两个条件,计算机系统就可能为用户提供适应用户个性化需要的信息. 如图 1 所示.

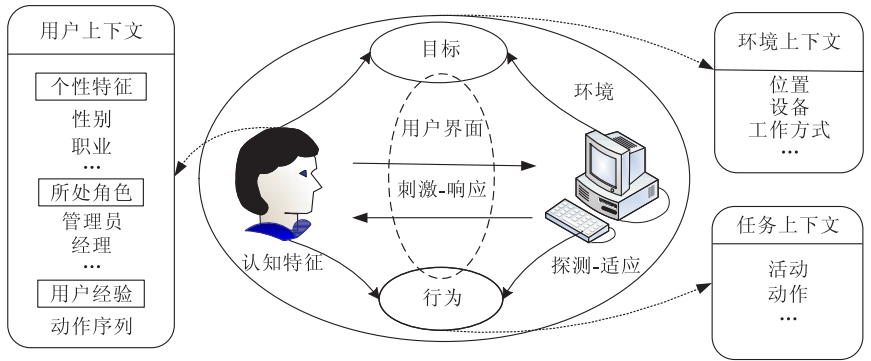


图 1 基于认知心理学的人机交互环境

为了完成特定的交互任务,用户需要在计算机中执行一系列交互动作,这些交互过程会增加用户的认知和记忆负担,而且随着用户使用系统经验的增长,这些重复执行的交互步骤也会降低用户的体验. 因此,人机交互过程中,最佳的方式是尽可能发挥计算机的计算能力,使系统智能地适应用户. 计算

机的主要任务是对根据用户的认知特征、个性偏好和使用经验等建立用户界面模型、记录用户的交互方式和交互行为,结合上下文环境分析预测用户的意图,主动为用户提供合适的界面布局和信息呈现方式,智能地为用户提供自然的交互方式.

根据心理学相关研究成果,人们在感知事件时

习惯于将连续事件依据各类特征分割为若干有意义的活动. 事件边界即为相邻活动之间的间隔,它能够辅助用户更好地理解、记忆事件活动. 利用上下文信息为交互提供辅助支持,包括三类上下文信息:用户上下文、任务上下文和交互环境上下文. 用户上下文主要用于描述、记录用户的个性特征、所处角色、用户体验等;任务上下文包括交互任务的目标、活动以及一系列用户的交互过程;环境上下文则涉及交互任务的设备、工作方式及位置等. 借助各类上下文,将传统交互中的以人为认知主体的交互策略扩展到个体与交互环境的相互作用.

3.2 模型定义

认知特征、个性偏好等个体差异使得不同用户对用户界面有不同的要求,影响用户正确理解界面的可视表达方式. 而且,对同一个系统而言,专业人员和初学者为完成同一任务所采取的行为是不同的,同一用户随着对系统的不断熟悉其行为特征也是不同的. 另外,计算机也有适应个体或群体用户的需求.

在自适应系统中,用户模型描述了用户的个性特征、工作环境中所处的角色以及用户使用系统的经验. 用户模型 (User Model, UM) 是一个四元组, $UM = \langle UserID, Personality, Role, Experience \rangle$, $UserID$ 代表用户唯一标识符, $Personality$ 表示用户个性特征, $Role$ 表示用户角色, $Experience$ 表示用户体验.

3.2.1 用户的个体特征

即使有相同的角色、相同的工作经历,不同用户也会因个体差异有不同的需求. 例如习惯使用左手的用户,对界面布局和呈现方式与习惯于右手的用户有所区别. 个体特征包括两个方面:一是个体所具有的客观属性,如性别、年龄、学历、认知能力等等;另一方面主要是用户的情感偏爱等主观属性. 催化和激发用户的感知刺激主要由用户的个体特征所决定.

$Personality = \langle Sex, Age, Education, Cognitive, Preference \dots \rangle$.

偏好 (Preference) 是人们根据环境、知识、感知和认知的结果而对动作做出的有倾向性的选择,是对系统中已有属性的选择. 为定义偏好,我们引入供给 (Affordance) 这一概念. 定义为系统为用户提供的可选属性集合,例如屏幕分辨率、字体、颜色、交互方式等. 偏好 (Preference) 可描述为 $Preference = \{Aff(i) | Aff(i) \in AffSet, 1 \leq i \leq n\}$. $Aff(i)$ 是系统供给集合 $AffSet$ 的元素, n 表示用户偏好的总数.

3.2.2 用户角色

用户角色表示群体用户可以执行的行为,角

色 (Role) 是一个三元组, $Role = \langle RoleID, RoleName, \{Action\} \rangle$. $RoleID$ 表示角色标识符, $RoleName$ 代表角色名, $Action$ 是同一角色下的许可的交互动作. 角色和动作是多对多关系,即一个角色可以拥有多个交互动作,一个交互动作也可以隶属于多个角色. 同理,用户和角色也是多对多的映射关系.

3.2.3 用户体验

用户在完成任务过程中,需要执行一系列的交互动作来达到目标,这些执行序列作为用户使用经验保存在系统中. 随着用户使用系统时间的增多,用户体验的不断增加,用户则能更快地完成任务. 反过来,如果系统能不断学习用户体验并结合上下文环境分析预测用户的交互意图,则能提供更好的自适应用户界面. 这些经验决定系统自适应用户的智能程度,影响着用户使用系统的效率和体验.

本文中,用户体验是为了达成目标而执行的一系列交互动作,每个动作表示原子的、作用在界面对象上并有作用效果的交互任务,比如对界面对象的平移、旋转和缩放等. 我们将系统中的交互动作 $Action$ 统一定义为一个六元组:

$$Action = \langle actionID, actionName, objectId, position, time, ResParam \rangle,$$

其中, $actionID$ 为该动作的标识, $actionName$ 为动作名称, $objectId$ 为动作的作用对象标识, $position$ 为动作发生位置, $time$ 为动作发生的时间, $ResParam$ 为动作效果特性参数,需根据特定动作的效果特性个性化定义. 例如,平移动作的动作效果特性参数 $ResParam = \langle fromPos, toPos \rangle$, $fromPos$ 和 $toPos$ 分别为平移对象的起始和结束位置.

因此,用户体验 $Experience$ 定义为

$$Experience = \langle Action(t) | t = t_1, t_2, \dots, t_n, \text{ and } t_1 < t_2 < \dots < t_n \rangle.$$

4 自适应用户界面模型

Puerta 提出了一种通用的界面模型,基本组成元素包括用户模型、领域模型、任务模型、表示模型和交互模型,各个组成模型从不同侧面对界面进行描述^[24]. 用户模型 (User Model) 侧重描述特定平台下不同类型的用户角色、用户特征等. 领域模型 (Domain Model) 侧重描述用户通过界面能够浏览、访问、操作的应用领域信息. 任务模型 (Task Model) 侧重描述用户通过界面所能完成的任务. 表示模型 (Presentation Model) 侧重描述界面上的可视组件. 交互模型 (Dialog Model) 侧重描述用户与界面对象

交互的过程。用户模型、领域模型和任务模型属于抽象模型,表示模型和交互模型属于具体模型。本文以上述通用界面模型的各个组成元素为基础,结合自适应界面的特点和一定的界面自适应策略,构建自适应用户界面模型。其中,用户模型、领域模型、任务模型和表示模型属于界面的静态组成部分,交互模型属于界面的动态交互行为。

自适应用户界面模型是一个六元组,可描述为 $AUI = \langle UM, DM, TM, PM, IM, APM \rangle$, 其中, UM 为用户模型, DM 为领域模型, TM 为任务模型, PM 为表示模型, IM 为交互模型, APM 为自适应策略模型。由于本文中自适应策略模型是从具体模型中直接生成,我们把自适应策略模型划归为具体模型。

4.1 界面静态组成元素

定义 1. 用户模型 UM 的定义见第 3.2 节。

定义 2. 领域模型 DM 定义为 $DM = \langle \{Entity\}, \{RL\} \rangle$, 其中, $Entity = \{Attribute_1, Attribute_2, \dots, Attribute_n\}$, $Entity$ 表示含有若干属性的业务对象; RL 表示实体之间的属性引用。

定义 3. 任务模型 TM 定义为 $TM = \langle \{action\} \rangle$, 其中, $action$ 为交互任务。

定义 4. 表示模型 PM 定义为 $PM = \langle \{VCP\} \rangle$, 其中, VCP 为可视化组件, $VCP = \langle DCP, GCP, ICP \rangle$, DCP 为数据属性, GCP 为组件可视属性, ICP 为交互属性。

4.2 界面动态交互行为

4.2.1 交互上下文

交互上下文是决定或影响人机交互过程和对话方式的各种相关信息资源,可以来自用户、设备与系统以及外部环境等实体。结合前面的分析,影响界面自适应的主要因素是当前交互的用户上下文、任务上下文和环境上下文。

定义 5. 自适应用户界面交互上下文: $Context = \langle UC, TC, EC \rangle$, 其中 UC 为用户上下文, TC 为任务上下文, EC 为环境上下文。

定义 6. 用户上下文 UC 定义为 $UC = \langle User \rangle$,

$user \in UserSet$ 其中, $User$ 为当前用户, $UserSet$ 表示为系统的所有用户集合。

定义 7. 任务上下文 TC 定义为 $TC = \langle System, type, name, des, time \rangle$, 其中, $System$ 为当前执行任务的系统状态, $type$ 为动作类型, $name$ 为动作名称, des 为动作描述, $time$ 为动作发生的时间。

定义 8. 环境上下文 EM 定义为 $EM = \langle device, worktype, location \rangle$, 其中, $device$ 为设备, $worktype$ 为工作方式, $location$ 为用户所处位置。

4.2.2 基于上下文感知的交互模型

交互是用户通过物理设备完成一系列操作,界面将操作序列转化为可执行指令并执行的过程。建立交互模型的目的是定义符合用户认知特征的交互技术,为界面设计提供指导原则。由于人的认知能力受交互过程中可得到资源的影响^[25],超过一定数量的资源会给用户带来较大的认知负担。此外,硬件设备成本、物理输入设备资源数量以及其所支持的操作原语类型的限制也会增加用户的交互负担,因此需要借助扩展逻辑空间的逻辑工具辅助交互过程,以解决复杂的交互任务问题。以笔交互为例,笔只有简单的点击、勾画、旋转等几个基本操作(交互原语),而交互任务种类繁多,仅对图形的任务操作就有平移、缩放、旋转、删除、裁剪等若干种,有的任务还相对复杂,单纯依靠笔的基本操作很难实现自然流畅的交互。因此,需要将笔输入设备根据用户意图映射成符合人们认知习惯的虚拟逻辑工具。据此,提出了基于上下文感知的交互模型。

定义 9. 交互模型 IM 定义为 $IM = \langle OptSet, LGTSet, Context, MapAlogrithm, FuseAlogrithm \rangle$ 。

用户完成复杂任务可能需要多个辅助工具。因此,某一时刻物理设备与虚拟逻辑工具并不是一一对应的映射关系,当交互动作发生时,物理设备可能被映射为多个逻辑工具。交互任务决定映射结果,用户的个性偏好和交互上下文对映射结果具有一定的影响。即相同任务在不同上下文环境中完成,逻辑工具映射结果可能不同。映射过程表示为 Map ,如图 2 所示。

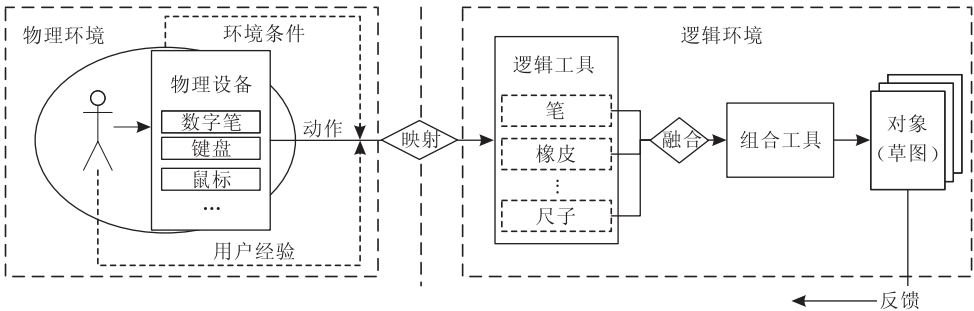


图 2 交互模型

定义 10. 映射过程 Map 定义为 $Map(Device, Context) \rightarrow \{LGT_1, LGT_2, \dots, LGT_n\}$.

由于受物理输入设备的限制,传统交互方式在同一时刻只能有一个辅助工具处于激活状态,用户需要在多个逻辑工具之间来回切换,影响了交互的流畅性.在该交互模型中,系统将映射的逻辑工具集融合为一个整体,相互配合,从而有效地减少交互中断.融合过程表示为 Fuse,如图 2 所示.

定义 11. 融合过程 Fuse 定义为 $Fuse(\{LGT_1, LGT_2, \dots, LGT_n\}) \rightarrow Set(LGT)$.

4.3 自适应策略

4.3.1 基于动作的自适应策略

自适应策略分为 3 类:(1)基于动作的自适应策略;(2)基于上下文的自适应策略;(3)基于策略的自适应策略^[26].本文所提出的自适应用户界面中的自适应策略是基于动作的,但也结合用户所在的上下文环境进行预测.本文中所有的动作都是在一定的交互上下文环境中执行的,只有在相同交互上下文中的相同类型动作才视为相同动作重现.基于动作的自适应策略就是系统通过不断记录用户的个性化信息、系统当时的状态参数和所处的环境参数等上下文信息,从而形成用户的交互历史,结合当前上下文,从以往的用户交互经历中提取用户的交互意图和兴趣,从而对用户未来要执行的动作进行预测.

用户动作的粒度可以是低层次的原子操作事件,也可以是经过封装的高层次的交互动作.低粒度的操作事件,如鼠标的单击、双击、键盘的按键事件等,能最大程度地记录下用户在系统中的操作行为.用户动作的粒度对交互历史的构建和用户的语义理解都有重要影响.交互历史采集动作的粒度太小或太大,都可能导致驱动系统状态转换的动作缺少完整语义,不易被用户理解.本文所研究的自适应动作是以用户为中心的交互任务模型中的交互动作,如对图形操作的平移、缩放、旋转等动作,对表单操作的添加、删除、保存等等.

4.3.2 自适应策略模型

用户与系统交互的过程采用时序的方式保存,形成操作的偏序集合-用户体验.我们可以根据执行操作的时间密度将用户体验聚类,从而产生用户体验 $Experience$ 的一个划分 $Experience = \{E_1, E_2, \dots, E_m\}$,每一个分块 $E_i (1 \leq i \leq m)$ 代表在相应时间内完成的操作序列,这些序列中蕴含着用户与系统的交互模式^[8].我们在子分块基础上挖掘出交互历史中的用户交互规律,从而发现用户与系统交互的一

些个性偏好,智能地为用户提供自适应界面.为了提高自适应计算效率,我们进一步分析整合用户交互历史,建立自适应策略模型.

定义 12. 自适应策略模型 APM 为一个五元组 $APM = \langle \Omega S, \Omega P, \Omega O, A, B \rangle$,其中,

$\Omega S = \{s_1, s_2, \dots, s_i, \dots, s_n\}$ 为一组系统状态的有限集合;

$\Omega P = \{p_1, p_2, \dots, p_i, \dots, p_m\}$ 为一组原子操作的有限集合;

$\Omega O = \{a_1, a_2, \dots, a_i, \dots, a_t\}$ 为一组用户操作行为的观察值的有限集合;

$A = \{a_{ij}\} = \{L(O_t = a_i | X_t = s_i)\}$ 是系统处于状态 s_i 下,操作观察值 a_i 的匹配序列的平均长度概率分布;

$B = \{b_{ij}\} = \{P(O_t = a_i | X_t = s_j)\}$ 是系统处于状态 s_j 下,操作观察值 a_i 的概率分布.

自适应交互的实现方法就是匹配从用户交互历史中挖掘出的交互模式与当前即将发生的动作.传统的方法是通过选择两个连续的动作对作为模式,计算在前一个动作发生后,下一个动作发生的条件概率,根据当前的已发生动作,找出对应最大概率的下一个动作.事实上,通过查看用户的交互历史记录可以看出用户交互模式跨越不只两个动作,这对于向前回溯多个动作的预测就显得无能为力.为了解决这个问题,我们需要改变模式匹配长度.匹配模式越长,捕捉当前的状态越好,预测的效果越好.即我们要选择匹配当前动作的模式是最长的交互序列.假设一个小整数 k ,在时刻 t 状态 s 时预测动作 a ,通过计算 $L_t(s, a)$ 取得在状态 s 时,以动作 a 结束的最大长度为 k 的序列的长度的平均值.

定义 13. 匹配序列平均长度概率 $L(a | s) = \frac{L_t(s, a)}{\sum_i L_t(s, a_i)}$.

尽管我们把匹配模式的长度看作主要预测指标,但是特定状况下,一个指定的动作可能并不是长模式的一部分,它的发生可能与另一动作密切相关,为了说明这种可能性,我们在此引入动作频率函数 $f(s, a)$ 表示在状态 s 下动作 a 发生的次数.

定义 14. 动作发生概率 $P(a | s) = \frac{f(s, a)}{\sum_i f(s, a_i)}$.

定义 15. 动作发生评价 $R_i(s, a) = \alpha L(a | s) + (1 - \alpha) P(a | s)$.

其中,用参数 $\alpha (0 \leq \alpha \leq 1)$ 作为考量两个预测指标的影响因子.表示在模式长度和动作频率上进行

折中,当 $\alpha=1$ 则只考虑匹配模式长度,当 $\alpha=0$ 时则只关注动作频率.

如图 3 所示,我们用一个短的交互历史序列为例,当前状态为 s_3 , $k=3$,匹配即将发生的动作的最长序列集合为 $\{2,1,0\}$,因此 $Lt(s_3,a_3)=(0+1+2)/3=1$.假定所有动作的总值 $\sum_i l_t(s,a_i)=5$,动作

a_3 在界面状态 s_3 时发生的次数为 30 次, s_3 状态时总共有 100 次的动作发生历史, $\alpha=0.8$.则动作 a_3 在 t 时刻界面状态 s_3 时的动作发生评价为

$$\begin{aligned} R_t(s_3,a_3) &= 0.8 \times \frac{1}{5} + 0.2 \times \frac{30}{100} \\ &= 0.16 + 0.06 = 0.22. \end{aligned}$$

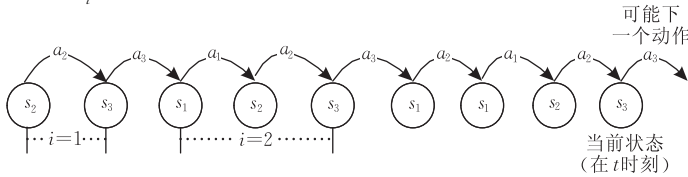


图 3 交互动作序列与状态变迁

5 自适应用户界面体系架构

根据相关研究和开发实践,本文提出自适应用户界面的体系架构,如图 4 所示.该体系结构分为 4 层:底层为数据层,中间层为模型层和自适应层,上层为界面层.数据层存放应用领域的信息.模型层由用户模型、任务模型、环境模型组成,描述自适应所需的信息集合.自适应层由自适应推理模块、自适应决策模块和交互上下文收集器组成,推理引擎通过系统事件消息处理机制捕捉事件,对事件进行分析、合并和整理,形成交互任务原子事件序列,结合上下文识别交互任务,并把识别结果传入用户模式识别模块中,进而结合知识库和规则库识别用户

的交互模式,通过意图预测模块预测用户最可能的交互行为候选;自适应决策模块基于传入的交互行为候选,结合交互历史中的系统自适应反馈评价进行自适应交互行为的决策;自适应层是实现用户界面自适应功能的算法和调度部分.界面层呈现自适应层推荐的结果.层与层之间以通信相互关联:来自系统底层的消息通过消息事件引擎形成交互历史,并为自适应引擎提供了交互上下文.自适应模块由信息收集和自适应规则来确定,它通过连续不断地收集和分析上下文信息和交互历史信息,采取一定的自适应策略,结合领域知识库和推理规则,为界面层提供自适应的功能.总体上,自适应层通过与上层的界面层和底层的模型层以动态在线迭代的方式通信,从而实现了用户界面的自适应功能.

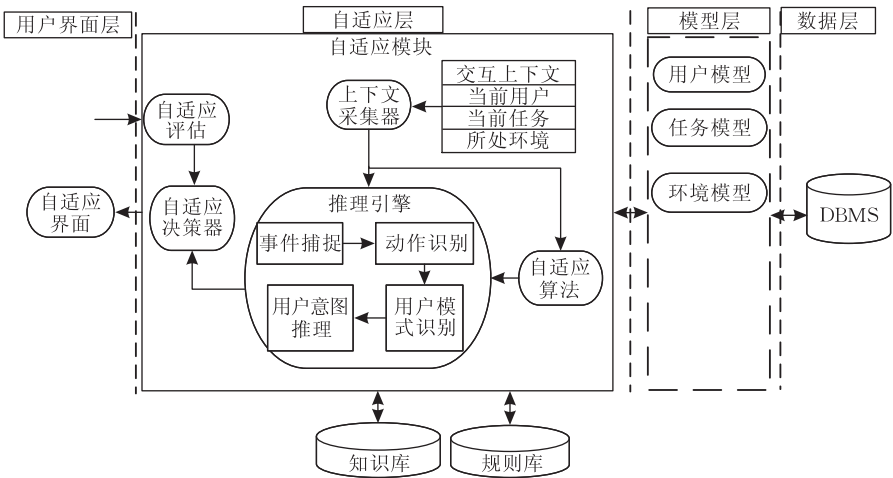


图 4 自适应用户界面体系架构

6 关键技术

基于用户经验(交互历史)的动作预测方法通过对经验动作序列进行挖掘,利用交互历史机制所记

录的用户动作序列构建用户动作模式,进而对用户将要执行的动作进行预测,为用户提供自适应界面支持,如推荐相应的动作和系统状态等,以减少用户操作负担和提高用户交互体验.基于用户经验的动作预测方法包括两部分:交互动作序列模式构建和

交互动作预测.

6.1 交互动作序列模式构建

人们在交互过程中会自觉或不自觉地遵循一定的交互方式,自适应用户界面关注的重点就是通过历史交互动作序列的研究,发现人们交互中存在的模式,从而对用户未来的行为进行预测.

为了预测用户下一步要执行的动作,系统会选取当前长度为 n 的动作序列 S_1 ,然后到用户模式中搜索,找出长度为 $n+1$ 的动作序列 S_2 ,其前 n 长度的动作子序列与 S_1 完全一致,则 S_2 的第 $n+1$ 个动作为用户可能执行的动作.构建交互动作序列模式主要算法思想:偶尔出现一次的交互动作序列为预交互模式,反复出现的交互动作序列作为交互模式加入模式库中,在最大模式长度限制下,匹配模式的序列尽可能长.假定存在一个正在执行的交互动作 a ,交互动作集合为 C ,新动作集合为 E .如果 $a \in E$,则产生一个以 a 作为结束项的交互动作序列模式,作为预交互模式;否则,若 a 不属于 C ,取以 a 为结束项,连续的若干动作与预交互模式比较,将其中相同的序列段作为以 a 为结束项的交互模式加入到模式库中,并把 a 加入到集合 C 中.如果 a 属于 C ,则到模式库查找与它局部匹配的序列.如果存在,则加入到模式库中;如果不存在,则预示着这段序列可能蕴含一个新交互模式.具体算法如算法 1.

算法 1. 交互动作序列构建算法.

输入: 用户当前的交互历史序列和交互模式库

输出: 用户交互动作序列模式

算法过程:

1. 如果 $a = \text{NULL}$,算法结束.否则,若 $a \in E$,则转向步 2;否则,若 $a \in C$,转向步 6;否则转向步 3.
2. 设交互模式库中的最大交互模式长度为 N ,在交互上下文历史记录集合中,取以当前动作为结束项、长度为 N 的连续序列,作为预交互模式 $preS$, a 从集合 E 中取出,转向步 5.
3. 取当前以 a 为结束项、长度为 N 的连续序列,与预交互模式 $preS$ 比较,其中相同的序列段为以 a 为结束的交互模式 S .
4. 把 S 加入到模式库中,更新 S 序列模式中各个动作的发生评价 R_i .
5. 继续下一个动作结点,转向步 1.
6. 对于以 a 为结束项的历史记录序列,如果在交互模式库中没有与它局部匹配的模式,则预示这段记录序列中可能存在一个新模式,转向步 2. 否则,产生一个新的交互模式 S ,转向步 4.

下面对这个算法的时间复杂度和空间复杂度进行分析.首先我们假定一个指定系统交互历史在线跟踪机制所捕获的交互动作的类型数为 m ,当前交

互动作序列的长度为 n .算法在每次交互动作发生时,将新的交互动作加入到交互动作序列中,形成交互动作序列模式,假设新交互动作所在序列的总长度为 p 的话,则 p 的最大值为交互动作的类型数为 m .因此,算法的时间复杂度为 $O(m)$.交互动作序列模式采用 Tried 树数据结构存储,Tried 树的最大深度为交互动作序列的总长度 n .Tried 树每个结点的最多子树数为 m ,Tried 结点最多是每个交互动作子序列在 Tried 树只出现一次.因此,Tried 树中的最大结点数为当前交互动作子序列数.在动作序列模式构建过程中,将长度为 n 的交互动作序列构建为交互历史模型,最简单的情况是交互动作序列为一个线性序列,此时当前交互动作子序列数为 $n(n+1)/2$.因此,交互动作序列模式构建算法的空间复杂度为 $O(n^2)$.

6.2 交互动作预测

在构建用户动作模式基础上,对用户下一步要执行的动作进行预测.预测用户下一步动作的方法有多种.例如,可以找出交互上下文下一个最大可能性动作,也可以找出最大长度匹配序列的动作预测等等.本文采用综合计算最大匹配长度和最大发生概率的折中方法.算法思想是:与当前动作序列匹配长度和待发生动作发生概率最大的序列,对应的待发生动作即为预测的动作.在当前交互上下文环境下,已有动作序列从当前动作开始逐步向后递增序列长度,和模式库中的模式匹配,取得所有预测动作候选集合 C .每个候选预测动作在每次序列匹配时,动作发生评价 R_i 取最大时的匹配序列长度.最后从候选集合中取出最大 R_i 的预测动作作为预测结果.为了提高预测效果,减少预测错误,规定一个动作发生评价阈值 R_{th} .若 R_i 大于 R_{th} ,则预测有效,否则不作预测.具体算法见算法 2.

算法 2. 交互动作预测算法.

输入: 当前交互动作序列

输出: 预测的交互动作

算法过程:

1. 取长度 L 为 1 的当前交互动作序列 S ,使用此序列在交互模式库中查找,计算所有匹配序列中的下一个动作发生评价 R ,计算得到交互动作初始预测集 $C = \{ \langle C_i, R_i, i \rangle \mid i \in [1, L] \}$.
2. 假定最大模式长度为 N , $M = \min(L, N)$,如果 $L > M$,转向步 4,否则转向步 3.
3. 当前交互动作序列长度 $L = L + 1$,沿着当前交互动作序列反向读取上一个动作,形成新的长度为 L 的当前交互动作序列.依次取出预测集 C 中交互动作 C_i ,置在长度为 L 的当前交互动作序列后,形成新的长度为 $L + 1$ 的动作序

列 S_i . 在模式库中查找, 计算所有 C_i 的动作发生评价 R'_i . 如果 $\exists i(R'_i > R_i \wedge i \in [1, L])$, 则更新 C 集合对应的动作评价和序列长度, 转向步 2.

4. 取出 C 集合中最大 R_{\max} , 及对应预测动作 C_{\max} , 如果 $R_{\max} < R_{th}$, 则 $C_{\max} = \text{NULL}$, 表示不作预测.

5. 最后的 C_{\max} 为最终预测的交互动作, 并更新动作序列模式库和信息库, 算法结束.

交互动作预测算法采用当前交互动作序列从当前交互动作逐步回溯增长匹配长度, 在交互动作序列模式库中查找, 并将得到的候选预测动作集合逐步评价后取最佳的预测动作. 算法的执行次数与最大模式长度和模式库长度成正比. 设定交互动作序列模式最大长度为 N , 交互动作序列模式数为 m , 因最大模式长度不变, 逐步递增匹配长度计算数量级为 $O(1)$, 而系统采用每次到序列模式库中查找, 则每次模式查找时间复杂度为 $O(m)$, 因此交互动作预测算法时间复杂度为 $O(m)$.

7 建模方法

针对上述自适应用户界面模型, 本节重点介绍该模型的构建方法和步骤. 自适应界面模型建模方法是从抽象模型到具体模型逐步构建的过程. 用户模型是建立一个良好的自适应界面模型的基础. 以此模型为出发点, 提取领域内的信息, 形成领域模型; 提取领域内的管理任务, 形成任务模型; 结合具体交互设备, 依据领域模型元素和任务模型中的命令集合, 设计平台设备上的可视组件对象, 形成表示模型; 根据任务模型中的任务功能集合, 设计合适的交互原语和命令集合, 形成交互模型; 自适应策略模型是系统从表示模型和交互模型中自动构建的, 如图 5 所示.

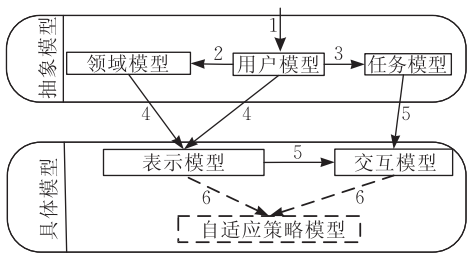


图 5 AUIMBEA 的建模过程

具体过程如下:

(1) 建立良好的用户模型

第 1 步是明确用户的个性特征(Personality), 对 *Personality* 给出定义并进行描述. 即用户习惯左手还是右手操作; 用户对色彩、字体等的偏好. 用户偏好可以为设计表示模型对象提供帮助, 例如对象

的布局、颜色、大小等.

第 2 步明确用户角色, 对用户所在的应用中所处角色进行分类描述. 即用户是管理员, 还是超级用户或者其它角色. 按照角色不同分配可执行的交互任务、可查看的领域信息和改写的领域信息等.

第 3 步, 因用户体验是系统自动记录交互历史, 不需要手动建模.

(2) 从用户模型到领域模型

设计人员在建立领域模型时, 需要从用户模型出发, 列举出待管理的实体对象, 分析各实体对象的属性, 然后将若干相关属性组织在一起整理化简, 定义为实体(Entity), 并把一个实体的属性来源于另一个实体的属性定为关系(RL), 选择合适的名称进行命名, 直到所有的领域信息用实体联系模型描述完成.

例如虚拟家居定制系统中的家居管理, 家居实体由家居序号、名称、类别、2D 示意图、3D 示意图、价格和说明等属性组成, 家居类别实体是由类别号、类别名称和父类别等属性组成, 两个实体之间存在一个关系, 即家居实体中的类别来源于家居类别实体中的类别号属性.

(3) 从用户模型到任务模型

用户模型中的角色已定义了该角色下可执行的任务. 在设计任务模型时, 每个交互动作对应一个任务. 例如界面的切换、可视对象的创建、数据的保存等. 在建立任务模型时, 根据上下文信息, 定义用户需要完成的交互任务, 定义功能集.

以 PC 机上用笔装饰衣柜侧面的任务为例. 用户模型中对该场景的描述为

$MS = \langle (Goal_1, Goal_2), (E_1, E_2) \rangle$, $Goal_1 = \langle PC, IC_1, Pen \rangle$, 其中 $IC_1 = \langle Tools, OpTime, Decorating \rangle$; $Goal_2 = \langle PC, IC_2, Pen \rangle$. 其中 $IC_2 = \langle Bureau, OpTime, Decorated \rangle$, $(E_1, E_2) = (Tap, Tap)$.

对应的任务模型中的任务定义为

$Task = \langle AT_1, AT_2 \rangle$, $AT_1 = \langle Tap, Tools, R, IC_0, Select, IC_1 \rangle$, $IC_0 = \langle Tools, OpTime, Null \rangle$, 其中, $R = \langle R_{bureau}, R_{texture} \rangle$, IC_1 和 IC_2 定义同上. $AT_2 = \langle Tap, Bureau, IC_1, Decorate, IC_2 \rangle$. *Select* 与 *Decorate* 是抽象的功能描述.

(4) 从用户模型、领域模型到表示模型

设计表示模型首先需要明确界面中的可视组件对象, 包括交互对象和应用对象, 根据用户模型可显示的领域模型元素选择应用对象, 根据用户可实现的交互任务选择交互对象. 进而明确对象属性, 根据用户模型中的用户偏好, 设计应用对象、交互对象的

外观属性,例如用户对色彩的选择、字体的要求、界面的布局等。

(5) 从任务模型和表示模型到交互模型

交互模型的构建主要是对交互上下文环境进行描述:包括交互设备、交互原语、命令、应用反馈以及系统反馈。具体来说,就是根据任务模型中每个任务的事件集合,描述在用户界面中的要执行任务的原子操作(如单击、双击等等);结合上下文,描述出对应表示模型中交互对象的命令(如激活、删除);根据任务模型中任务上下文描述应用反馈;根据系统需要描述系统反馈。

(6) 从表示模型和交互模型到策略模型

策略模型是系统自动构建的。系统结合上下文环境,自动将来自交互模型的交互原语信息构建成策略模型中的原子操作事件,执行交互任务后自动记录系统所处的状态等等。

8 应用实例

我们采用 VC++ 开发工具和 Direct3D 渲染引擎,利用 SQL Server 2005 数据库开发了一个家居设计定制系统 Decorator1.0。该系统定位在房地产售楼、室内装修、家居销售等行业,以直观、快速、自然地实现家居装修方案的设计为主要目标,通过笔交互、三维造型、三维真实感图形渲染和实时漫游等技术,使客户能很容易地参与到自己家居的设计过程中。面向虚拟家居设计展示系统主要分为两大模块。第一部分是针对三维可见有形体的操作,这类交互任务在符合客观世界物理规律的虚拟环境中完成。图 6 为演示实例。



图 6 自适应界面实例

第二部分是对各种非有形信息的操作与处理,即对海量的虚拟家居设计方案的数据信息的操作与处理。该部分针对最终需要完成信息洞察、综合分析得出结论等任务的用户,包括可视组件分析界面和

交互历史模块两大模块。可视组件分析界面展示了配置好的可视分析组件集,并提供了丰富的可视组件及可视组件之间的交互操作,主要包括图元操作、数值操作、视图操作和内容操作。交互历史模块实现了交互历史在线跟踪机制,监测用户的交互操作,构建用户的交互历史库,并提供给用户交互历史导航界面及历史知识注释界面,同时实现了基于交互历史的自适应布局机制和基于动作预测方法的可视分析状态推荐界面。交互历史在线跟踪机制监测用户的交互动作,添加可视分析状态到交互历史库中,并同步更新交互历史可视化界面。交互历史可视化导航界面允许用户做任意的回溯操作,回溯到任意的先前分析状态,并通过注释操作添加中间的分析结论和分析思路。自适应布局则融合在可视组件分析界面中,设计了聚焦布局工具以便于完成自适应布局。可视分析状态推荐融合在交互历史导航界面中,通过预测状态工具可以弹出可视分析状态推荐界面。图 7 为经过自适应布局的可视分析界面,图 8 为用户经验可视化界面。

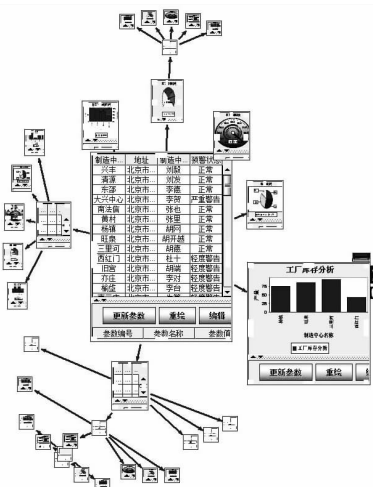


图 7 自适应布局后的可视化界面

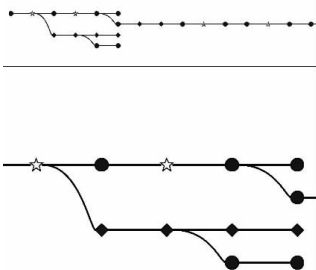


图 8 用户体验可视化界面

下面以复用三维装饰室内家居方案为例,分析基于用户经验的自适应界面的推荐过程。在家居定制的实践中发现,选用同类家具的用户希望推荐最常用的室内环境装饰风格,也就是说一定类型的家

具与一定的室内环境存在一定的关联性. 因此, 根据所选家具选用多数用户共同喜好风格的装饰方案提供给当前用户, 是一项常见的交互活动. 为客户推荐常用装饰方案的交互过程往往相同, 一般过程如下: 选中室内家具, 过滤已有方案, 点击最常用方案, 放大方案效果图, 确认推荐方案. 如用数字笔为用户 Tom 提供客厅装饰过程, 其动作序列历史描述为 $\{ \langle 1, 'Tom', Sofa, Select, time_1, null \rangle, \langle 2, 'Tom', Sofa, Search, time_2, schemeList1 \rangle, \langle 3, 'Tom', scheme1, Select, time_3, targetGraph1 \rangle, \langle 4, 'Tom', targetGraph1, Resize, time_4, null \rangle, \langle 5, 'Tom', scheme1, Select, time_5, null \rangle \}$. 后 4 个交互动作作为序列常常紧随第一个交互动作之后, 具有自适应的价值. 系统把整个交互动作序列作为潜在模式记录下来. 当再次发现该序列模式时, 就不断提高该模式的自适应评价指标. 经过一段时间后, 当评价达到一定的阈值时, 再次选中家具后, 就直接提供最常用方案的推荐结果, 从而大大减少用户的交互工作量. 由上面例子分析可知, 系统在学习用户的交互历史经验后, 能根据所处上下文环境准确预测用户的意图, 智能地提供用户自适应界面, 从而减少用户的交互步骤, 提高用户的交互效率和体验.

9 实例评估

为了验证本文研究结果, 本文针对 Decorator1.0 设计了一组实验, 目标在于考察基于经验感知的自适应用户界面的易用性和易学性. 在实验过程中, 选择了六位设计人员作为被试, 其中 3 名被试具有较高的计算机操作水平(用 A、B、C 表示), 另外 3 名被试(用 D、E、F 表示)只具备一般的计算机操作能力, 属于非专业用户. 实验任务要求每一位被试利用 Decorator 原型系统在规定时间内先后完成两个简单方案的设计(用 T1 和 T2 表示). 用工具 DesignMyRoom 重复做一个相同的方案(用 T1' 表示). 在实验开始阶段, 首先要求被试人员花费 15~20 min 熟悉 Decorator 和 DesignMyRoom 系统与任务描述. 在实验过程中, 记录被试人员每个子任务的操作完成时间及总任务的完成时间; 根据交互动作历史记录, 标注所有回溯操作, 并统计总任务的回溯总次数; 统计用户在实验过程中的操作失误总次数, 如表 1 所示. 实验结束, 每一个被试者完成问卷调查, 问卷包括 6 个评估维度, 评分范围为 1 分~10 分. 评估结果如图 9 所示.

表 1 实验数据

Testee	Task	Time spent/min	BackTracking times	Error times
A	T1	25	4	6
	T2	21	2	3
	T1'	32	7	11
B	T1	21	3	2
	T2	19	1	1
	T1'	28	9	5
C	T1	27	6	4
	T2	21	3	2
	T1'	32	8	7
D	T1	34	10	8
	T2	30	5	4
	T1'	40	13	13
E	T1	30	8	7
	T2	27	3	5
	T1'	36	12	8
F	T1	40	13	11
	T2	32	9	8
	T1'	43	16	15

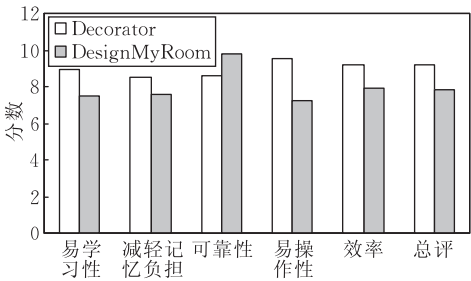


图 9 评估结果

由表 1 可以看出, 专业人员做第一方案时通常需花费 24.7 min, 在做第二个方案时花费时间可缩短至 20.3 min; 非专业用户在做第一方案时需花费较长的时间, 平均都在 35 min 以上, 做第二个方案时, 花费时间大大减少到 29.7 min; 而且对所有被试者而言做第二个方案时, 相比第一个方案, 回溯次数由平均 7.3 次降至 4.6 次, 错误次数由平均 6.3 次降至 3.8 次; 当重新用 DesignMyRoom 做一个相同方案时, 所有被试者的平均花费时间高达 35 min, 回溯次数平均达 10.8 次, 错误次数平均达 9.7 次. 图 9 表示除了可靠性外, 其它各项指标都比用传统设计较高, 且有几个指标都明显高于传统界面设计. 由此得出结论: (1) 用户在熟悉系统一段时间后, 不管是专业用户还是非专业用户, 完成任务所需花费的时间有所减少, 尤其是非专业用户所花时间减少更显著, 表明该系统能够主动学习用户经验, 更智能地适应用户, 交互过程更加自然流畅; (2) 交互历史库的构建对用户交互任务的执行具有有效的辅助作用. 基于交互历史分析的自适应虚拟家居定制系统能够减少用户的交互操作, 改善家居设计活动的用户体验, 并在一定程度上提高了工作效率. 另外, 系统的可靠性还有待在未来进一步提高.

10 结 论

本文研究基于认知理论的用户模型. 在此基础上, 提出了一个基于经验感知的自适应用户界面模型, 从静态组成元素、动态交互行为和界面自适应策略三个方面对自适应界面进行分析和描述. 然后提供了该模型的实现架构以及关键实现技术, 给出了自适应界面模型的建模方法. 最后, 以虚拟家居定制为背景, 给出一个自适应界面原型及实验评价. 原型及评价表明, 本文提出的自适应用户界面模型与传统自适应界面模型相比, 具有以下优点: (1) 关注用户的交互历史经验, 为自适应界面自主学习用户经验和智能地适应用户奠定基础; (2) 结合用户交互上下文, 为用户提供更自然流畅的交互方式和交互次序; (3) 基于用户经验的交互动作预测方法能够从用户经历中发掘用户的交互动作模式, 有助于提高界面自适应效果, 减少用户认知负担和交互行为; (4) 注重序列匹配长度和动作发生频率相结合的自适应策略, 大大提高自适应界面的智能程度.

由于时间所限, 依据该模型所构建的自适应系统尚有一些不足, 自适应效果有待进一步提高. 未来我们将进一步优化动作预测方法, 增强动作预测效果, 改进界面布局机制和交互方式, 使界面适应用户更加自然流畅.

致 谢 向对本文提出建议的同行和审稿人表示感谢!

参 考 文 献

- [1] Schneider-Hufschmidt M, Kuhme T, Malinowski U. Adaptive User Interfaces: Principles and Practice. New York: Elsevier Science Inc, 1993
- [2] Viano G, Parodi A, Alty J. Adaptive user interface for process control based on multi-agent approach//Proceedings of the Working Conference on Advanced Visual Interfaces (ACM AVI '00). Palermo, Italy, 2000: 201-204
- [3] Liu J, Wong C K, Hui K K. An adaptive user interface based on personalized learning. IEEE Intelligent Systems, 2003, 18(2): 52-57
- [4] Burnett M, Cook C, Rothermel G. End-user software engineering. Communications of the ACM, 2004, 47(9): 53-58
- [5] Malinowski U, Kühme T, Dieterich H. A taxonomy of adaptive user interfaces//Proceedings of the Conference on People and Computers VII. York, United Kingdom, 1993: 391-414
- [6] Findlater L, McGrenere J. A comparison of static, adaptive, and adaptable menus//Proceedings of the SIGCHI Conference on Human Factors in Computing Systems. Vienna, Australia, 2004: 89-96
- [7] Lavie T, Meyer J. Benefits and costs of adaptive user interfaces. International Journal of Human-Computer Studies, 2010, 68(8): 508-524
- [8] Martins C, Azevedo I, de Carvalho C V. The use of an adaptive hypermedia learning system to support a new pedagogical model//Proceedings of the 5th IEEE International Conference on Advanced Learning Technologies. Kaohsiung, China, 2005: 832-833
- [9] Johnson C M, Johnson T R, Zhang Jiajie. A user-centered framework for redesigning health care interfaces. Journal of Biomedical Informatics, 2005, 38(1): 75-87
- [10] Fink J, Kobsa A, Nill A. Adaptable and adaptive information access for all users, including the disabled and the elderly//Proceedings of the User Modelling. ChiaLaguna, Italy, 1997: 171-173
- [11] Kobsa A, Koenemann J, Pohl W. Personalized hypermedia presentation techniques for improving online customer relationships. The Knowledge Engineering Review, 2001, 16(2): 111-155
- [12] Langley P, Fehling M. The experimental study of adaptive user interfaces. Institute for the Study of Learning and Expertise; Palo Alto, CA. Technical Report: 98-3, 1998
- [13] Igarashi T, Hughes J F. A suggestive interface for 3D drawing//Proceedings of the 14th Annual ACM Symposium on User Interface Software and Technology. Orlando, USA, 2001: 173-181
- [14] Pu P, Viappiani P, Faltings B. Increasing user decision accuracy using suggestions//Proceedings of the SIGCHI Conference on Human Factors in Computing Systems. Montreal, Canada, 2006: 121-130
- [15] Brusilovsky P. KnowledgeTree: A distributed architecture for adaptive e-learning//Proceedings of the 13th International World Wide Web Conference. New York, USA, 2004: 104-113
- [16] Yoo J H, Ahn K S, Jun J, Rhee P K. A recommendation system for intelligent user interface: Collaborative filtering approach//Proceedings of the 8th International Conference on Knowledge-Based Intelligent Information & Engineering Systems. Wellington, New Zealand, 2004: 869-879
- [17] Sun Zheng-Xing, Peng Bin-Bin, Cong Lan-Lan et al. Study on user adaptation for on-line sketch recognition. Journal of Computer-Aided Design & Computer Graphics, 2004, 16(9): 1207-1215 (in Chinese)
(孙正兴, 彭彬彬, 丛兰兰等. 在线草图识别中的用户适应性研究. 计算机辅助设计与图形学学报, 2004, 16(9): 1207-1215)
- [18] Duarte C, Carrio L. A conceptual framework for developing adaptive multimodal applications//Proceedings of the 11th International Conference on Intelligent User Interfaces. Sydney, Australia, 2006: 132-139
- [19] Xu Li-Shuang, Zhou Ming-Jun, Deng Chang-Zhi et al. A conceptual framework for developing adaptive pen-based user interface. Journal of Software, 2008, 19(10): 2681-2693 (in Chinese)

(徐礼爽, 周明骏, 邓昌智等. 自适应笔式用户界面开发的概念框架. 软件学报, 2008, 19(10): 2681-2693)

- [20] Thompson C A, Goker M H, Langley P. A personalized system for conversational recommendations. *Journal of Artificial Intelligence Research*, 2004, 21: 393-428
- [21] Rogers S, Fiechter C N, Langley P. An adaptive interactive agent for route advice//*Proceedings of the 3rd Annual Conference on Autonomous Agents*. Seattle, USA, 1999: 198-205
- [22] Tsandilas T, Schraefel M C. Usable adaptive hypermedia systems. *Journal of Hypermedia*, 2004, 10(1): 5-29
- [23] Venkatakrishnan B A, Murugesan S. Adaptation of Web pages for hand-held devices//*Proceedings of the 5th Interna-*

tional Conference on Web Engineering. Sydney, Australia, 2005: 435-440

- [24] Puerta A, Eisenstein J. Towards a general computational framework for model-based interface development systems. *Knowledge-Based Systems*, 1999, 12(8): 433-442
- [25] Norman D A, Bobrow D G. On data-limited and resource-limited processes. *Cognitive Psychology*, 1975, 7(1): 44-64
- [26] Toninelli A, Montanari R, Kagal L, Lassila O. Proteus: A semantic context-aware adaptive policy model//*Proceedings of the 8th IEEE International Workshop on Policies for Distributed Systems and Networks (POLICY'07)*. Bologna, Italy, 2007: 129-140



FAN Yin-Ting, born in 1975, Ph. D. candidate, lecturer. His research interests include human-computer interaction and adaptive user interface.

TENG Dong-Xing, born in 1973, Ph. D., associate professor. His research interests include visual analysis techniques and user interface techniques.

YANG Hai-Yan, born in 1980, Ph. D., assistant pro-

fessor. Her research interests include human computer interaction and sketch user interface.

MA Cui-Xia, born in 1975, Ph. D., associate professor. Her research interests include human computer interaction, sketch computing and context-aware interaction techniques.

DAI Guo-Zhong, born in 1944, professor. His research interests include human-computer interaction, computer graphics.

WANG Hong-An, born in 1963, Ph. D., professor. His research interests include real-time intelligence, user interface.

Background

Adaptive user interface is an important division in Human-Computer Interaction. It stands for the future trends of end user interfaces. The research goal of adaptive user interface is to reduce the display and interactions of customized systems according to the users' individual needs, application environments and conditions.

According to the current researches on adaptive user interface, there are two major deficiencies: on the one hand, the experiences which users interact with system are not made full use of, so the system presents static behaviors on interface adaptation, and there is no self-learning the user's experiences to adapt to them dynamically; on the other hand, the researches on adaptive user interface are more concentrated on specific applications, while the researches on adaptive user interface model and its implementation framework are not enough, and the interaction mechanism and expression for adaptive user interface model lack of systematic study. Therefore current researches cannot effectively improve the design and development of adaptive user interface, and restrict the popularization of adaptive user interface in more applications.

The goal of this study is to propose an adaptive user interface model based on experience awareness. It aims to address the problem that the user interface software cannot self-learn the user's interaction experiences, and cannot obtain

their intent accurately.

The main contributions of the work are that we propose an adaptive user interface model based on experience awareness, establish a unified description of static interface elements, dynamic interactive actions as well as adaptive strategies, and study the implementation architecture, key techniques and modeling method of adaptive interface model. The advantages of the proposed model are provided: (1) based on the user's interaction history, the adaptive interface can self-learn user's experiences and adapt to them intelligently. (2) the model can provide more natural and fluent interaction sequences. (3) the prediction method based on interaction actions can help to improve the adaptive effect by self-learning user's experiences, and reduce their cognitive loads and interactive actions. (4) the adaptive strategy combined the length of sequence matching with frequency of action can improve the intelligence of adaptive user interface.

This work was supported by the National Natural Science Foundation of China under Grant No. 60703078 and No. 61173057, the National Basic Research Program (973 Program) of China under Grant No. 2011CB302205, the National High Technology Research and Development Program (863 Program) of China under Grant No. 2011AA040501.