

**BI+CHUKONG JIAOHUJIEMIAN
DE SHEJICELUE YU YANJIU**

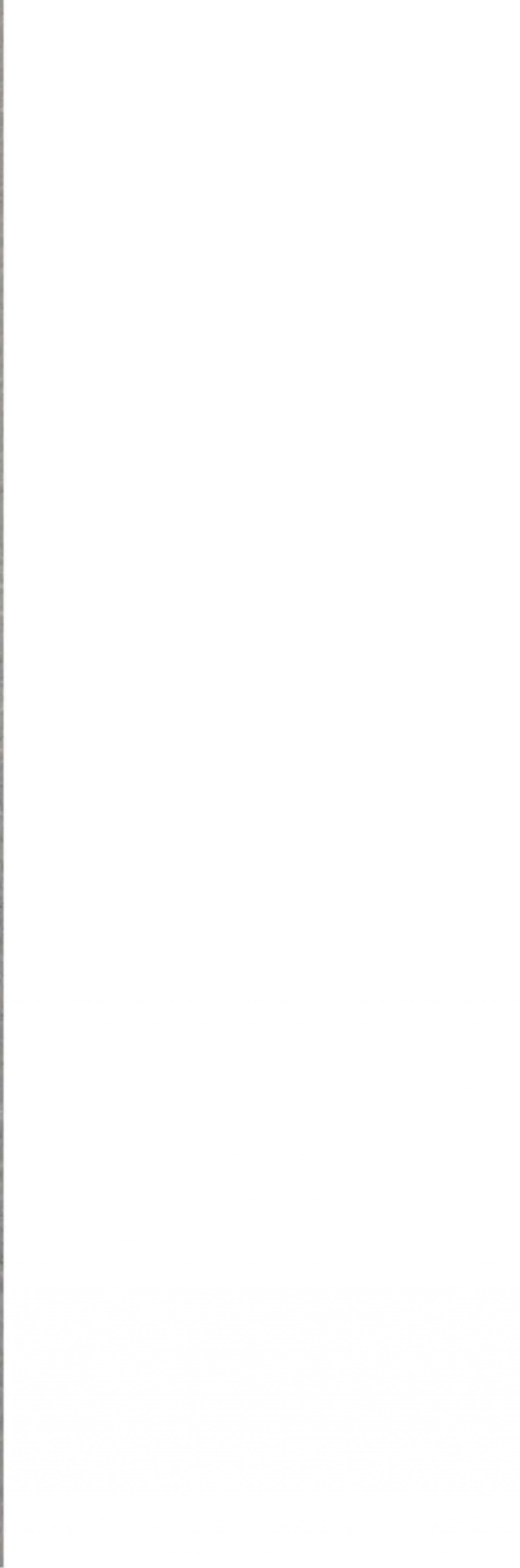
笔+触控交互界面的 设计策略与研究

殷继彬等
著



云南大学出版社
YUNNAN UNIVERSITY PRESS

随着硬件技术的快速发展，支持笔+触控双手交互的设备也越来越受欢迎，相应的关于笔+触控交互方式的研究也变得更加重要。与传统的基于键盘和鼠标交互的WIMP用户界面相比，笔+触控结合的交互方式不但更加直观和自然，而且通过笔输入通道拓宽交互界面的输入方式。笔的输入模式中最重要的一个属性就是笔压输入，本书主要针对笔+触控界面中的笔压输入和笔压应用进行研究。







出品人：吴云
策划编辑：叶枫红
责任编辑：石可
装帧设计：刘文娟



ISBN 978-7-5482-2654-3



9 787548 226543 >

定价：36.00元

**BI+CHUKONG JIAOHUJIEMIAN
DE SHEJICELUE YU YANJIU**

笔+触控交互界面的 设计策略与研究

殷继彬 钱 谦 王 锋 姜泽华 陈苗云 程国军 左亚敏 著



云南大学出版社
YUNNAN UNIVERSITY PRESS

图书在版编目 (CIP) 数据

笔+触控交互界面的设计策略与研究 / 殷继彬等著

— 昆明 : 云南大学出版社, 2016

ISBN 978-7-5482-2654-3

I. ①笔… II. ①殷… III. ①人机界面—程序设计—研究 IV. ①TP311.1

中国版本图书馆CIP数据核字 (2016) 第113384号

出品人: 吴云

策划编辑: 叶枫红

责任编辑: 石可

装帧设计: 刘文娟

BI+CHUKONG JIAOHUJIEMIAN DE SHEJICELUE YU YANJIU

笔+触控交互界面的 设计策略与研究

殷继彬 钱谦 王锋 娄泽华 陈苗云 程国军 左亚敏 著

出版发行: 云南大学出版社

印 装: 云南南方印业有限责任公司

开 本: 787mm×1092mm 1/16

印 张: 12

字 数: 215千

版 次: 2016年6月第1版

印 次: 2016年6月第1次印刷

书 号: ISBN 978-7-5482-2654-3

定 价: 36.00元

社 址: 昆明市翠湖北路2号云南大学英华园内

邮 编: 650091

电 话: (0871) 65033244 65031071

网 址: <http://www.ynup.com>

E-mail: market@ynup.com

本书若有质量问题, 请与印厂调换。(联系电话: 0871-65148757)

前 言

笔+触控界面融合了笔式和触控交互的优势，有效地克服了触控界面中固有的“肥手指”问题，改善了触控界面的低输入精度问题，并具有双手操作的直观、自然特征，拓宽了计算机信息输入通道，符合人机交互的智能化发展方向。近年来，它正逐步成为人机交互（Human Computer Interaction，简称HCI）一个新的研究热点。

本书紧扣人机交互的发展趋势，主要针对笔+触控交互这一前沿性交互方式，力争在其基础研究方面有所突破和创新，进而丰富和完善基于笔+触控界面的人机交互理论和技术体系，推动其开发应用和普及。笔+触控交互目前处于概念性研究阶段，缺乏完善的基础理论体系，同时不同类型的输入模态（例如模糊和精确输入）的组合使用常常引起有别于传统人机界面的、不容忽视的人因问题。本书主要从三个方面对此加以深入研究：一是研究在笔+触控交互模式下用户对多输入参数的控制能力，并给出参数初值和交互控制量之间的最优映射关系曲线，研究多输入模态的耦合算法，提出混合手势设计方法；二是结合界面中的人因工程、认知心理方面的重要问题，研究笔+触控界面交互范式、用户界面设计和评估原则；三是对关键交互任务提出一个新的模型，解决此领域至今尚无一个被广泛接受的人体工效理论模型的问题。本书的研究课题为下一代自然交互界面研究提供了有力的理论基础和方法支持，实现若干原始创新，有助于提高和推广笔+触控界面技术。

全书共三篇，并且每篇都设有一个实验项目，以便于读者更好地理解。第一篇笔+触控交互技术，设计了一个基于Steering轨迹任务的通道穿越实验，并对用户使用惯用手和非惯用手执行触摸任务的表现进行研究。第二篇基于多点触控界面的笔+触控交互手势设计与研究，利用appBook多平台移动应用制作工

具作为载体，以HTML5标准进行web编程并在iPad3上进行测试开发。本篇将给出制定简单、直观和鲁棒的笔+触控用户界面混合手势设计方法和指导性准则，推动笔+触控交互的实际应用。第三篇基于多点触控的桌面文档操作交互的设计与研究，分析单点触摸和多点触摸的属性并提出相应的数据结构形式。通过对微软Windows操作系统中桌面文档图标的操作详细分析，确定图标的操作方式和流程，然后讨论在触摸操作下的系统总体架构及如何实现单个图标与多个图标操作的不同手势。

由于作者的知识与写作水平有限，书中缺点和错误在所难免，殷切希望广大读者批评指正。

作 者
2016年3月

目 录

第一篇 笔+触控交互技术	1
概 要	1
1. 绪 论	2
1.1 选题背景与研究意义	2
1.2 国内外研究现状及发展动态	3
1.2.1 基于多点触控交互的桌面系统介绍	3
1.2.2 触摸输入的研究	4
1.2.3 基于手势交互的桌面系统	5
1.2.3.1 手势识别技术	5
1.2.3.2 手势输入的形式	6
1.2.4 笔压输入的研究	7
1.2.5 双手交互技术及理论	7
1.2.5.1 双手操作机制研究	8
1.2.5.2 笔+触控交互研究	10
1.3 小 结	11
2. 人机交互基础理论介绍	12
2.1 人机交互模型简述	12
2.2 人机交互界面设计评价模型	13
2.2.1 Fitts Law模型	13
2.2.2 Steering Law模型	14
2.3 笔+触控交互技术介绍	16
2.3.1 触控交互技术	16
2.3.2 笔交互技术	18
2.3.3 笔交互与触控交互关键特征对比	19

2.4 小 结	19
3. 基于笔压控制的双手通道穿越实验的设计	20
3.1 实验程序开发技术概述	20
3.3.1 Xcode开发工具	20
3.3.2 Objective-C概述	21
3.3.3 iOS平台概述	22
3.3.4 Jot Touch SDK介绍	23
3.2 实验设备及开发环境	23
3.2.1 实验软硬件	23
3.2.2 Jot Touch 4.0连接与使用	24
3.3 主要实验内容及正式实验设计	25
3.4 实验程序及执行步骤	26
3.4.1 程序输入事件处理	26
3.4.2 实验执行的主要过程	27
3.5 小 结	28
4. 实验结果分析与讨论	28
4.1 实验一：确定通道穿越实验通道宽度的范围	28
4.1.1 实验目的	28
4.1.2 实验内容	29
4.1.3 实验参与者及任务安排	31
4.1.4 实验结果	32
4.2 实验二：自然书写与自然握笔时的笔压分析实验	37
4.2.1 实验目的	37
4.2.2 实验内容	37
4.2.3 实验参与者及任务安排	39
4.2.4 实验结果	40
4.3 实验三：基于笔压控制的双手通道穿越实验	41
4.3.1 实验目的	41
4.3.2 实验内容	42
4.3.3 实验参与者及任务安排	44
4.3.4 实验结果	44
4.4 小 结	49

5. 笔+触控界面中基于压力的交互技术的设计与实现	49
5.1 笔+触控交互技术探究	49
5.2 基于压力的交互技术设计与实现	52
5.2.1 基于压力操作的双手拓展手势技术	52
5.2.2 基于压力双手操作的离散目标选择改进方法	53
5.2.3 基于压力的双手多级菜单操作	54
5.3 基于笔压的双手交互技术在图像编辑程序中的应用	55
5.4 界面外基于压力操作的双手交互技术设想	58
5.5 小 结	58
6. 总结与展望	59
6.1 本篇工作总结	59
6.2 未来工作及展望	59
参考文献	60
 第二篇 基于多点触控界面的笔+触控交互手势设计与研究	66
概 要	66
1. 绪 论	67
1.1 选题背景及研究意义	67
1.1.1 选题背景	67
1.1.2 选题意义	70
1.2 国内外研究现状及发展动态分析	72
1.3 研究内容、目标	73
2. 人机交互基础理论	74
2.1 人机交互介绍	74
2.1.1 人机交互概念	74
2.1.2 人机交互与人机界面	74
2.1.3 人机交互与其他学科的关系	74
2.1.4 人机交互技术发展	75
2.2 自然人机交互主要技术手段	77
2.2.1 笔式交互技术	77
2.2.2 多点触控交互技术	77
2.2.3 语音交互技术	78

2.2.4 视觉交互技术·····	78
2.2.5 多通道交互技术及其他交互技术·····	78
3. 多点触控及常用触控手势的实现 ·····	79
3.1 实验平台和开发环境·····	79
3.2 多点触控web开发·····	79
3.2.1 触摸事件·····	80
3.2.1.1 触摸事件介绍·····	80
3.2.1.2 web中触摸事件的监听·····	81
3.2.1.3 创建和监听自定义事件·····	84
3.2.1.4 开发建议·····	85
3.2.1.5 设备支持·····	86
3.2.2 手势事件·····	87
3.3 常用触控手势·····	87
4. 笔画识别算法 ·····	92
4.1 人机交互中的笔画识别·····	92
4.2 \$1手势识别算法·····	94
4.2.1 \$1手势识别算法面临的挑战·····	94
4.2.2 \$1手势识别算法的原理·····	95
4.2.2.1 原始手势数据处理过程·····	95
4.2.2.2 旋转不变性分析·····	101
4.2.2.3 \$1识别器的局限性·····	103
5. 基于多点触控界面的笔+触控交互手势设计与实现 ·····	104
5.1 笔+触控交互介绍·····	104
5.2 实验平台和开发环境·····	105
5.3 手势设计与实现·····	105
5.3.1 设计原理·····	105
5.3.2 6种混合手势组合·····	106
5.4 笔+触控手势在图片管理系统中的应用·····	111
6. 实验设计与分析 ·····	115
6.1 实验内容·····	115
6.2 实验结果分析·····	115
7. 总结与展望 ·····	116

7.1 本篇总结及未来发展展望	116
参考文献	117
第三篇 基于多点触控的桌面文档操作交互的设计与研究	122
概 要	122
1. 绪 论	123
1.1 研究背景	123
1.2 研究现状与发展	125
1.3 相关研究	126
1.4 本篇的主要内容	127
1.5 本篇的研究意义	127
2. 开发环境	128
2.1 硬件设备	128
2.2 软件工具	128
2.2.1 Unity简介	129
2.2.2 JDK简介	130
2.2.3 Android SDK简介	130
2.3 调试和导出	131
2.4 小 结	132
3. 技术与分析框架介绍	132
3.1 触 摸	133
3.1.1 单点触摸	133
3.1.2 多点触摸	135
3.2 交互手势	136
3.2.1 单点触摸手势	136
3.2.2 多点触摸手势	137
3.2.3 单点触摸与多点触摸的比较	138
3.3 交互手势与交互任务	139
3.4 Unity触摸	140
3.4.1 两大坐标系之间关系	140
3.4.2 触摸核心Touch	142
3.5 项目框架	142

3.5.1 主要手势及任务·····	142
3.5.2 图标设计·····	143
3.5.3 单点触摸操作·····	144
3.5.4 两点触摸操作·····	146
3.5.5 三点操作·····	156
3.5.6 手势操作的组合·····	157
3.5.7 圆盘菜单·····	158
3.6 小 结 ·····	159
4. 项目实现和代码编写 ·····	159
4.1 MonoDevelop脚本编辑器 ·····	160
4.2 Unity脚本的生命周期 ·····	160
4.3 图标的实现 ·····	160
4.3.1 图标的制作·····	161
4.3.2 图标脚本Icon.cs ·····	161
4.4 触摸手指信息InputControfler.cs ·····	163
4.5 单点触摸操作的实现 ·····	165
4.6 两点触摸操作 ·····	166
4.6.1 缩放旋转操作·····	166
4.6.2 单个图标操作的实现SingleOperation () ·····	171
4.6.3 多个离散图标操作的实现DiscreteOperation () ·····	172
4.6.4 同一类型图标操作SameTypeOperation () ·····	173
4.6.5 连续图标选择操作ContiguousSelection () ·····	174
4.6.6 桌面菜单操作·····	175
4.6.7 移动图标到文件夹操作MoveToFolder () ·····	177
4.7 小 结 ·····	178
5. 实验测试与分析 ·····	179
5.1 实验测试 ·····	179
5.2 实验分析 ·····	179
6. 总结与展望 ·····	180
6.1 总 结 ·····	180
6.2 展 望 ·····	180
参考文献 ·····	181

第一篇 笔+触控交互技术

概 要

随着硬件技术的快速发展,支持笔+触控双手交互的设备也越来越受欢迎,相应地关于笔+触控交互方式的研究也变得更加重要。与传统的基于键盘和鼠标交互的WIMP用户界面相比,笔+触控结合的交互方式不但更加直观和自然,而且通过笔输入通道可以拓宽交互界面的输入方式。笔的输入模态中最重要一个属性就是笔压输入,本篇主要针对笔+触控界面中的笔压输入和笔压应用进行研究。本篇主要的内容如下:

第一,本篇设计了一个基于Steering轨迹任务的通道穿越实验,并对用户使用惯用手和非惯用手执行触摸任务的表现进行研究,目的是为了确定正式实验中穿越通道的最小宽度。通过对通道穿越时间和出错率的分析,得出用户直接触摸执行通道穿越任务的通道宽度不能小于30 pixels的结论。

第二,我们对用户自然书写和草图绘制时的笔压数据进行了记录,整理后发现超过95%的笔压分布在300至1 500之间;同时我们对用户自然握笔时的笔压进行了记录,统计后发现笔尖压力有一段明显的非零值。综合分析后得知,自然握笔时的笔压很难被用户控制和使用。因此,在基于笔压的交互技术设计时,应当考虑到自然握笔时的笔压。

第三,为了进一步研究用户控制笔压的能力和局限性,我们完成了一个基于双手交互的通道穿越实验。实验中我们把可控制的笔压分为1至7层,并且分析和比较了惯用手和非惯用手控在不同压力层次下制笔压任务的性能。通过实验发现多达7层的笔压能被用户控制,但是在环型任务中非惯用手的出错率最高达到了64.5%。除此之外,我们还得到了其他的一些重要结论:惯用手控制笔压的能力优于非惯用手,控制笔压任务比手指触摸任务更难,但是惯用手控制笔压的平均时间比非惯用手要多一点。

第四,在本篇实验研究的基础上,我们对来自双手的输入方式进行了讨论和

分类,并提出一种基于压力的笔+触控结合的交互方式。

第五,我们提出了三种基于压力控制的交互技术,设计了一个图像编辑应用,并实现一些交互技术进行简单应用。

本篇是笔+触控交互技术的基础研究,基础研究对笔+触控界面中基于压力的交互技术设计有明显的指导意义。

1. 绪 论

1.1 选题背景与研究意义

人机交互技术(Human-Computer Interaction Techniques)是计算机科学与技术中的一个重要分支,其中涉及物理学、统计学、心理学等多个学科知识,是一项研究用户和计算机之间交互关系的技术。人机交互主要通过一些输入输出设备(如鼠标、键盘、摇杆和显示器等)、计算机系统或软件来为用户提供相关信息和提示。人机交互的发展史是一部从人类适应计算机到计算机不断适应人类的发展史,它发展至今共经历了以下主要阶段:早期的手工作业阶段、作业控制语言及交互命令语言阶段、图形用户界面(Graphical User Interface,简称GUI)阶段、网络用户界面阶段和多通道、多媒体的智能交互阶段,如图1-1-1所示。人机交互的设计风格也从命令行用户界面逐步发展到以“所见即所得(WYSIWYG)”为特征的图形用户界面,目前正朝着自然智能的交互方向发展。与此同时,用户界面交互技术设计准则也从“以技术为中心”发展逐步转向“以用户为中心(User Centered Design,简称UCD)”发展的新阶段。

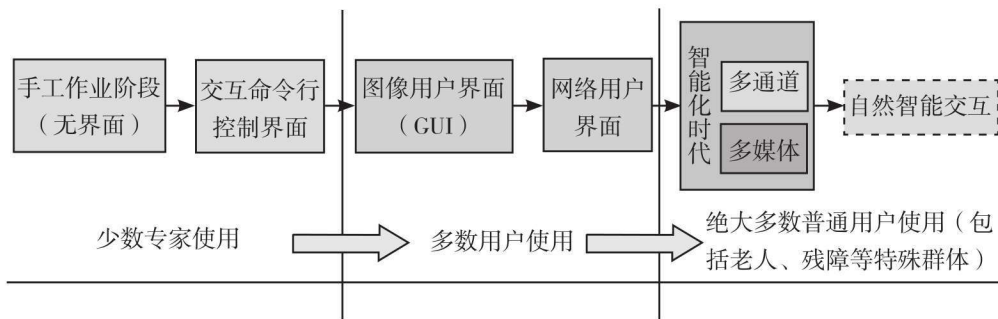


图1-1-1 人机交互各阶段发展的示意图

随着计算机技术的快速发展和科技水平的逐步提高,越来越多的人享受到了人机交互技术的发展成果,人机交互技术的发展在给我们生活带来了方便的同时也受到人们越来越多的关注,逐渐成为现代信息产业中的一个强有力的竞争焦点。美国总统信息技术顾问委员会的《21世纪信息技术报告》中把“人机交互和信息管理”列为21世纪重点发展的4项信息技术之一。人机交互的研究和发展在我国也受到了格外重视,我国的一些重点院校和科研院所都把人机交互技术列为重点研究项目,这些研究项目也受到了国家重点资助。

多点触控技术的突破,使人与计算机之间的交互方式逐渐突破了传统键鼠所构成的物理空间的限制。而且计算机的外在形式也在发生着变化,从固定到便携,从庞大到精微,甚至从有形到无形,未来不同的技术将会加速趋向融合,迫切需要有新的交互方式来适应新时代的变化,交互的中心也逐渐从“以机器为中心”向“以人为中心”转移。目前,已经出现了许多新的交互方式,如手势识别、语音识别、表情识别、脑机接口和笔式交互等,它们为人机交互的发展带来了新的挑战 and 机遇。

硬件技术的快速发展使计算机的信息处理和输出能力不断增强,而用户有限的输入能力与计算机的强大的信息处理和输出能力形成了极大反差,这种输入输出严重的不平衡成为阻碍人机交互发展的瓶颈,笔+触控结合的交互方式就是在这样的背景下提出的。笔+触控结合的交互方式有效地拓宽了人机交互的输入通道,同时让用户体验到了笔和触摸所带来的不同交互体验,因此有不少研究者把目光转向了基于双手的笔+触控交互的研究当中。由于该方向的研究时间并不长,还属于刚刚起步阶段,所以有关此方面的研究成果并不多,但是笔和触控结合的优势和潜力已经很明显,这将会吸引更多的研究者转向笔+触控的研究。在笔+触控结合的交互方式中有两种不同的输入通道,而两者的属性又有天然区别,因此对笔+触控的结合方式、双手的劳动分工以及笔触和手触的输入属性的研究都将成为笔+触控界面中的研究重点。

1.2 国内外研究现状及发展动态

1.2.1 基于多点触控交互的桌面系统介绍

多点触控技术的发展使市场上涌现出很多基于多点触控交互的桌面系统,这也为笔+触控界面中基于双手的交互技术的研究与设计带来新的机遇和挑战。目前比较流行的主要有以下几种:

① Smart Skin: 它是一种利用传感器感知用户交互过程的桌面系统。该系统