

**The Memory Glasses: Wearable Computing for  
Just-in-Time Memory Support**

by

Richard W. DeVaul

Submitted to the Program in Media Arts and Sciences, School of  
Architecture and Planning

in partial fulfillment of the requirements for the degree of

Doctor of Philosophy in Media Arts and Sciences

at the

MASSACHUSETTS INSTITUTE OF TECHNOLOGY

June 2004

© Massachusetts Institute of Technology 2004. All rights reserved.

Author

Program in Media Arts and Sciences, School of Architecture and Planning

April 5th, 2004

Certified by .....

.....

and

Toshiba Professor of Media Arts and Sciences

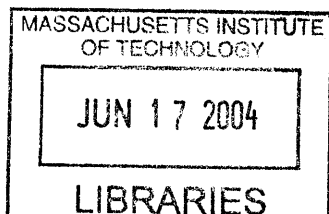
Thesis Supervisor

Accepted by .....

.....

Dr. Arthur Lippman

Chairman, Department Committee on Graduate Students



ROTCH



# **The Memory Glasses: Wearable Computing for Just-in-Time Memory Support**

by

Richard W. DeVaul

Submitted to the Program in Media Arts and Sciences, School of Architecture and Planning

on April 5th, 2004, in partial fulfillment of the requirements for the degree of Doctor of Philosophy in Media Arts and Sciences

## **Abstract**

This thesis documents a body of wearable computing research surrounding the development of the Memory Glasses, a new type of proactive memory support technology. The Memory Glasses combines features of existing memory support technologies (such as PDAs) with a context aware delivery system and a low-attention cuing interface. The goal of the Memory Glasses is to provide effective just-in-time memory support while mitigating some of the distraction and over-reliance problems that can result from the use of more conventional memory support technology.

The Memory Glasses research is a synthesis of the author's six years of work on wearable computing. This thesis documents the author's intellectual contributions in the areas of wearable computing hardware architectures, software architectures, and human-computer interaction. Specific topics include the MITHril wearable computing research platform, the Enchantment middlewear, the MITHril Real-Time Context Engine, the author's modified Seven Stages of Action model and five principles of low-attention wearable human computer interaction, as well as the author's research in the use of subliminal cuing for just-in-time memory support.

Although memory support is the unifying theme of this dissertation, the author's research has seen application in a number of other areas, including the mapping of social networks, research in human physiology and biomedical applications, and group situation awareness and command, control, and communications. A selection of these applications is briefly presented as support for the importance of the author's intellectual contributions.

Thesis Supervisor: Alex P. Pentland

Title: Toshiba Professor of Media Arts and Sciences



# The Memory Glasses: Wearable Computing for Just-in-Time Memory Support

by

Richard W. DeVaul

Thesis Committee:

Thesis Advisor \_\_\_\_\_

\_\_\_\_\_

Alex P. Pentland

Toshiba Professor of Media Arts and Sciences

Massachusetts Institute of Technology

Thesis Reader \_\_\_\_\_

\_\_\_\_\_

Asim Smailagic

Professor, Computer Engineering and Computer Science

Carnegie Mellon University

Thesis Reader \_\_\_\_\_

\_\_\_\_\_

Jeremy Wolfe

Associate Professor, Brain and Cognitive Science

Massachusetts Institute of Technology

Thesis Reader \_\_\_\_\_

\_\_\_\_\_

ad Starnier

Assistant Professor, College of Computing

Georgia Institute of Technology



## Acknowledgments

Thanks to all of the wonderful people I've worked with during my years at the Media Lab — there are far too many to name, so here are just a few: My advisor Sandy, who demonstrated great faith in my research, who gave me the resources and independence to pursue ridiculously ambitious projects, and who made all of the work in this thesis possible. To Thad Starner, whose visionary leadership of the MIT Wearable Computing group inspired me to join the borg, who convinced me that wearable computing could be relevant and useful as well as cool, and in whose footsteps I've tried to follow in inspiring others and spreading the faith. Without Thad, this thesis would be on some other topic entirely, and might even be from some other institution. Thanks to my other thesis readers: Asim Smailagic, whose work in context awareness, rapid prototyping, and user-centered design for wearable computing is inspiring, and to Jeremy Wolfe, the world's leading expert on visual search, whose willingness to critique the psychophysics of a wearable computing researcher is a testament to great generosity of spirit.

Thanks to Ed Keyes, a brilliant wearables hacker who first proposed the use of a subliminal UI for reminders. Thanks to Steve Schwartz for the “Smart Vest” prototype and his years of collaboration on the MIThril project. Thanks Brad Rhodes and Josh Weaver and Lenny Fonner and Dana Kirsch and Jennifer Healey and all of the “old guard” borg who made the wearable computing group a real community of researchers. Thanks to Jonathan Gips and Mike Sung and Anmol Madin and Chris Elledge and Daniel Barkalow and David Kaplowitz and Nathan Eagle and Martin Martin and Ron Caneel and Juan Carlos Barahona and Ivan Chardin and Jacky Mallet and Chris LaPre and all the others I have worked with who have continued that tradition.

Thanks to Chris Wren and Tanzeem Choudhury and Brian Clarkson and Sumit Bassu and Tony Jebara and Lee Campbell and Yuri Ivanov and Nitin Sawhney, who were friends as well as colleagues, and from whom I learned a great deal. To Ben Vigoda and Yael Maguire in the Physics and Media Group, for mathematical modeling clue and friendship. To Ari Benbasat, for engineering collaboration and friendship, and proving that Canadian engineers can be just as competent (and wear as much black) as any other MIT hacker.

Thanks to my family, for their unwavering love and support throughout the years, and who have inspired me with their accomplishments and devotion. I really can't say enough about how important you are.

Thanks to my many friends who supported me during the ups and downs of my work at the Media Lab. To E'beth Haley, for lots of vegan dinners, inspiration on the Aikido mat, and all the black T-shirts. To Shoshana Gourdin, who among other things helped schedule subjects for my last experiment, freeing me to concentrate on my research and writing. To Spike Holcomb, for the amazing food, amazing drinks, and inspiring life. And last (and not least) to Vicka Corey, Ph.D, without whom my experimental design, my living situation, and my stress about my research would be much the worse.

I am so glad that I did not have to do it without you.

# Contents

<b>1</b>	<b>Introduction</b>	<b>23</b>
1.1	Organization of the Thesis . . . . .	25
<b>2</b>	<b>Memory Support</b>	<b>27</b>
2.1	Introduction . . . . .	27
2.2	Remembering and Forgetting . . . . .	27
2.3	Memory Support Technology . . . . .	28
2.3.1	Idealized Memory Support . . . . .	29
2.3.2	Memory Aids vs. Memory Surrogates . . . . .	29
2.3.3	Memory Support vs. Sharing Experience . . . . .	30
2.3.4	The Dangers of Memory Support . . . . .	31
2.4	Organizers . . . . .	32
2.4.1	Organizers and Context . . . . .	32
2.4.2	Proactive vs.. Passive Organizers . . . . .	33
2.4.3	Context Awareness . . . . .	33
2.4.4	Records . . . . .	35
2.4.5	Reminders . . . . .	36
2.4.6	The Advantages of Context Awareness . . . . .	37
2.5	Memory Support in Context . . . . .	38
2.5.1	Mnemonic Devices . . . . .	38
2.5.2	Paper Organizers . . . . .	40
2.5.3	Memory Drugs . . . . .	40
2.5.4	PDA's . . . . .	40

2.5.5	Forget-me-not . . . . .	41
2.5.6	The Wearable Remembrance Agent . . . . .	42
2.5.7	The Memory Glasses . . . . .	42
<b>3</b>	<b>MIThril Hardware</b>	<b>45</b>
3.1	Origins: Lizzy . . . . .	45
3.2	Motivations for MIThril . . . . .	45
3.3	Between Lizzy and MIThril . . . . .	47
3.3.1	Smart Vest . . . . .	47
3.3.2	Sony Vaio Picturebook Vest . . . . .	47
3.3.3	Cell Computing Vest . . . . .	48
3.4	MIThril 2000 . . . . .	48
3.4.1	MIThril Overview . . . . .	50
3.4.2	Memory Glasses Platform Requirements . . . . .	51
3.4.3	MIThril 2000 Architecture . . . . .	52
3.4.4	Packaging . . . . .	53
3.4.5	Sensors and the MIThril body-bus . . . . .	55
3.4.6	Computing Nodes and the MIThril 2000 Body Network . . . . .	57
3.4.7	User Interface Hardware . . . . .	59
3.4.8	MIThril 2000 System Software . . . . .	61
3.4.9	MIThril 2000 Evaluation and Experience . . . . .	62
3.5	MIThril 2003 . . . . .	64
3.5.1	Overview . . . . .	64
3.5.2	Architecture . . . . .	65
3.5.3	Computing Nodes . . . . .	65
3.5.4	Sensor Hubs . . . . .	66
3.5.5	Sensor Hardware . . . . .	67
3.5.6	Networking . . . . .	67
3.5.7	MIThril 2003 Evaluation and Experience . . . . .	68
3.6	Beyond the Memory Glasses: Other Applications for MIThril . . . . .	68

3.6.1	Parkinson’s Project . . . . .	68
3.6.2	SAAB WISE Project . . . . .	68
3.6.3	Shortcuts Project . . . . .	69
3.6.4	David Kaplowitz: Daily Use for Systems Administration . . . . .	69
<b>4</b>	<b>MIThril Software</b>	<b>71</b>
4.1	Linux Foundations . . . . .	71
4.2	Middlewear: Software for Real-Time Context Awareness . . . . .	71
4.2.1	The Enchantment Whiteboard . . . . .	72
4.2.2	The Enchantment Signal System . . . . .	73
4.3	The MIThril Real-Time Context Engine . . . . .	74
4.3.1	Classifier Development: Three-Class Model Example . . . . .	76
4.3.2	Choice of Sensor . . . . .	77
4.3.3	Location of Sensor . . . . .	78
4.3.4	Sampling Rate and Resolution . . . . .	78
4.3.5	Data Collection . . . . .	78
4.3.6	Feature Selection . . . . .	79
4.3.7	Model Selection . . . . .	80
4.3.8	Model Training . . . . .	81
4.3.9	Model Evaluation . . . . .	81
4.3.10	Real-Time Implementation . . . . .	82
4.3.11	Other Models . . . . .	82
4.3.12	Conclusions . . . . .	83
4.4	Beyond the Memory Glasses: Other Applications . . . . .	83
4.4.1	Group Interaction Research using the Enchantment Middlewear . . . . .	83
4.4.2	Biomedical Applications of the MIThril Real-Time Context Engine . . . . .	84
<b>5</b>	<b>Wearable Human Computer Interaction for Memory Support</b>	<b>87</b>
5.1	Introduction . . . . .	87
5.2	From Command Line to Cue . . . . .	88
5.2.1	Direct Manipulation, Distraction, and Memory . . . . .	89

5.2.2	Rethinking the Interaction . . . . .	90
5.3	Wearable Interaction: Extending Existing Models . . . . .	91
5.3.1	The Seven Stages of Action . . . . .	92
5.3.2	Seven Stages of Action in “Wearables World” . . . . .	93
5.3.3	Implications of the Modified Norman Model for Wearable Interac- tion Design . . . . .	94
5.3.4	GOMS Models . . . . .	95
5.3.5	GOMS Models in “Wearables World” . . . . .	96
5.4	The Five Principles of Low-Attention Interaction for Wearable Computing .	97
5.5	Early Memory Glasses Prototypes . . . . .	99
5.5.1	Medical Applications . . . . .	99
5.5.2	Amnesia/Prosopagnosia Memory Glasses Prototype . . . . .	101
5.5.3	Evaluation . . . . .	102
5.6	Anduin Window Manager . . . . .	104
5.6.1	Evaluation . . . . .	105
5.7	Anduin/Enchantment-based Memory Glasses . . . . .	107
5.7.1	Evaluation . . . . .	109
5.8	The Limitations of Low-Attention Interaction Strategies . . . . .	110
5.9	Subliminal UI . . . . .	111
5.9.1	Summary of Cuing Experiment Design . . . . .	112
5.9.2	Summary of Results: the Subliminal Visual Cuing Experiment . . .	113
5.9.3	Summary of Results: the Overt Visual Cuing Experiment . . . . .	114
5.9.4	Summary of Results: the Masked Superliminal Audio Cues Exper- iment . . . . .	114
5.9.5	Larger Experiment Analysis . . . . .	115
5.9.6	Implications . . . . .	116
<b>6</b>	<b>Intellectual Contributions</b>	<b>119</b>
6.1	Hardware Architectures for Wearable Computing: MIThril 2000/2003 . . .	119
6.1.1	Clothing Integration without Textile-based Computing Hypothesis .	120

6.1.2	Wearability Hypothesis . . . . .	121
6.1.3	Scalability Hypothesis . . . . .	122
6.1.4	The MITHril Body Bus and Body Network . . . . .	122
6.1.5	MITHril 2000 Kopin Video Driver Engineering . . . . .	124
6.2	Software Architectures . . . . .	125
6.2.1	Enchantment Middlewear . . . . .	126
6.2.2	Enchantment Whiteboard . . . . .	126
6.2.3	MITHril Real-Time Context Engine . . . . .	128
6.3	Wearable Human Computer Interaction . . . . .	129
6.3.1	Cuing Research . . . . .	133
<b>7</b>	<b>Conclusions and Future Work</b>	<b>135</b>
7.1	MITHril Hardware Conclusions . . . . .	135
7.2	MITHril Software Conclusions . . . . .	135
7.3	Human Computer Interaction Conclusions . . . . .	136
7.4	Future Work . . . . .	137
<b>A</b>	<b>Attention and Wearable Interaction</b>	<b>139</b>
A.1	Introduction . . . . .	139
A.2	Over reliance and Memory Support . . . . .	140
A.2.1	Probabilistic Context Classification and Misleading Users . . . . .	140
A.3	Attention and Task Interference for Wearable HCI . . . . .	141
A.4	Subliminal Perception and Wearable HCI . . . . .	142
A.4.1	Unattended Stimuli . . . . .	142
A.4.2	Unconscious Cognition . . . . .	143
A.4.3	Subliminal Perception . . . . .	143
A.4.4	Subliminal Cuing for Wearable and Mobile Applications . . . . .	145
A.4.5	Subliminal Cuing and Attention . . . . .	145
A.4.6	Miscues and Subliminal Perception . . . . .	146
A.4.7	Masked Superliminal Audio Cues . . . . .	146

<b>B</b>	<b>Memory Support Experiments</b>	<b>149</b>
B.1	Introduction . . . . .	149
B.2	Experimental Procedure . . . . .	149
B.2.1	“Social Memory Assistant” Scenario . . . . .	150
B.2.2	Design Overview . . . . .	150
B.2.3	Procedure . . . . .	151
B.2.4	Memory Trials . . . . .	152
B.3	Subliminal Visual Cuing Experiment . . . . .	153
B.3.1	Hypotheses . . . . .	154
B.3.2	Orientation . . . . .	154
B.3.3	Experiment-specific Procedure . . . . .	154
B.3.4	Equipment . . . . .	154
B.3.5	Cuing Strategy . . . . .	155
B.3.6	Results . . . . .	156
B.3.7	Miscues and Spreading Activation . . . . .	158
B.4	Overt Visual Cuing Experiment . . . . .	159
B.4.1	Hypotheses . . . . .	160
B.4.2	Orientation . . . . .	160
B.4.3	Experiment-specific Procedure . . . . .	160
B.4.4	Equipment . . . . .	160
B.4.5	Cuing Strategy . . . . .	160
B.4.6	Results . . . . .	161
B.5	Masked Superliminal Audio Cues Experiment . . . . .	162
B.5.1	Hypotheses . . . . .	163
B.5.2	Orientation . . . . .	163
B.5.3	Experiment-specific Procedure . . . . .	163
B.5.4	Equipment . . . . .	165
B.5.5	Cuing Strategy . . . . .	165
B.5.6	Results . . . . .	167
B.6	Overall Experimental Analysis . . . . .	168

B.7 Experimental Conclusions . . . . . 170  
B.7.1 Subliminal Visual Cuing Experiment Conclusions . . . . . 170  
B.7.2 Overt Visual Cuing Experiment Conclusions . . . . . 170  
B.7.3 Masked Superliminal Audio Cuing Experiment Conclusions . . . . . 171  
B.7.4 Overall Experimental Conclusions . . . . . 172



# List of Figures

3-1	Members of the original wearable computing group wearing Lizzy-style wearables. Picture by Sam Ogden. . . . .	46
3-2	MIThril functional schematic. This drawing was done early in the development of MIThril, and there are some minor differences between this schematic and the more modern MIThril 2000 prototype shown in Figure 3-3.	49
3-3	One possible MIThril 2000 configuration. . . . .	50
3-4	The author wearing his MIThril 2000 system, with and without a shirt covering. . . . .	53
3-5	MIThril 2000 soft mount. . . . .	54
3-6	MIThril cable anchor. . . . .	55
3-7	MIThril body bus junction. The Body Busy uses Hirose HRS3516-SR connectors for locking, high-insertion-cycle, high-torque-rated, strain-relieved connections. . . . .	56
3-8	The first version of the Crystal Tag IR active tag was developed by Ari Benbasat Rich DeVaul. Picture by Josh Weaver. . . . .	57
3-9	A MIThril body-net hub. . . . .	58
3-10	MicroOptical clip-on without IR tag reader. . . . .	60
3-11	BrightStar ipEngine1 with an early prototype MIThril Kopin video driver board. . . . .	61
3-12	MIThril 2003 system, comprised of the Zaurus PDA (right) with Hoarder sensor hub and physiological sensing board (top), EKG/EMG/GSR/ temperature electrodes and sensors (left) and combined three-axis accelerometer, IR tag reader, and RS232/I2C bridge (bottom) . . . . .	66

4-1	The MIThril Real-Time Context Engine . . . . .	75
4-2	MIThril accelerometer sensor . . . . .	77
4-3	Three State Gaussian Activity Classifier: Spectral Features Plus Classifier Evaluation on Testing Data. . . . .	80
5-1	Donald Norman’s Seven Stages of Action model of human-system interac- tion. . . . .	92
5-2	Modified Seven Stages of Action model representing both the mediation and the divided attention state produced by the use of an interactive wear- able computing system. . . . .	94
5-3	HMD display from the Amnesia/Agnosia Memory Glasses prototype, show- ing information about a location in the Borglab (the dev-tty1 workstation) and a person (Sandy Pentland). . . . .	102
5-4	A screen grab of the Anduin-based Memory Glasses system delivering an overt cue – the name and project information for the author. . . . .	107
5-5	A screen grab of the Anduin-based Memory Glasses system showing the user’s position on room-level map. The delivery of this information is trig- gered by the user’s action context — walking. . . . .	108
B-1	MIThril 2000 BSE Dock video driver hardware and MicroOptical QVGA clip-on display . . . . .	155
B-2	Example Cue and Mask Image . . . . .	156
B-3	Histograms comparing the effect of correct (“C”) and incorrect (“X”) masked short-duration visual cues. . . . .	157
B-4	Histograms showing subject performance on “n”, “C”, and “X” cue condi- tions for the masked short-duration visual cues experiment. . . . .	159
B-5	Histograms showing subject performance on “n”, “C”, and “X” cue condi- tions for the overt visual cues experiment. . . . .	161
B-6	Histograms comparing the effect of correct (“C”) and incorrect (“X”) overt visual cues. . . . .	162

B-7 Spectrogram of “Dorothy” audio cue used in masked superliminal audio cues experiment. The time axis is in seconds. . . . . 165

B-8 Histograms showing subject performance on “n”, “C”, and “X” cue conditions for the masked superliminal audio cues experiment. . . . . 167

B-9 Histograms comparing the effect of correct (“C”) and incorrect (“X”) masked superliminal audio cues. . . . . 168



# List of Tables

5.1	Within-subjects hypothesis testing for masked short-duration visual cues. The “n” condition is the control condition. The “C” condition is the correctly cued condition. The “X” condition is the miscued condition. For further explanation, see Appendix B. . . . .	113
5.2	Within-subjects hypothesis testing for overt visual cues. The “n” condition is the control condition. The “C” condition is the correctly cued condition. The “X” condition is the miscued condition. For further explanation, see Appendix B. . . . .	114
5.3	Hypothesis testing for masked superliminal audio cues. The “n” condition is the control condition. The “C” condition is the correctly cued condition. The “X” condition is the miscued condition. For further explanation, see Appendix B. . . . .	115
B.1	Within-subjects hypothesis testing for masked short-duration visual cues. . . . .	157
B.2	Between-subjects hypothesis testing for masked short-duration visual cues. . . . .	158
B.3	Between-subjects hypothesis testing for overt visual cues. . . . .	162
B.4	Within-subjects hypothesis testing for overt visual cues. . . . .	163
B.5	Between-subjects hypothesis testing for masked superliminal audio cues. . . . .	167
B.6	Hypothesis testing for masked superliminal audio cues. . . . .	168
B.7	Frequency of misled answers on “X” condition trials across cuing-method conditions. . . . .	169



# Chapter 1

## Introduction

This thesis grew out of my personal experience with wearable computing and memory support technology. My introduction to wearable computing occurred shortly after I arrived at the Media Lab in the fall of 1997; I met Brad Rhodes sitting on the front steps of the Lab wearing a dashing beret and a Lizzy[77] wearable computer. Brad Rhodes would later go on to complete his dissertation on the Remembrance Agent and just-in-time information retrieval in May of 2000[61], by which time I had switched advisors and become a full time wearables researcher myself.

I started using a PalmPilot Personal Digital Assistant (PDA) organizer shortly after I got to the Media Lab in '97 — it was given to me for use as a research tool by my then advisor, John Maeda. Prior to using the Palm, my personal experience with memory support technology was mostly confined to the use of paper organizers, which I depended on during my undergraduate and previous three years of graduate education.

I immediately recognized the value of the Palm. It provided capabilities similar to those of paper organizer, but it was searchable and provided alarms for calendar appointments. The original Palm PDA was by no means perfect: It crashed regularly, requiring frequent backup. It was slightly too large for my pockets. The screen was fragile, and the stylus was easily lost. However, the added features of a clock, an alarm, and searchable interface more than made up for these deficiencies. Soon I was almost completely dependent on the Palm to keep track of the logistical details of my life.

The same digital organizer capabilities were provided by the Lizzy wearable that I

started using in the summer of '98, albeit in the context of a much more general and powerful application framework. The Lizzy presented an “always on, always present” interaction style using the head-mounted display that mostly obviated the need for the Palm’s audible alarms. Further, the Lizzy provided the ability to extend the simple time-awareness provided by the PDA’s clock with the ability to sense other aspects of the user’s state and environment; Brad Rhodes used the Lizzy as the platform for his location and textually-context-aware Remembrance Agent research. However, the human factors of the Lizzy were much more awkward than the Palm PDA. The Lizzy was a more powerful tool, but more difficult to use and less compatible with my daily activities.

When I became a full time wearables researcher in '99, I wanted to bridge the usability and functionality gap between the portable and mobile technologies in common use in the late 90’s (PDAs and cell phones) and the more powerful, exotic research systems being investigated by wearable computing researchers at the Media Lab. Memory support provided an ideal framework for this research agenda, since it allowed me to investigate new technologies and interaction techniques within the context of the nearly universal problem of supporting human memory.

The focus of this thesis is wearable computing for memory support, and in particular a proactive, context-aware[75] just-in-time memory support system employing a novel interaction technique — subliminal cuing — to minimize the demands on the user’s attention. This research agenda evolved over time, and represents a synthesis of several areas of major interest:

1. Platforms and architectures for wearable computing systems.
2. Wearable context awareness: using body-worn resources to sense and classify in real-time important aspects of the user’s state, environment, or activity.
3. Wearable human-computer interaction issues, especially those related to distraction, divided attention, and task interference.
4. The science of memory and memory support technology.

The Memory Glasses is the synthesis of this research agenda, combining all of these in-

terests and more besides. Along the way, I also worked on applications in biomedical monitoring for soldier medicine and Parkinson's medication state monitoring, built a prototype wearable for the Swedish military, helped colleagues map social networks, and lead the MIThril wearable computing research effort at the MIT Media Lab.

## 1.1 Organization of the Thesis

This thesis is organized by research topic:

Chapter 2 frames the memory support problem, and puts the Memory Glasses approach into the context of existing memory support technology and current research. It introduces important terms used later in this thesis, such as context awareness, and makes the case for why a new approach to memory support is needed.

Chapter 3 describes the evolution of the MIThril hardware architecture developed for the Memory Glasses project, and explains how the requirements of a light-weight, robust, always-present memory support informed its development. The MIThril hardware architecture has proved an important tool for other research projects at the Media Lab and elsewhere, and some of these are mentioned.

Chapter 4 describes the evolution of the MIThril software architecture, and how it was shaped by the requirements of the Memory Glasses project. The emphasis in this chapter is on an interprocess-communications system and modular signal processing and real-time classification system developed to support the real-time classification of sensor data for context awareness. Much effort was spent on the problem of real-time user activity classification, and this problem and the solution we developed (the MIThril Real-Time Context Engine) is described in some detail. These tools have proved useful for biomedical and physiology research, and some of these applications are described.

Chapter 5 frames the memory support human-computer interaction problem, and describes some of the problems attendant with conventional direct-manipulation interfaces in a wearable context. To address these problems, I extend conventional human-computer interaction models to take into account the divided attention and mediation effects associated with wearable computing. Based on this analysis I propose five principles for low-attention

wearable human-computer interaction design, and then go on to describe prototype Memory Glasses implementations and a graphical windowing environment that embody these interaction principles. I summarize my research in just-in-time cuing for wearable memory support.

In Chapter 6 I state what I believe to be my primary intellectual contributions in this doctoral research. In Chapter 7 I summarize my conclusions and describe future research directions.

In Appendix A, I provide a survey of the wearable HCI and cognitive science literature as it relates to the low-attention wearable interaction problems addressed in this thesis. In Appendix B I describe my cuing research in detail.

# Chapter 2

## Memory Support

### 2.1 Introduction

In this chapter I attempt to frame the memory support problem in a systematic way that allows me to talk about paper organizers, PDAs, and the Memory Glasses (as well as other instances of memory support technology) in a uniform way.

### 2.2 Remembering and Forgetting

It would be hard to overstate the importance of memory in the operation of human thought and daily life. Our memories are a very important part of what make us who we are. Without the effective operation of learning and memory, little can be accomplished in any sphere of human endeavor.

Information theory posits a fundamental equivalence between communication and memory[69]; Transmitting a message in space is equivalent to transmitting a message in time<sup>1</sup>. Our memories are, in some sense, messages from our past selves that are received in the present. Or, in the many instances that human memory fails us, these messages are not received, or are received in a garbled form. Failures of memory, however infrequent or common they may be, have real-world consequences, hence the need for memory support.

---

<sup>1</sup>In fact, the UNIVAC I used acoustic mercury delay lines as principle data storage. According to on-line documentation, 10 11-decimal-digit data words could be stored in a single 55.75 inch long “register.” (<http://ed-thelen.org/comp-hist/vs-univac-mercury-memory.html>)

The operation of human memory is complex, and memory can fail in a number of ways. Schacter[66] has identified seven important classes of human memory failure, which he terms the “seven sins of memory.” These are:

**Transience** decreasing accessibility over time

**Absent-mindedness** lapses of attention; forgetting to do things

**Blocking** temporary inaccessibility of stored information

**Misattribution** attributing memories to incorrect source; false recognition

**Suggestibility** implanted memories

**Bias** retrospective distortions produced by current knowledge and beliefs

**Persistence** unwanted recollections that people cannot forget.

Schacter argues that these apparent defects of memory are actually the expression of important memory features without which normal thought and experience would be impossible.

If Schacter is correct, then in some sense our memories must be unreliable in order to be functional. This is unfortunate, but by understanding better the ways in which we remember (and forget) we can work to improve the function of our own memories and to design systems that compensate for the failings of human memory. As Norman[48] points out, designing systems that require the user to keep large amounts of operational knowledge or state in memory is a recipe for bad design. However, as any user of office telephones (or Unix command lines) can tell you, such designs abound, and in some instances are the only reasonable solution to a complex problem<sup>2</sup>.

## 2.3 Memory Support Technology

Memory support is a common and important application of information technology, from notes scribbled on bits of paper and strings tied around fingers to PDAs and complex record keeping systems. The complexity of modern life combined with the fallibility of human

---

<sup>2</sup>To quote Alan Kay, “Easy things should be easy; hard things should be possible.”

memory make memory support an increasingly important part of daily life. As a colleague of mine once put it, “I’ve been dependent on PDAs my entire life. Too bad they only recently became available.”

### **2.3.1 Idealized Memory Support**

The ideal memory support technology would be a seamless extension of endogenous memory. It would rectify all of the seven sins of memory identified by Schacter, would impose no additional physical or cognitive burdens on the user, and would be indistinguishable from the operation of a flawless endogenous memory to both the internal and external observer<sup>3</sup>. The realization of such a system is clearly beyond the scope of present technology, requiring as it does not only a perfect understanding of the functional neurology of memory and (presumably) a direct brain interface constructed at the nano scale, but also the ability to perfectly predict what the user will need to remember in the future, so as to guide her attention appropriately to avoid the sin of absentmindedness. (In the absence of recognition, a comprehensive capture of the user’s perceptions before the process of attention-directed attenuation might also suffice.)

Although unattainable, the formulation of the idealized system is instructive in what it leaves out: namely, any additional effort or thought on the part of the user to encode and recall memories, and any additional distraction or cognitive load imposed by use.

### **2.3.2 Memory Aids vs. Memory Surrogates**

For the purposes of this discussion, I divide memory support technology into two classes: memory aids, that are intended to improve the function of one’s own endogenous memory, and memory surrogates, that record information for later delivery, obviating the need to remember at least some information. These classes are not exclusive — taking notes on a

---

<sup>3</sup>Brad Rhodes’s, a well known wearables researcher and author of the Remembrance Agent, had the following comment on this idealized system; If he were the user he would want to know when and to what degree the remembered data was coming from the exogenous memory system vs. his endogenous one. My concern is that this additional meta-information, if present, would present a significant distraction. Further, since our hypothetical system is infallible and would correct any flaws in the endogenous memory, there really isn’t any need to know where the data is coming from — it will always be correct.

lecture may allow one to access the recorded information later, thus acting as a memory surrogate. However, the process of taking the notes may itself be helpful for learning the material, thus acting as a memory aid. If the notes are thrown away after the lecture, then the notes (or rather, the act of creating them) may function as a pure memory aid. If one attends the lecture and takes no notes, but later studies a transcript, that transcript may function as a pure memory surrogate.

Memory-enhancing drugs are an example of a technology that is intended to operate as “pure” a memory aid, for example the herb Ginkgo Biloba and the neuropeptide Vasopressin. (See Section 2.5.3 for more information.) Likewise, flash cards are used by students everywhere to aid in memorization drills. A range of mnemonic devices have been developed over the centuries, ranging from simple grouping to more complex methods such as the *method of loci*, (see Section 2.5.1 below).

However, most contemporary memory support information technology is designed to function primarily as a memory surrogate. We do not, as a rule, take notes on lectures and then throw the notes away, nor do we enter information into our PDAs with the intention of erasing it without ever looking at it again. The paper, like the PDA, is designed to retain the information for latter access.

### **2.3.3 Memory Support vs. Sharing Experience**

Memory surrogate information technology involves the external<sup>4</sup> recording of information for later retrieval. For example, I may take notes on a lecture and then re-read them later to refresh my memory. However, once written, other people could potentially read and benefit from them<sup>5</sup>. Once an experience has been recorded in some externally accessible form, it can be shared with others, allowing for a kind of “remembering other people’s memories.” The close connection between information technology for memory support and the sharing of experience is reflected in the organization of current research conferences on this subject; For instance, I was recently reviewing submissions for Pervasive Computing

---

<sup>4</sup>An implantable memory surrogate is well within the realm of possibility even now, so the designation “external” really means “non endogenous, potentially accessible by others.”

<sup>5</sup>Assuming I choose to share them, that the notes are not completely unintelligible or misleading, *etc.*

2004's workshop on Memory and Sharing of Experience.

The organizing theme for this thesis is personal memory support, by which I mean supporting an individual's ability to remember things they have seen, heard, or otherwise experienced. Much of the memory support technology described here also has applications to experience sharing, and more generally to groupware and computer supported collaborative work. However, the recorded form of an experience is not the same thing as the experience itself, *e.g.* the experience of reading my lecture notes is not the same experience as attending the lecture.

We can take advantage of this difference to reduce the amount of information captured (or presented) for the support of personal memory. An associative theory of memory suggests that effective memory support is possible by presenting the user with cues rather than complete information. (A *cue* being summary, partial, or related material that is simpler than or different from a more complete representation of the information to be remember.) The critical point being that the amount of information that is captured and presented to the user for memory support may be considerably less than the amount of information required to effectively share an experience. The use of cues is explored in more detail in Chapter 5.

### **2.3.4 The Dangers of Memory Support**

The use of paper organizers, PDAs, and other memory surrogate technologies comes at a cost. Without exercise, memory skills atrophy[66]. By relying on memory surrogates we risk forgetting (or never learning) important information.

By way of anecdote, I used to memorize phone numbers almost effortlessly until I began to carry a cell phone with number storage. Today I have close friends who's phone numbers I don't know, because my cell phone remembers them for me. Occasionally I misplace my phone, or fail to charge it, and I loose access to the stored numbers. Perhaps worse, I am no longer as good at memorizing phone numbers as I used to be, which gives me even more of an excuse to rely on the phone's memory rather than my own.

This problem isn't specific to cell phones, or memory surrogates in general. Once relied upon, the failure of any memory support technology could cause problems, leaving us less

capable than before.

The failure of a memory aid that supports memorization (encoding) could cause difficulties in learning new material. The failure of a memory aid that supports recall could similarly cause difficulty in accessing previously learned material.

And even when a memory support functions as intended, there can be negative consequences for the performance of the task that is being supported. This phenomenon has been documented by Ockerman[50] and is discussed in more detail in Chapter A.

As designers of memory support technology, we must be cognizant of the costs as well as the benefits of the technology we create.

## **2.4 Organizers**

One model for memory support is the organizer, in which the user records and access structured information. The premise of this model is that items of information are structured in a meaningful way to facilitate later retrieval. This model is important, because it reflects the way in which people use a wide range of current memory support technology, including paper organizers, diaries, and PDAs.

### **2.4.1 Organizers and Context**

Organizer applications provide a contextual structure, allowing items of information to be associated with a particular context of relevance. For instance, a calendar organizer allows the user to associate items with particular dates or times of relevance — a time context. Address book organizers structure information by name, or social context. Charts and maps allow for notations based on location context. Checklists allow information to be structured by action context, such as procedures for aircraft inspection<sup>6</sup>.

---

<sup>6</sup>Charts and checklists rarely start out blank, but are often more useful if they support annotation. For the purposes of this discussion, I am mostly interested in the way location and action contexts can be used in organizers to structure the user's records.

## 2.4.2 Proactive vs.. Passive Organizers

One of the important functional differences between a PDA calendar application and a paper calendar is that a PDA has a clock, and is capable of getting the user's attention at particular times and delivering specific appointment-related information. This proactive calendar-plus-alarm-clock is qualitatively different than the passive calendar provided by the paper organizer. Both the PDA calendar and the paper calendar can be used to record appointments, but the paper calendar cannot act to deliver the reminder at the appointed time.

In order for the PDA calendar to be proactive, it needs two things. First, it has to know the date and time. Second, it must take some meaningful action to support the user's recall. The first capability is provided by an internal clock, the second is typically provided by an audio- or vibration-based alarm intended to get the user's attention, combined with a small visual display to present the specifics. These features are instances of the more general classes of *context awareness* and *proactive reminder delivery*.

## 2.4.3 Context Awareness

The clock in the PDA calendar provides the application an awareness of time. Because the PDA calendar knows what time it is, actions can be triggered based on this knowledge. Thus, the PDA calendar is an *time-aware* application. Importantly, the PDA doesn't have to ask the user for the current time — an internal clock provides this information without the user's intervention<sup>7</sup>. Time awareness is one type of context awareness.

*Context awareness* is the use of non-explicit user input. This input is typically gathered through the use of sensors<sup>8</sup> and real-time analysis of sensor data.[52, 4, 78, 75]. Context awareness can be used to support a range of applications from group coordination and situation awareness to biomedical monitoring[14], some of which are discussed later in this thesis. Although the use of context awareness can reduce the need for explicit user

---

<sup>7</sup>Typically the user must set, or reset, the clock from time to time, but this interaction is occasional, and not related to the triggering of a particular action. For the purposes of this discussion I assume every memory support technology has some fixed "set up" cost which we can, for the moment, ignore.

<sup>8</sup>One can regard a clock as a kind of "time sensor" that measures and counts the oscillations of a resonator.

input for some applications, it comes at a cost. The cost of adding context-awareness to an application ranges from the trivial (most digital devices include a real-time clock) to the significant, such as the computational costs associated with real-time computer vision. The engineering tradeoffs involved in context aware applications are discussed later in Chapters 3 and 4.

Time awareness is clearly an important feature in a time-context-structured organizer, but it is not the only type of “awareness” possible. Additional sensors and data-analysis can automatically provide the organizer application with such useful information as the location of the user, to whom they are talking, whether they are sitting, walking, or driving a car, etc. Thus “time awareness,” which allows actions to be taken conditioned on date and time of day, can be extended with “location awareness,” which allows actions to be conditioned on location, “activity awareness,” which allows actions to be conditioned on activity, and “social awareness,” a special case of activity awareness that allows actions to be conditioned on the user’s social state (with whom they are interacting, in what modalities, *etc.*)

### **Other Types of Context**

Other types of awareness are possible, such as awareness of the state of remote events (the stock market, the availability of colleagues, *etc.*), devices (is the printer low on toner?), *etc.*. This type of awareness may be useful for triggering the delivery of information for a variety of context-aware applications. However, if the context is truly remote and cannot be directly perceived by the user, then any action taken by an application based on this remote context will have some aspect of communication (bringing new information to the attention of the user) in addition to supporting the user’s memory of something they have already directly experienced. Sensing such context might be quite useful in a practical reminder application (*e.g.* When Alice gets out of her meeting, remind me to ask her about Bob’s report.) but the communication aspect of this type of reminder delivery breaks the “pure” individual memory support model that is the primary focus of this thesis. In addition, delivering such reminders would necessarily require getting the user’s attention (again, the “new information” aspect of communication) and thus would not be compatible with the subliminal cuing strategies explored later in this thesis.

## 2.4.4 Records

An *organizer record* is an item of information with an associated context, such as an appointment recorded in a calendar organizer. The record may have several types of context associated with it, such as time, location, and social context (with whom the meeting is to take place). However, only certain types of context are meaningful to the organization of the records. In the case of a paper calendar, records are organized by time, making time the *distinguished context*. If appointments are also cross indexed with an address book, then both time and social context would be distinguished contexts.

### Creating Records

In order to create an organizer record, the user must specify, at a minimum, a distinguished context, zero or more additional contexts, and a “body” of additional information that may be empty. For example, an appointment in a calendar might consist of a time context (distinguished), a location context, and a social context. Additional information, such as a meeting agenda, or notes taken at the meeting itself, might be captured in the body.

The way in which the user records and edits this information depends on the application. In the case of a paper organizer, the information is simply noted on the appropriate part of the appropriate page corresponding to the distinguished time context, and many PDAs borrow heavily from this interaction model.

### Viewing and Searching Records

In a paper organizer, there is generally only one distinguished context for an organizer, and no other way to view or search records. For example, in a paper calendar all records are organized by time context, regardless of other content. In order to find an appointment with Alice, the user must search by date. One can't simply recognize the presentation of the data by social context to show all of appointments with Alice.

A digital organizer may support more than one distinguished context. For example, the calendar and address book applications in a PDA-style organizer could employ records with both time and social distinguished contexts. This would allow the user to browse or

search the address book organized by name (social context) and see a cross-referenced list of calendar appointments, or vice versa.

In order to make this efficient, the address-book organizer and the calendar organizer must have the same understanding of distinguished context, *i.e.* the address book must understand that Alice in the calendar appointment and Alice in the contacts list are the same person. (In commercially available PDA organizers with which the author is familiar, this functionality is approximated by allowing searches for records based on keywords. Thus a search for “Alice” would return hits in both the address book and the calendar.) A full featured digital organizer with multiple distinguished contexts (such as time context, social context, location context, and activity context) should allow the viewing of organizer records structured by any combination of contexts desired.

This operation would best supported by storing reminder records in a relational database, with the contents structured by distinguished context. Such a contextual record database would allow fast query by context, and make both viewing and the proactive delivery of record information fast and efficient. This scheme would also support “flat file”-style keyword searches for information in record bodies.

## **2.4.5 Reminders**

The ability to view and search records cross-referenced by distinguished context is a powerful feature, but awareness of context can transform a passive information storage and retrieval system into a proactive memory support. Conceptually this is an extension of the explicit user-initiated search function described above made automatic and continuous, with the user’s current context as the search term. (This type of search function is the basis of Brad Rhodes’ Remembrance Agent[63].)

Further, the body of the record is extended with actions to be triggered when this search gets a “hit.” These actions are analogous to callback functions in a conventional event-driven user interface. Actions range from the simple, such as raising an alarm, to the complex, such as executing arbitrary programming code.

For example, a PDA-based calendar application typically allows one to associate alarms

with records. These alarms are triggered based on the record's distinguished context: time.

A *reminder* is a digital organizer record with an associated action to be triggered based on context. Although most existing digital organizer applications are time-aware only, in principle any type of context could be used. Likewise, most existing digital organizers can use context to trigger a limited range of actions, such as raising alarms, but the triggered action can, in principle, be anything intended to support the user's memory – including actions that suppress other actions or modify the way in which the system interacts with the user.

Different types of reminder actions are discussed in Chapter 5.

## **2.4.6 The Advantages of Context Awareness**

There are several reasons to generalize time-aware organizers to operate with other types of context:

### **Flexibility**

One advantage of adding more general context-awareness to an organizer is the flexibility it provides. Reminders are often relevant to situations or interactions that occur independent of schedule. One might want a reminder delivered in a particular social or location context independent of time, such as “The next time I see Sandy, remind me to ask about the NSF grant proposal,” or “The next time I'm at a grocery store, remind me to buy eggs.”

The addition of activity-awareness allows an organizer to act as a context-aware task guidance system[50], providing task-specific memory support. (This is explored in more detail in Chapter 5.)

### **Reduced Distraction**

Another advantage of context awareness in a memory aid is that it can be used to recognize when certain reminders are unnecessary. For example, a time-based meeting reminder need not be delivered if the location-aware organizer can sense that the user is already at the meeting, or traveling to it with a likely on-time arrival.

Context awareness can be used to suppress reminders for other reasons as well. An activity-aware or socially-aware reminder system could suppress reminders based on what the user is doing (*e.g.* “Don’t interrupt me while I’m driving”), to whom the user is talking, etc.

By reducing the number of unnecessary reminders, the value to the user per reminder goes up, and the potential distraction goes down. This is particularly important for cognitively impaired users who may have a diminished capacity for handling interruptions and switching between tasks.

## **2.5 Memory Support in Context**

Earlier in this chapter I introduce a framework and terminology for describing memory support technology, making occasional references to existing technology where illustrative.

In this section I use the terminology previously established to describe a number of existing memory support technologies and the Memory Glasses. This list is not intended to be comprehensive, but rather to establish the relationship between the Memory Glasses and other types of personal memory support. This listing is roughly chronological in ordering.

### **2.5.1 Mnemonic Devices**

Arguably the oldest memory support technology, mnemonic devices have been used for thousands of years to support human memory. They are still taught today to college students and others as study techniques. Mnemonic devices are examples of pure memory aids, as they function by improving the operation of endogenous human memory.

What follows is a partial listing of mnemonic devices quoted from a list provided by the Utah State Academic Resources Center:

**Grouping** Classify lists on the basis of some common characteristic. Remembering the key element of the group is a key to remembering all the items. An example would be grouping minerals by metals or stones.

**Rhymes** Set what you need to remember to a common rhyme. One example is: "In four-teen hundred and ninety-two Columbus sailed the ocean blue."

**Acronyms** Acrostics - Use acronyms or acrostics to help you remember lists of words. The first letter from each word in a list forms a key word, name, or sentence. One example is in music, "every good boy does fine" are the lines on the treble clef. FACE stands for the spaces on the treble clef. Acronyms are now part of our language. Consider IBM, AT&T, CIA, MGM, MIT, and FBI.

**Visual Association** Association involves linking two ideas. When you are memorizing lists of words, you can link words by using images. The "Peg" system allows you to remember sequences of ten unrelated items in the appropriate order. It requires you to first remember 10 "peg words:" one = bun, two = shoe, three = tree, four = door, five = hive, six = sticks, seven = heaven, eight = gate, nine = vine, ten = hen. If you have ten words that you need to remember, you visualize each word interacting with the "peg word" that you already know.

**Loci** Greek and Roman orators who had to remember long speeches used the method of "loci" to trigger their memories. In using the method of loci, visualize a room or route that is familiar to you. Then place each item you wish to remember in a location along that route and "pick it up" as you take a mental walk around that room.

This is especially helpful in giving speeches or having to remember lists of important points in a correct sequence. For example, if you wish to memorize the presidents of the United States in order, you might deposit a dramatic image of each president in strategic location around your house.

A great deal more could be said about mnemotechnics, though such is not the focus of this thesis. Readers interested in sophisticated mnemonic techniques may be interested in Matteo Ricci[74], a 16th century Jesuit priest famous for teaching the Chinese to construct "memory palaces." A scholarly overview of the history of memory (including Ricci and many others) can be found in Yates' "The Art of Memory[91]."

### **2.5.2 Paper Organizers**

Paper address books, calendars, charts, *etc.*, are all instances of organizers, since they allow the structuring of information by appropriate context for later retrieval. Organizers are primarily memory surrogate technologies, since “they remember so that you don’t have to.”

### **2.5.3 Memory Drugs**

Memory drugs are a relatively new development in memory support technology. They work by enhancing the operation of endogenous memory at a physiological, biochemical level. The herb Ginkgo Biloba has been studied (and marketed) as a memory enhancing “nutritional supplement[72],” though the scientific evidence for its effectiveness is inconclusive. The neuropeptide Vasopressin has 20 years of scientific evidence for its memory-enhancing effects, though the mechanism by which it operates is unclear[65]. Recent advances in the understanding of the neuroscience of memory have made memory-enhancing drugs an area of active research[39].

### **2.5.4 PDAs**

The Personal Digital Assistant, or PDA, can be traced back to at least 1984, when Psion introduced the Pocket Organizer and the Organizer II in 1986. These devices were simple “pocket computers” that employed a small screen and keyboard, and provided basic organizer features in the terminology of this thesis. Although somewhat successful, the early Psion organizers were not greatly popular or widely known outside of the UK (the more popular Series 3, released in 1991, eventually sold 100,000 units)[38].

The famous (and famously unsuccessful) Apple Newton was the first widely-known PDA-class device to be introduced to the American market in 1993. It was followed by a string of failures from other major technology companies including Tandy, Caseo, and Sharp. Palm, a software company that had written software for many of the earlier PDAs (including the Newton, the Zoomer, and others) became frustrated with the lack of focus of the PDAs of the early 90s and developed their own hardware. They partnered with U.S.

Robotics (a modem company) to manufacture the device. The result was the the first widely successful PDA, the PalmPilot, which shipped to stores in early 1996[58].

The original PalmPilot was followed by a succession of refined models and imitators, but the core features of small size, simplicity, and organizer functionality have remained largely unchanged.

### **2.5.5 Forget-me-not**

The Forget-me-not system was developed by Mik Lamming and Mike Flynn[36] at the Rank XEROX Research Centre as an innovative solution to the problem of supporting human episodic memory. Employing a combination of wearable tag/PDA devices (the ParcTab[84]) and instrumented infrastructure, the Forget-me-not system was designed to capture salient events in the lives of the users and record these in a central biography database. The Forget-me-not system captured room-level location context, social context (who was in a room), workstation context (email exchange, printing, and file sharing), and internal telephone activity, all structured chronologically.

This database could then be queried by context to generate filtered episodic biographical summaries, displayed in iconographic form on the ParcTab. The goal of this capture and biographic summarization was to provide the user episodic summaries of salient events. For example, the system would allow the user to frame queries such as “Who did I call after talking to Bob in my office last Wednesday?” Or, “Was Alice at the meeting with Bob and Cathy?”

To the extent that the user queries the system regarding events in the user’s personal biography<sup>9</sup> the system behaves as a context-aware organizer. However, rather than using context to trigger the delivery of information, this system automatically captures context to support subsequent manual query.

---

<sup>9</sup>The Forget-me-not system as described in [36] uses a centralized database for the capture of biographic information. Thus, this database could support complex queries relating to the biographies of any or all users. Such a system would be ideal for many types of social networks research, though as the authors acknowledge the centralization of this data raises clear privacy concerns.

## 2.5.6 The Wearable Remembrance Agent

The Wearable Remembrance Agent is a location-aware and textually-context-aware document search and retrieval system developed by Brad Rhodes[60, 61] as an extension of the Remembrance Agent desktop textual-context-aware document retrieval system[62]. The Wearable Remembrance Agent, implemented on a Lizzy[77] with Locust[76] tag-based indoor location awareness provides the user with a continuous background search for related information based on the user's location and textual context (the local contents of the text document being viewed or edited). The wearable RA provided some organizer-like features, since documents (such as class notes) were automatically tagged with time and location context. In addition, the wearable RA allowed the user to manually tag and search for documents by social context and textual context. The RA software was implemented as an extension of the Unix emacs text editor.

Less focused than a classic organizer application, the RA primarily served as a document retrieval system. These documents could include familiar "record of experience" information, such as the user's own notes and email, or equivalently large bodies of previously unfamiliar text, such as a publication abstract database. The user could choose what information to index, and thus to what degree the system operated as a memory surrogate rather than a research tool<sup>10</sup>.

## 2.5.7 The Memory Glasses

The Memory Glasses is a proactive, context-aware organizer that allows the user to schedule reminders for delivery by time, location, social, and action context, and is capable of a range of actions for reminder delivery. A number of Memory Glasses prototypes have been implemented using the MITHril hardware platform[15, 64], which is described in Chapter 4.

Like the Wearable RA, the Memory Glasses allows documents to be associated with time, location, and social contexts, but the Memory Glasses is capable of automatically

---

<sup>10</sup>I have found the RA to be quite useful in indexing email (communications from others) and notes (communications from myself) in both a wearable and desktop context. The usefulness of this hybrid communications/memory support application once again points out the fundamental similarity of communications and memory.

classifying all of these contexts using the MIThril Real-Time Context Engine[14]. This same context-classification capability allows for the logging of user context to create Forget-me-not style biographies of location, social, and action context that do not depend on instrumented infrastructure or a centralized server. The MIThril Real-Time Context Engine and associated software tools are described in Chapter 4

Perhaps more importantly, the Memory Glasses directly addresses the divided attention and over-reliance problems posed by conventional memory surrogate system by allowing the use of subliminal as well as overt cues for reminder delivery[13]. Human-computer interaction for memory support is explored in Chapter 5 and Appendix B.



# Chapter 3

## MIThril Hardware

### 3.1 Origins: Lizzy

My first research wearable was a Lizzy[77], which I constructed in the summer of 1998. The Lizzy was a PC104 board stack packaged in a ballistic nylon satchel with a strap that was typically worn “bandoleer” style across the chest. The head mounted display, a Reflection Technology Private Eye, was mounted on a pair of plastic safety glasses that clipped to the front of the strap when not in use. The Twiddler chording keyboard was held in the left hand or attached by a Velcro strap to the bottom of the satchel.

The Lizzy was important because it was one of the first general-purpose wearable computing research platforms, and it my first experience with the daily use of a head-mounted display and one-handed chording keyboard for typing. However, I had difficulty with human factors of the satchel-based design, and by the time I became a full-time wearable computing researcher in 1999 I was interested in developing a research platform with improved ergonomics and more flexible connectivity. This led to the MIThril project.

### 3.2 Motivations for MIThril

From the beginning of the project, the MIThril hardware platform was designed to support the Memory Glasses application. The close relationship between the requirements of the Memory Glasses application and the capabilities of the MIThril hardware platform resulted



Figure 3-1: Members of the original wearable computing group wearing Lizzy-style wearables. Picture by Sam Ogden.

in a co-evolution of the platform and application, each influencing the other.

The MIThril hardware platform is important to this thesis, because it provided the vehicle to explore the software, hardware, human factors, and cognitive science that shaped the development of the Memory Glasses. In addition, MIThril 2000 (and in particular the video driver hardware) provided a nearly ideal research tool for the subliminal visual cuing research.

I proposed the MIThril project as framework to focus the development efforts of the wearables group on the creation a wearables research platform that would be truly wearable. My belief was that until we had a research platform that was really unobtrusive and comfortable, strangeness and discomfort would dominate the the popular perception of our research and relegate its plausible applications to the military and industrial sphere.

There was an immediate, practical benefit to improving the wearability of our research gear; It would be more comfortable for research subjects. Since I was interested in doing memory support research with populations ranging from undergraduates to the elderly, having a platform that was comfortable and compatible with a range of conventional clothing and every-day activities was very desirable.

I also had a personal motivation for the development of the new platform; I wanted a

wearable that *I* could feel comfortable using on a daily basis. I have great respect for the heroic efforts of Thad Starner and Steve Mann to integrate the early generation of prototyped research wearables into their daily lives, but did not want to repeat that experiment. My goal was a wearable that was as close to invisible as possible.

### **3.3 Between Lizzy and MITHril**

Before beginning work on the “sensor vest” project that became MITHril, members of the wearables group experimented with other clothing-integrated designs. At the same time, other groups were continuing to develop more conventionally packaged systems, including the CMU “Spot” wearable[17] and the Charmed CharmIt, a commercialization of the Lizzy concept based on a fast Transmeta CPU and extended connectivity options.

#### **3.3.1 Smart Vest**

Steve Schwartz’s work on the Smart Vest showed that non-trivial computing and wireless networking capability could be unobtrusively and comfortably integrated into a clothing-based package[67]. Unfortunately, this system lacked any means for directly interacting with the user. However, it provided the inspiration for using small-footprint Linux single-board computers in a vest form-factor “chassis.”

#### **3.3.2 Sony Vaio Picturebook Vest**

My first experiment with a clothing integrated wearable was based on the Sony Vaio Picturebook, a small windows 98 notebook computer available in ’99. I installed Linux on this machine, which I then packaged into the back of a leather vest with a MicroOptical display and Twiddler keyboard. This was my daily-use research wearable by the time we went to ISWC 99.

This system was functional, but the human factors were awkward; Although it was relatively unobtrusive and did not interfere with most ambulatory activities, the notebook computer was positioned in the small of the back. This made sitting uncomfortable. In

addition, the Vaio posed limited IO options, requiring a PCMCIA card for either wireless networking or serial IO. Since it supported only one card, both were not possible at the same time.

### **3.3.3 Cell Computing Vest**

To remedy some of the connectivity limitations of the Vaio vest, I experimented with a Cell Computing Plug-and-run Pentium III-based small form factor computer module. This system was overall somewhat thicker than the Vaio, but provided two PCMCIA, one CF, and dual serial ports, as well as support for a laptop IDE hard drive and a VGA out. I packaged the Cell Computing hardware and laptop hard drive in an ABS plastic sandwich that provided additional mechanical strength and protection. This was then packaged into the back of the same leather vest that I'd used for the Vaio, along with batteries, an integrated power bus wiring harness, a tag reader, a CDPD cell modem and a MicroOptical display driver. To make the package more comfortable, I added sculpted foam cushioning.

This configuration provided good connectivity and relatively high-performance computing. Also, the integrated power rail made for a single set of batteries, resulting in lower total battery-weight[42]. Unfortunately, the Cell Computing core burned a substantial amount of power, and the system required almost three pounds of batteries (two Sony InfoLithium NPF950s) to get better than four hours of uptime.

This system worked, but I felt that it was too heavy and bulky (as well as being too hot) to meet my ergonomic requirements.

## **3.4 MIThril 2000**

The MIThril 2000 hardware platform project grew out of my dissatisfaction with commercially available wearable hardware (such as those made by Xybertnaut[30]) and our existing prototype wearable research platforms (the Lizzy[77] and those mentioned above). Each had its merits, but none seemed appropriate for the Memory Glasses research application.

My primary concern with MIThril's immediate predecessor was the weight, bulk and

## Mithril FUNCTIONAL DIAGRAM

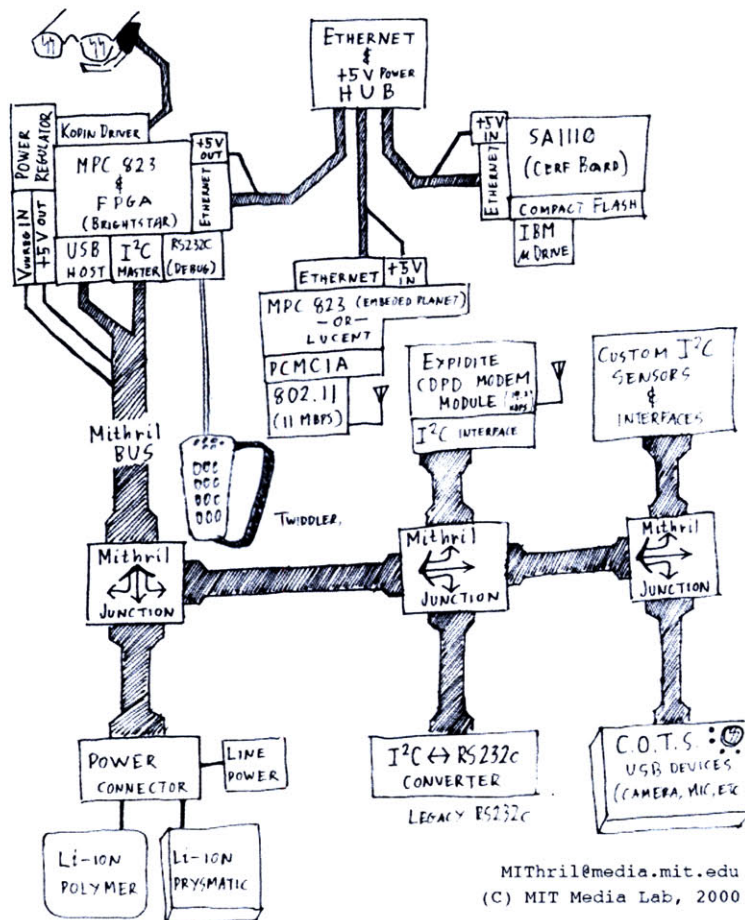


Figure 3-2: MITHril functional schematic. This drawing was done early in the development of MITHril, and there are some minor differences between this schematic and the more modern MITHril 2000 prototype shown in Figure 3-3.

thermal discomfort<sup>1</sup>. Addressing these physical human factors issues in a more elegant, flexible framework became my primary goal in developing the new platform. In addition, the system would be clothing integrated and comfortable enough that it could be used by ordinary people in real social settings. Such a system would need to be comfortable, adaptable, unobtrusive, and easy to work with.

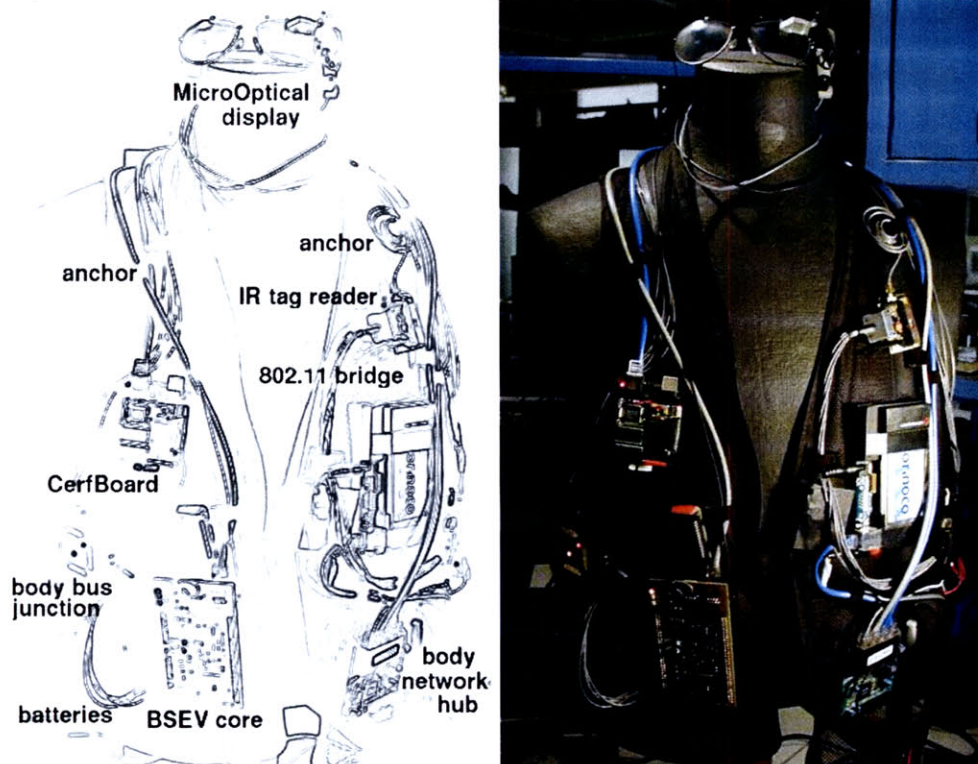


Figure 3-3: One possible MITHril 2000 configuration.

### 3.4.1 MITHril Overview

The MITHril 2000 hardware platform is a clothing-integrated, modular, flexible platform for prototyping wearable computing research applications. A full MITHril system (like the one shown in Figure 3-3) combines a body-worn cluster of networked Linux computers, sensors, high-bandwidth wireless networking, mass storage and battery power in a package that weighs less than four pounds and runs for six to eight hours on a single set of batteries. This configuration provides a light-weight head-mounted display with custom-engineered driver electronics and the ability to connect a chording keyboard or Palm folding keyboard as the primary symbolic input device.

The MITHril 2000 system allows for a wide range of sensors and peripherals to be connected to a on-body network of computing and networking resources. MITHril 2000 supports a range of configurations, from a multi-node Linux-based body-worn computer network with head mounted display to simple microcontroller-based data acquisition sys-

<sup>1</sup>Of course there were other problems as well, and the aesthetic inelegance of the engineering was both-  
ersome, but these were manageable difficulties by comparison.

tem.

From the beginning, MITHril was shaped by the requirements of the Memory Glasses application. However, MITHril technology has been used in a number of Media Lab projects, including the thesis research of Tanzeem Choudhury (the “Shortcuts” social networks mapping project) and Josh Weaver (the Parkinson’s medication state classification project). It has also provided the foundation for collaboration with other research institutions, including the Rochester Center for Future Health and most recently the Army Natick Labs. MITHril was chosen by media lab sponsor SAAB as the foundation of the SAAB Aerospace/SAAB Avionics’ wearable situation awareness and command and control system — the WISE project, which was presented by Cecilia Lundin at the Spring 2003 TTT meeting.

### **3.4.2 Memory Glasses Platform Requirements**

Beyond computing and wireless networking, we needed a flexible platform for research in sensing the user’s context. Further we needed a platform that would allow us to interact with the user in unobtrusive ways. Finally, the physical human factors and wearability were critical. Based on our analysis, we drew up a the following list of requirements for the Memory Glasses research platform:

1. **Wearability:** As much as possible, the hardware should disappear into the user’s ordinary clothing, leaving only a minimal interface. Reliability and ergonomics must be high, weight must be low, and uptime must be long. Ideally, the user puts the system on in the morning, takes it off in the evening, and completely forgets about it in between except when actively using the functionality.
2. **Flexibility:** The widest range of physical and functional configurations should be accommodated by the design. Reconfiguring the system should be a simple matter of reconnecting components, not resoldering boards.
3. **Sensing and Bandwidth:** A wide range of sensors and protocols should be supported, ranging from cheap off-the-shelf hardware to custom microcontroller-based devices.

The on-body sensing bus should provide sufficient bandwidth and flexibility for the simultaneous use of heterogeneous sensors, such as accelerometers, ECG/EMG sensors, microphones, and cameras<sup>2</sup>.

4. Interaction: The system should support a range of unobtrusive peripherals, including a light-weight head-mounted display, audio input and output, a chording keyboard for text entry, clothing-integrated peripherals such as embroidered keypads, *etc.* These peripherals should be easily reconfigurable based on application and task, and should form the only visible or noticeable components of the system.
5. Computation and Networking: On-body computing power should be low-power, distributed and scalable, supported by a wired peer-to-peer body network and low- and high-bandwidth wireless networking options, *e.g.* CDPD cell modem and 802.11.
6. Open Design: ideally, all components of the platform, hardware and software, should be fully published and unencumbered by restrictive licensing that might discourage collaboration or foster wasted effort through re-invention.

These were ambitious goals, and presented significant technical hurdles given the limitations of embedded hardware in 2000.

### 3.4.3 MIThril 2000 Architecture

The MIThril architecture can be broken down into five major categories: packaging, body bus and sensing, body network and computing, user interface, and software. The clothing-integrated package constrains the overall form, weight, and mechanical properties of the system. The body-bus connects sensors to computing nodes, and the body network connects computing nodes to each other and to off-body networks. Diagrams of the logical and physical organization can be found in Figure 3-2 and Figure 3-3.

The original MIThril 2000 included one pound (six to eight hours) of lithium-ion batteries, two computing nodes (BSEV and CerfBoard with 1GB IBM MicroDrive), an 802.11

---

<sup>2</sup>As a practical matter, this implies either one very flexible, easy-to-use high-speed bus or a hybrid high-speed (fast, difficult to work with) and medium-speed (easier to work with) combined bus, which was the design we chose for MIThril 2000.



Figure 3-4: The author wearing his MIThril 2000 system, with and without a shirt covering.

bridge, two I<sup>2</sup>C sensors, seven body-bus cables and two junctions, a body-network hub, a MicroOptical QVGA clip-on HMD and a Twiddler Chording keyboard. The total power consumption was approximately 4 watts when idle with peaks as high as six or eight during periods of high disk-access and 802.11 use. The addition of a light cotton/poly shirt “chassis” brings the total system weight to three and three-quarters pounds, lighter than a typical leather jacket.

### 3.4.4 Packaging

Packaging is an extremely important contributor to overall wearability, flexibility, and re-configurability. The first MIThril package was a light-weight mesh vest suitable for wearing under a shirt or jacket. This package, though not truly clothing-integrated, provided excellent comfort, access, and configurability.

This vest evolved into a zip-in vest liner (see Figure 3-4). This liner was made of a light-weight, cool mesh fabric and structural belting, and is compatible with a range of outer wear options, from cotton shirts to Armani suits. Off-the-rack clothing is easily modified to accommodate the liner with the addition of a zipper. Distributing the system



Figure 3-5: MIThril 2000 soft mount.

across the torso maximizes the useful sensor and component real-estate, allowing for the convenient placement of computing nodes, microphones, cameras, breathing and heart rate sensors, accelerometers, directional tag readers, *etc.* The package distributes the weight of the system evenly across the shoulders, resulting in a comfortable, balanced feel very similar to ordinary clothing. The positioning of equipment on the body was chosen as a subset of the locations suggested by [20], providing the user not only a full range of motion, but also the ability to sit comfortably and to wear ordinary outer wear without significant interference.

MIThril 2000 components were mounted on the liner using the “soft mount” system, a combination of Velcro and tie-on fabric panels (see Figure 3-5). The soft-mount system allows components to be easily removed and repositioned to accommodate reconfigurations or washing the liner.

Cables were routed on the body using Velcro and neoprene cable anchors (Figure 3-6). The anchors hold the cables securely but are easy to open for maintenance and fail gracefully under excessive load. Components are placed and cables are routed to avoid the contact-points along the back and shoulders, allowing the wearer to sit comfortably, move naturally, and even fall down without damaging the equipment — as I inadvertently discovered during a cell-phone-induced bike crash.

The primary problem with the MIThril 2000 packaging proved to be the large number of cables and components, which required careful routing and placement.

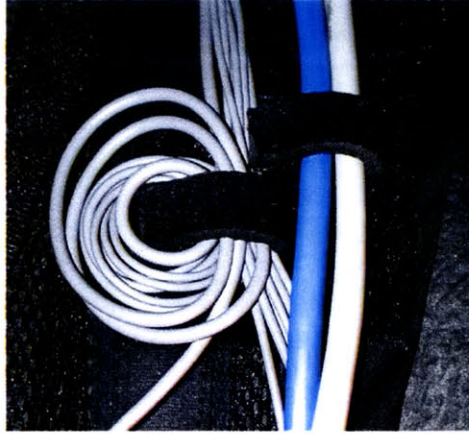


Figure 3-6: MIThril cable anchor.

### 3.4.5 Sensors and the MIThril body-bus

In MIThril 2000, on-body connectivity for sensors, peripherals, and power is provided by the MIThril body bus. The body bus, composed of body-bus cables and junctions (Figure 3-7), is a branching single-connection power/data bus that provides regulated 5V and unregulated 12V power, USB, I<sup>2</sup>C, the Dallas Semiconductor one-wire protocol, and three unused twisted-pair connections through a stranded 16 conductor cable. The body-bus connectors are locking, strain-relieved, Hirose 3500 connectors rated for 20,000 insertion cycles.

Through its branching structure and multi-protocol design, the MIThril body bus simplified on-body device placement and cable routing. The components connected by the body bus were designed to be thin and flat, and as small as possible. Although MIThril 2000 systems had many body-bus components joined with mechanical connections, the robust connectors and careful attention to strain relief made body-bus-related failures uncommon. The modular design proved very helpful for troubleshooting.

The MIThril 2000 body bus included the USB1 protocol in order to provide medium-to-high (12 megabit per second) bandwidth and compatibility with off-the-shelf USB cameras and microphones. A MIThril body bus junction was developed that acted as a USB 1.1 compliant hub in order to support this functionality.

Unfortunately, the USB protocol is complex, and the USB master functionality of the BSEV node conflicted with the use of the Twiddler chording keyboard. On the bench we

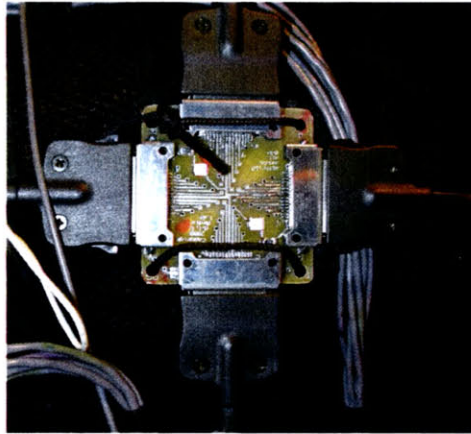


Figure 3-7: MITHril body bus junction. The Body Busy uses Hirose HRS3516-SR connectors for locking, high-insertion-cycle, high-torque-rated, strain-relieved connections.

were able to demonstrate support for a number of off-the-shelf cameras and audio devices, including the D-Link DSB-C300 camera (OmniVision OV511 chipset) and audio devices by Tenex, but USB functionality has not yet been used in on-body applications.

Much more important to the real-world use of the MITHril body bus is the I<sup>2</sup>C protocol. We selected the I<sup>2</sup>C protocol to compliment USB on the body-bus. The Phillips Inter-IC or I<sup>2</sup>C protocol[68] is a multi-device two-wire serial protocol commonly used in industrial and embedded applications. The I<sup>2</sup>C protocol provides less bandwidth than USB (400 kbps in High-speed mode) but is much simpler to implement and suitable for a wide-range of low-bandwidth microcontroller-based sensing applications. A number of I<sup>2</sup>C-based MITHril sensors were designed by the author, including an IR tag reader (for use with the Crystal Tag System[86, 85] shown in Figure 3-8), a three-axis accelerometer, a smart battery board for hot-swappable battery power, and a multi-function sensor board combining a tag reader, an accelerometer, an I<sup>2</sup>C to RS232C converter, and modular analog sensor support. Body-bus sensors developed by others include a GPS, a microphone, and an ECG/EMG/GSR biosensing board.

The Dallas Semiconductor one-wire protocol is used by a variety of single-chip devices, including unique ID tags, temperature sensors, and battery monitor/chargers. Many body bus devices were designed to include one-wire ID chips, and we experimented with the use of this protocol for device ID and power control, but it has yet to be used in a real research application.

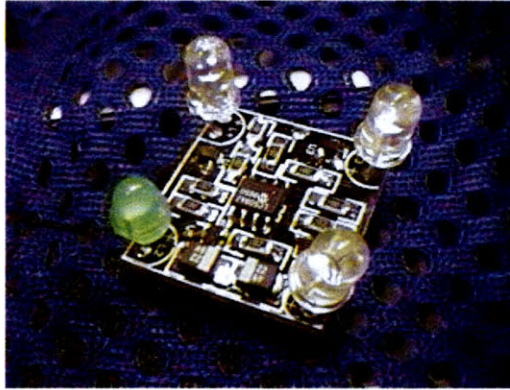


Figure 3-8: The first version of the Crystal Tag IR active tag was developed by Ari Benbasat Rich DeVaul. Picture by Josh Weaver.

The MIThril body-bus design facilitated our memory research applications by providing a modular, integrated framework for communicating with body-worn sensors. The simplicity of the I<sup>2</sup>C protocol combined with the multi-slave, passive-hub protocol design and moderate bandwidth have proved to be extremely useful and widely applicable in real research applications.

### **3.4.6 Computing Nodes and the MIThril 2000 Body Network**

On-body computing resources are provided by a peer-to-peer network of one or more high-performance low-power computing nodes. The only requirement the MIThril 2000 body network imposes on the choice of computing node is that all devices must be capable of 10 megabit per second Ethernet and run off of 5V regulated power. (Ergonomics and power-consumption impose additional practical constraints on the choice of computing nodes.) In addition, at least one node must support the the MIThril body network and provide the required user interaction resources.

The MIThril body network combines 10 megabit per second Ethernet and 5V regulated power, providing a single-connector power/data connection to all body-network devices. A 10-base Ethernet hub modified to distribute power as well as data (See Figure 3-9) ties the computing nodes together, and provides the option of bridging the body network to wired off-body networks through a crossover cable.

The “standard” MIThril 2000 implementation employs three nodes: the BSEV node,

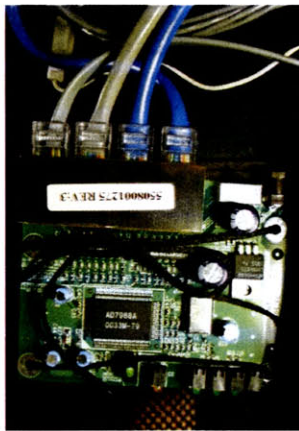


Figure 3-9: A MITHril body-net hub.

the CerfBoard file/application server , and an 802.11 wireless bridge, which are described in more detail below. However, the body network was designed to accommodate nearly any 10-base compatible computer, including higher-performance Pentium-based embedded systems suitable for real-time computer vision applications.

The BSEV node is a combination of the ipEngine1, an MPC823 based single-board computer from Brightstar Engineering, and a MITHril body-bus/video driver board (See Section 3.4.7) that provides the body-bus terminus and drives the head mounted display. The ipEngine1 provides a 66 MIPS MPC823 (a PowerPC-derived processor with integrated peripherals), 16 MB of RAM and an Altera FPGA. Combined with the body-bus/video driver board, the result is a moderately powerful and very flexible sensing and user interaction computing node that consumes less than two and a half watts of power. The BSEV node with display and network consumes less than three watts of power.

The SA 1110 Strong-Arm based Intrinsyc CerfBoard file/application server is a low-power computing resource, and in combination with a one gigabyte IBM MicroDrive acts as the on-body NFS file server. (The XScale CerfBoard was not yet available in 2000) The CerfBoard provides 200 MIPS of computing power and 32 MB of RAM, allowing it to run complex applications. The total power consumption of the CerfBoard with MicroDrive ranges from less than one watt to as high as three depending on disk activity.

The original 802.11 bridge was a repackaged Lucent WaveLAN Ethernet converter and Orinoco card, which provided 11 megabit per second wireless networking. Due to the

relatively large size and high power consumption of this device, we subsequently replaced it with a second CerfBoard with a Compact Flash 802.11b card running Linux. This version of the wireless bridge was able to function as a full masquerading firewall and an additional 200 MIPS of networked computing resource.

At the time we were developing MITHril 2000, we were collaborating with Urs Anliker, Paul Lukowicz, and Gerhard Troester of the Wearable Computing Lab at ETH in Zurich in the development of the WearARM, a modular high-performance low-power computing node which was intended provide a superset of the functionality of the BrightStar ipEngine1, the CerfBoard, and the 802.11 bridge. When combined with a body-bus/video-driver board similar to the one currently in use in the BSEV node, it was intended provided a single-node solution for a wide range of MITHril applications.

The development process took longer than anticipated, and due to the difficulties of production and driver writing we never brought this system up in a functioning MITHril system.

### **3.4.7 User Interface Hardware**

When designing MITHril 2000, we considered the user interface to be one of the most important aspects of the design, since it is only through the interface that the user interacts with the applications. The user interface also generally places the highest demands on the user's attention and strongly shapes the user's experience. Limitations in the UI hardware place hard limits on the ways applications can interact with the user. Our design philosophy was, "the interface is the computer."

Although both audio and visual output were of interest, we were most interested in the possibilities of using a light-weight, monocular head mounted display that could provide usable text and color graphics but would obscure as little of the user's field of view as possible. At the time we began work on MITHril 2000, the MicroOptical Clip-on QVGA display (based on the Kopin CyberDisplay 320 Color chip) proved the best candidate.

The MicroOptical QVGA clip-on is a small, light display that clips to the wearer's glasses and provides a full-color, quarter-VGA image at 60Hz refresh, or 180Hz grayscale.



Figure 3-10: MicroOptical clip-on without IR tag reader.

The MicroOptical is unobtrusive and obscures little of the user's field of view. In 1999, the MicroOptical Clip-on QVGA was the state-of-the-art in small, light-weight head mounted displays.

The single biggest engineering challenge in the development of MIThril 2000 was the video driver circuit for the MicroOptical, which was part of the more general BSEV (Bright Star Engineering Video) interface board for the BrightStar ipEngine1. Rather than rely on the bulky, power-hungry VGA converter box that shipped with the MicroOptical displays, we decided to engineer our own driver to use the direct output of the MPC823's on-board LCD driver, bypassing VGA signal conversion. This eliminated the the bulky converter box and reduced power consumption by almost four Watts.

Had I any idea just how hard this engineering project would prove to be, I almost certainly would have looked for other alternatives or simply dealt with the frustrations of the existing video driver. As it turned out, the development of this hardware required the reverse-engineering of the MicroOptical hardware. It took many more months and six iterations to get the driver working properly. Unfortunately, this was not the end of the process, because a subsequent change MicroOptical made to the display unit itself broke our driver hardware (but not the MicroOptical reference design), requiring further troubleshooting<sup>3</sup>.

---

<sup>3</sup>Despite the best of plans, the engineering of the video driver circuit largely fell to me. And although I had excellent help from Ed Keyes and other members of the wearables group, the process was difficult and frustrating in the extreme. Although I learned a great deal very rapidly, I would recommend against undertaking the design of a Kopin video driver as one's first experience with medium-speed mixed signal design.

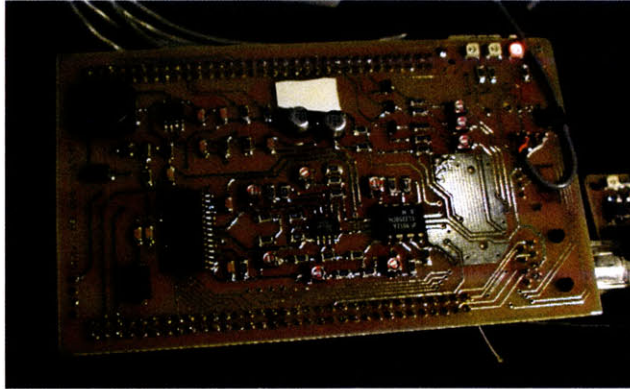


Figure 3-11: BrightStar ipEngine1 with an early prototype MIThril Kopin video driver board.

An important feature of the display driver is that it provides direct control of backlights and signal timing, allowing for software-switching between 60 Hz field-sequential color and 180 Hz high-brightness gray-scale modes. More importantly, the precise control of backlight timing made this combination of HMD and driver hardware perfectly suited to the subliminal visual cuing research presented later in this thesis.

For audio output, several USB audio devices were demonstrated in bench testing, though none in wearable application. In addition, I designed a body bus I<sup>2</sup>C audio output device, although the limited bandwidth of the I<sup>2</sup>C bus made this unsuitable for many applications, and full Linux sound driver software was never written. An I<sup>2</sup>C-based audio input device was subsequently designed and demonstrated by Chris Elledge, but operating only at low (8K) sampling rates.

For key entry, MIThril 2000 supported hot-swapping between a modified Twiddler chording keyboard and a modified Palm folding keyboard, with full Linux driver support. Microcontroller-based “button board” interfaces were also designed,

### **3.4.8 MIThril 2000 System Software**

The core operating system for MIThril is Linux. Several different Linux distributions were used for different components of the system, including a modified HardHat distribution for the BrightStar ipEngine1, Debian Arm Linux on the Intrinsic CerfBoard file/application server, and Intrinsic’s own light-weight Linux distribution for the wireless bridge Cerf-

Board.

The choice of Linux as the core operating system was important for the development of the MITHril platform, as it was necessary to develop custom device driver code for our I<sup>2</sup>C bus, USB master support on the BrightStar ipEngine1, and our Kopin video driver hardware. This would have been difficult on a closed-source alternative, such as WindRiver VxWorks or Windows CE. We chose Linux over other open-source alternatives (such as FreeBSD) because of the large and growing community of Linux developers, who proved an invaluable support resource during our development process.

In addition, the use of Linux has allowed us to standardize on a single core platform across our entire lab and development process. We literally have the same kernel (and in most cases the same distribution, the “stable” branch of Debian) running on MITHril 2000 embedded computing nodes to desktop development machines to servers. This uniformity has facilitated the portability of our middleware and other important software components across our infrastructure.

The middleware that supports virtually all MITHril research applications is described in Chapter 4. Our prototype user interface code and applications are described in Chapter 5.

### **3.4.9 MITHril 2000 Evaluation and Experience**

MITHril 2000 is now more than four years old. Although certain aspects of the system design proved very challenging (the Kopin video driver circuit chief among them) most of our design goals for MITHril 2000 have been met at the time of this writing.

We learned a number of important lessons working on MITHril 2000. One of the first lessons learned is the importance of strain relief and high reliability connectors for on-body connections. Unlike a hard chassis, clothing is constantly flexing and shearing with use. Although the instantaneous torque placed on body-bus connectors is ordinarily low, the cumulative strain from constant motion will cause failures in low-torque-rated low-duty-cycle connectors. Our first MITHril prototype was retired after less than a year of use due to connector failure, prompting a complete reengineering of the body-bus connector system.

Another lesson is the importance of high-quality design and fabrication of the textile

components of the system. Although “soft,” the textile packaging must be carefully designed and fabricated to meet demanding engineering, comfort, and fashion requirements. For instance, the MIThril vest liner combines multiple fabric types and structural belting to maintain its shape and support the equipment load without sagging or hanging away from the body. Many design iterations went into development of the soft-mount system and the placement and routing of components.

Our hypothesis that placing relatively small, low-aspect-ratio “hard” parts near to front and sides of the torso would protect them without the addition of a hard shell or extra padding has been supported. Evidence for this includes the basic reliability of the author’s MIThril 2000 system through years of daily use, including travel by car, subway, and airplane (prior to 9/11), and athletic activities such as biking<sup>4</sup>.

The main problem with the MIThril 2000 architecture is that it has proved too difficult and expensive to fabricate in large numbers<sup>5</sup>. As a result, no more than twenty systems were ever assembled, with at most six running simultaneously. Although this has proved sufficient for the author’s Memory Glasses research, it has made wider application of MIThril difficult.

### **Comments by David Kaplowitz**

David Kaplowitz has used MIThril 2000 on a daily basis for longer than anyone except perhaps myself. In a recent email, this is what he had to say about his experience with MIThril 2000:

Form factor, and fragility need to be worked on. The packaging is flexible, and useful. The battery life, for the amount of batteries I carry is wonderful. There should be a cleaner way to switch between batteries, and line current (induction) and I would really (REALLY) like a way to get regulated power that didn’t involve the brightstar board. Oh, and to integrate the old with the

---

<sup>4</sup>... including one dramatic bike crash, in which neither the author nor the MIThril system were seriously damaged, and in fact during which the system continued to gather accelerometer data.

<sup>5</sup>All fabrication of MIThril systems has been done in-house at the Media Lab, including the hand-stuffing of most of the custom hardware, and hand-assembly of most cables. I believe that had we successfully outsourced most of this work much larger numbers of MIThril 2000 units could have been up and running.

new, I'd like some way to leech power from the system to power a Zaurus, and maybe some way to get the "IP over USB" or what have you integrated, so I can show others pieces of what I see, without having to unmount my heads-up display. Oh, and perhaps another board or the like with a CF- $\mu$ cell-modem, for those times I'm not near an 802.11b source.

Some software on it that does "war-walking" and a gps are a lot of fun recreationally, it is very helpful for me to carry my brain (CVS enabled) around. Offline mail reading is a wonderful thing (Yay IMAP). Low light levels means that, when I'm awake at night, in a shared-sleep environment, I can compute (quietly, and with minimal light) and not disturb others.

I have yet to see the same social stigma you talked about, but then again, I might move in different circles.

I would also enjoy seeing the packaging smaller, such that I could simply put it on with my glasses, (say in the temple of them) and have a cut-out or reflector in the glass itself, with a camera at the bridge, and both microphones, and speakers at the ends of the temples, but that's simply because I'm greedy \*grin\* ...

## **3.5 MIThril 2003**

### **3.5.1 Overview**

More recently, MIThril 2000 has evolved into MIThril 2003, a multi-user PDA-centric configuration suitable for a range of group support/collaboration research applications[14]. MIThril 2003 extends the previous MIThril modular architecture into the domain of large-scale wireless group applications by leveraging the availability of inexpensive Linux-based PDA hardware.

Beginning in 2001 we saw the need for a light-weight data acquisition system that could talk to MIThril sensors, primarily for biomedical and social networks research applications. I proposed an architecture for a light-weight data acquisition system that became

the Hoarder, designed by Vadim Gerasimov. The Hoarder combined a microcontroller Compact Flash interface and real-time-clock developed by Vadim with a well-specified board-to-board connector to support sensor daughter boards and a MIThril body bus connector. In combination with a multi-sensor accelerometer/microphone/tag reader board I designed this became the first MIThril platform to support experiments with dozens of users. (See the description of the Shortcuts project below).

Although the Hoarder/MIThril sensor combination allowed for interesting data acquisition experiments (including Josh Weaver's Parkinson's project[87], described below), the interaction capabilities were extremely limited<sup>6</sup>. Further, although the hoarder supported a simple RF interface, true peer-to-peer networking was not possible.

By 2003, inexpensive, Linux-capable PDAs with significant signal processing and communications capabilities were available. By combining such devices with the Hoarder (and subsequently the SAK2, designed by Michael Sung) and MIThril body bus peripherals, we created a wearable with many of the capabilities of the original MIThril system, but at a fraction of the original MIThril 2000 system complexity and cost. The result is MIThril 2003. Although a MIThril 2003 system provides less processing power and sensor bandwidth than a multi-node MIThril 2000, it does provide sufficient capabilities for a range of important sensing, classification, and user-interaction applications.

### **3.5.2 Architecture**

The MIThril 2003 hardware architecture is a highly flexible, modular system that is tied together by wired/wireless networking protocols and a unified multi-protocol wired power/data bus, called the MIThril Body Bus, for sensors and peripherals.

### **3.5.3 Computing Nodes**

For applications requiring real-time data analysis, peer-to-peer wireless networking, full-duplex audio, or graphical interaction, MIThril 2003 employs a Linux-based PDA, such

---

<sup>6</sup>In 2002 I fabricated a prototype audio output extension to my multi-sensor board for the Hoarder that used a serial DAC and headphone driver. I intended this for audio recording and playback applications, including reminder recording and delivery, but this has yet to be used for any real research application.

as the Sharp Zaurus SL-5500 shown in Figure 3-12. We chose the Zaurus SL-5500 because it provides excellent Linux support combined with a range of capabilities in a small, inexpensive package.

The Zaurus SL-5500 is a complete embedded Linux system. It provides a 206-MHz StrongARM processor, 64 MB SDRAM, 16 MB Flash, CF and SD expansion slots, full duplex audio, qVGA color touch screen, and an integrated QWERTY keyboard. The StrongARM provides sufficient computing power for moderate bandwidth signal processing and real-time classification.

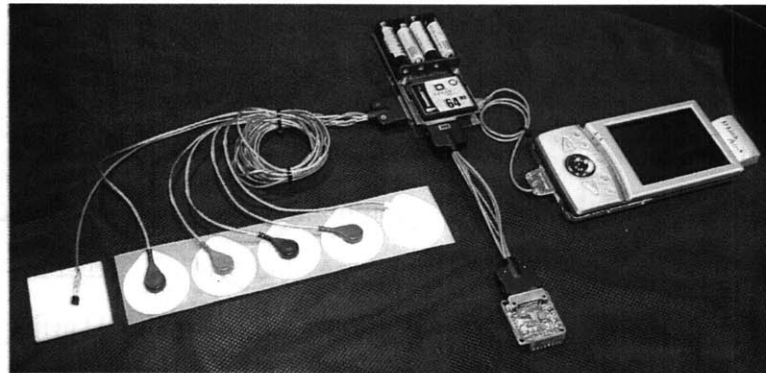


Figure 3-12: MITHril 2003 system, comprised of the Zaurus PDA (right) with Hoarder sensor hub and physiological sensing board (top), EKG/EMG/GSR/ temperature electrodes and sensors (left) and combined three-axis accelerometer, IR tag reader, and RS232/I2C bridge (bottom)

The CF card slot enables a variety of peripherals to be attached, including cell-phone modems, image and video cameras, Bluetooth and 802.11b (WiFi) wireless, and even head mounted displays; There is now a CF MicroOptical VGA color HMD produced by Interactive Imaging Systems. The Zaurus also provides an RS232 serial port, which we use to interface with the MITHril sensor hub described below.

### 3.5.4 Sensor Hubs

A sensor hub is used to interface the Zaurus SL-5500 with the MITHril body bus. This hub bridges between the RS232 serial interface on the Zaurus and the Phillips I2C multi-device serial protocol used on the MITHril body bus. Two sensor hubs are currently in use, the Hoarder and the SAK2.

The Hoarder provides stand-alone data acquisition capabilities in the form of a CF storage card interface, a real-time clock for sequencing, battery power, and an optional FM wireless transceiver. The Hoarder supports sensor daughter boards as well as I2C devices through the MIThril body bus.

The SAK2, recently designed by Michael Sung, is a multi-function sensor/hub device that provides many of the same capabilities as the Hoarder, but in a smaller package and without support for stand-alone data acquisition. The SAK2 fits inside a modified Zaurus docking cradle. In addition, the SAK2 provides regulated power for the Zaurus and MIThril bus and low-power addressable megabit peer-to-peer wireless (2.4 GHz) using the Nordic VLSI nRF24xx parts.

### **3.5.5 Sensor Hardware**

MIThril 2003 sensors are either stand-alone microcontroller-based devices that speak I<sup>2</sup>C on the MIThril body bus, or analog/digital sensor daughter boards that connect directly to the sensor hub. All I<sup>2</sup>C MIThril sensors work with MIThril 2003. In addition, the sensor hub supports two major types of daughter boards. The first is the previously-mentioned multi-sensor board combining a three axis accelerometer, IR tag reader, and analog microphone. The second is a physiological sensing board providing analog 2-channel EKG/EMG, 2-channel galvanic skin response (GSR), and skin temperature sensors.

### **3.5.6 Networking**

The Sharp Zaurus supports wireless IP networking through the CF interface using the 802.11b wireless protocol. This WiFi wireless networking capability is a crucial enabling feature, allowing us to implement multi-node, distributed wearable applications.

The 802.11b protocol provides sufficient bandwidth for many systems to simultaneously stream full-duplex audio and sensor data, as well as sending lower-bandwidth data such as text messaging and state information. With wireless connectivity, data can be streamed to off-body resources for a variety of purposes, including data logging and storage, visualization/display, and signal processing. Persistent network servers are also used

for directory services and resource discovery.

### **3.5.7 MIThril 2003 Evaluation and Experience**

MIThril 2003 is still in the early stages of its development, and has not achieved the stability or usability of MIThril 2000. However, it has been used in a number of group-interaction and social-networks research projects, and its stability and usability will improve with time.

## **3.6 Beyond the Memory Glasses: Other Applications for MIThril**

The following is a brief, incomplete survey of the use of MIThril hardware for applications beyond the Memory Glasses research that is the focus of this thesis. Other applications that relate more to MIThril middleware are mentioned at the end of Chapter 4.

### **3.6.1 Parkinson's Project**

Josh Weaver used a MIThril/Hoarder system to do the data acquisition for his M.S. thesis work[87]. This system combined a Hoarder with multi-sensor board with four I<sup>2</sup>C accelerometers worn on the wrists and ankles. It was used in a pioneering project to develop a medication-state classification system for Parkinson's patients.

### **3.6.2 SAAB WISE Project**

SAAB Aerospace and SAAB Avionics are using a MIThril 2000 vest prototype computer as the basis of a warfighter situation awareness system for use in their network-centric warfare demonstration project. The idea is to leverage the wireless networking and light-weight head-mounted display to provide forward spotters and troop on the ground to access and control UAV (unmanned autonomous vehicles) and sensor-net systems for improved theater situation-awareness and command/control/coordination applications.

### **3.6.3 Shortcuts Project**

Tanzeem Chourdy used the “Sociometer” device, a combination of MITHril/Hoarder, multi-sensor board, and Crystal Tag, in a package designed by Brian Clarkson, for mapping face-to-face social networks as part of her doctoral thesis work[9, 8]. By continuously recording audio and IR tag interactions, she was able to construct an interaction map of research groups in the Media Lab.

### **3.6.4 David Kaplowitz: Daily Use for Systems Administration**

David Kaplowitz, long-time user of the Lizzy, has been using a MITHril 2000 on a daily basis as a systems administration tool since 2001. David’s daily use for almost three years supports the utility and robustness of MITHril 2000.



# Chapter 4

## MIThril Software

### 4.1 Linux Foundations

As described in Chapter 3, Linux forms the foundations of the MIThril software architecture. The choice of Linux provided us a consistent, high-performance operating environment with excellent, development tools, and has facilitated the porting of MIThril middleware to other platforms. For example, MIThril middleware has been recently ported to run under WindowsXP.

### 4.2 Middleware: Software for Real-Time Context Awareness

Context awareness is a critical feature of the Memory Glasses. The user's location, activity, and social state must be classified in real time using body-worn sensors and computing resources. In order to minimize the size and weight of these on-body computing resources, efficiency is an important consideration. Flexibility is also important, since we would like to be able to apply a uniform approach to the classification of all relevant contexts. Further, MIThril 2000 is a distributed computing architecture in which computing resources were distributed across body-worn wired and wireless networks; A network-transparent, modular system would be required to take advantage of this architecture.

At the time we began our memory support research in 1999, no existing well-integrated set of context-awareness tools met our needs. As a result, we decided to engineer an integrated solution that would provide a solid foundation for future research in context-aware computing and real-time sensor classification.

To address these problems, I proposed the Enchantment interprocess communications and signal-processing/modeling architecture and used these tools to develop the MIThril Real-Time Context Engine.

The MIThril Real-Time Context Engine is a practical, modular framework for the development of real-time context classifiers and models. It is implemented using the Enchantment Whiteboard and Signal systems (described in more detail below), and Kevin Murphy's Bayes Net Tool Kit[46], a freely available, powerful graphical models and Bayesian inference system for Matlab.

This section briefly describes three important parts of the MIThril software infrastructure: the Enchantment Whiteboard, the Enchantment Signal system, and the MIThril Real-Time Context Engine. These tools provide the foundation for our Memory Glasses research applications.

### **4.2.1 The Enchantment Whiteboard**

The Enchantment Whiteboard[14]) system is an implementation of a whiteboard inter-process communications system suitable for distributed, light-weight embedded applications. Unlike traditional inter-process communications systems (such as RMI, unix/BSD sockets, etc.) which are based on point-to-point communications, the Enchantment Whiteboard is based on a client/server model in which clients post and read structured information on a whiteboard server. This architecture allows any client to exchange information with any other client without the attendant  $n^2$  complexity in negotiating direct client-to-client communication; in fact, such communication is possible without clients knowing anything at all about each other.

The Whiteboard architecture does for inter-process communication what web browsers and web servers do for document publishing it provides a uniform structure and systematic

organization for the exchange of information that does not require synchronous communications.

The Enchantment Whiteboard goes beyond the web server analogy by allowing clients to subscribe to portions of the whiteboard, automatically receiving updates when changes occur. It allows clients to lock a portion of the whiteboard so that only the locking client can post updates. The system even supports symbolic links across servers, allowing whiteboards to transparently refer to other whiteboards across a network. The Enchantment Whiteboard is also lightweight and fast, imposing little overhead on the communications.

The Enchantment Whiteboard is intended to act as a streaming database, capturing the current state of some system (or person, or group) and on modest embedded hardware can support many simultaneous clients distributed across a network and hundreds of updates a second. We have even demonstrated the ability to use the Enchantment Whiteboard with the Signal system for bandwidth-intensive VoIP-style audio communications. We have used the Enchantment Whiteboard as the basis of group support applications with real-time user interaction with up to 50 people.

The Whiteboard library presents an interface with the following core functionality: publishing, retrieving, subscribing, and locking. When data is retrieved from the Whiteboard, it is not generally removed and remains for other applications to query. According to the client's specification, data that is published to the Whiteboard can either be persistent or transient.

### **4.2.2 The Enchantment Signal System**

For higher bandwidth signals, especially those related to the sharing and processing of sensor data for context aware applications, we developed the Enchantment Signal[14] system. The Enchantment Signal system is intended to facilitate the efficient distribution and processing of digital signals in a network-transparent manner. The Enchantment Signal system is based on point-to-point communications between clients, with signal handles being posted on Whiteboards to facilitate discovery and connection. In the spirit of Whiteboard interactions, the Signal API abstracts away any need for signal producers to know who, how

many, or even if, there are any connected signal consumers.

Any type of structured numeric data can be encoded as a signal. Signal producers may be sensors, feature extractors, filters, or regression systems, and may produce other signals in turn. A typical organization is a sensor signal producer talking to a feature extraction signal consumer, which in turn produces a feature signal that is consumed by one or more modeling or regression systems.

The results of modeling or regression can themselves be signals, or (more typically) posted on a whiteboard for other clients to use. The Enchantment Signal system has been used, in conjunction with the Whiteboard, to support the sensing and real-time classification of physiological and audio data from groups as large as 50 people.

### **4.3 The MIThril Real-Time Context Engine**

Context awareness, or knowledge about the state of the user, task, or environment derived through non-explicit user input, is widely seen as a key to reducing the complexity of human-computer interaction tasks for portable and wearable computing applications[52]. Classifying some types of context is relatively easy. For instance, outdoor location can often be measured directly using GPS (The US Military's Global Positioning System). However, classifying user activity state is often much more difficult. As recently as two years ago, Laerhoven *et al.* proposed a 30 accelerometer system for classifying five user activity states [34].

We find that a systematic approach to activity classification based on modern machine learning techniques can greatly simplify the process of developing and implementing real-time activity classification models. The MIThril Real-Time Context Engine was developed to provide this systematic framework in a flexible, modular, open-source form.

The MIThril Real-Time Context Engine is an open-source, light-weight, and modular architecture for the development and implementation of real-time context classifiers for wearable applications. The high-level organization of the Context Engine is based on the classification process described by Duda and Hart[18]. It is implemented using the Enchantment Signal system and Kevin Murphy's Bayes Net Tool Kit[46].

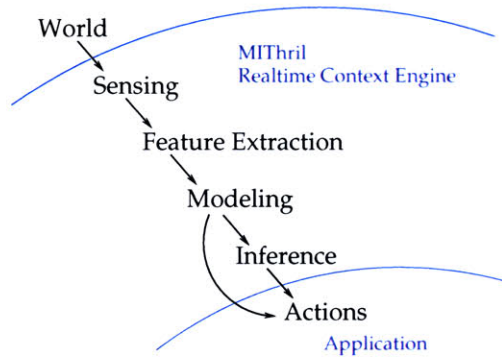


Figure 4-1: The MIThril Real-Time Context Engine

The classification methodology we employ is based on modern “white box” statistical modeling techniques. These techniques, such as Gaussian mixture models or HMMs, differ from older “black box” modeling techniques (like hidden-layer neural networks) in that the model structure and parameters lend themselves to meaningful interpretations and analysis. This is important, because it allows us to not only build classifiers that work, but also to understand how they might fail.

The MIThril Real-Time Context Engine abstracts the process of sensing, modeling, and inference as a four-stage process, as shown in Figure 4-1.

1. Sensing: In the sensing stage, a digital sensing device measures something in the real (analog) world, resulting in a digital signal of sampled values. For example, a microphone sensor converts continuous fluctuations in air pressure (sound) into discrete sampled values with a specified resolution, encoding, and sampling rate.
2. Feature Extraction: In the feature extraction stage, a raw sensor signal is transformed into a feature signal more suitable for a particular modeling task. For example, the feature extraction stage for a speaker identification classification task might involve converting a sound signal into a power spectrum feature signal.
3. Modeling: In the modeling stage, a generative or discriminative statistical model, (such as a Gaussian mixture model, Hidden Markov Model, or Support Vector Machine hyper-plane classifier, etc.) is used to classify a feature signal in real time. For example, a Gaussian mixture model could be used to classify accelerometer spectral features as walking, running, sitting, etc.

4. Inference: (Matlab implementation only) In the inference stage, the results of the modeling stage, possibly combined with other information, are fed into a Bayesian inference system for complex interpretation and decision-making.

The first three stages of this four-stage process have been implemented in C and C++ using the Enchantment Signal and Whiteboard system. (We have not yet had a practical need for a real-time inference stage.).

The process outlined above assumes that one already know what features to extract and the type, structure, and parameters of the model. In order to learn these parameters, one must first gather a certain amount of labeled training data, do appropriate analysis to choose features, and then learn the model parameters. The Real-Time Context Classifier toolkit provides data logging, labeling, and analysis tools to facilitate this process.

### **4.3.1 Classifier Development: Three-Class Model Example**

In this section we describe a general methodology for developing context classifiers. The specific example we have chosen is an accelerometer-based user activity classifier, but the same methodology can be applied to the development of classifiers for biosignals, audio data, or almost anything else.

This classification process described above assumes that features and model structure have been chosen, and that the model parameters are known. In order to develop the model, we employ a standard model development process like the one described in [18]:

1. Data acquisition. Gather labeled sensor data from real-world activities. Divide this data into training and testing (hold-out) data sets.
2. Feature Selection. Visualize the data, perform preliminary analysis and select appropriate features.
3. Model Selection. Based on the choice of features and other requirements, choose an appropriate model type and structure.
4. Model Training. Using desktop hardware, run an appropriate algorithm (such as EM) to learn the model parameters from the training feature set.

5. Model Evaluation. Test the performance of the classifier on the hold-out data. (Other model evaluation procedures are possible, such as leave-one-out cross validation.)
6. Model Implementation. Once model performance is acceptable, implement the model in a form suitable for the portable or wearable application.

These steps are discussed more fully below in the context of the development of a three-class activity model.

In this section we discuss each of the model development stages as they relate to the development of a three-class (running, walking, standing still) activity classifier. This classifier was developed specifically as a demonstration and tutorial application of the MIThril Real-Time Context Engine. All of the Matlab and C/C++ code used in this process are available on the web, including a Matlab tutorial and cookbook for training the model and exporting the features into our real-time system.

### 4.3.2 Choice of Sensor

The first step is choosing an appropriate sensor for the classification task. We chose a single three-axis accelerometer as the basis of our three-class motion classifier.

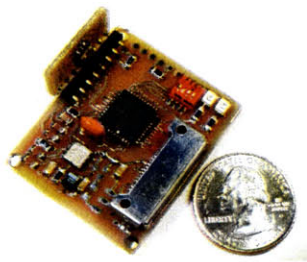


Figure 4-2: MIThril accelerometer sensor

We used a custom-designed microcontroller-based device using the ADXL202JE part from Analog Devices. The ADXL202JE is an inexpensive MEMS-based two-axis accelerometer that is accurate to better than 2% over a  $\pm 2$  G range. Our sensor uses two of these accelerometers mounted at right angles to each other, resulting in a four-axis sensor with one redundant axes. The design for this device is available on the web: <http://www.media.mit.edu/mithril/hardware/>.

Although we designed our own sensor to provide maximal flexibility and compatibility with the MITHril body bus, any suitably sensitive three-axis accelerometer device could have been used. An important feature of the MITHril real-time context engine methodology is that it is not wedded to any particular sensing or computing platform.

### **4.3.3 Location of Sensor**

Not all parts of the body are equally involved in all activities, making the choice of sensor location important. Since we were interested in gross body motion, we placed the accelerometer on the subject's belt, near the wearer's center of gravity.

### **4.3.4 Sampling Rate and Resolution**

The choice of sampling rate  $F_s$  is important since the the highest frequency that can be recovered from the sampled signal is half the sampling rate. This is the Nyquist frequency,  $F_N = F_s/2$ . The resolution is also important, because it (along with sample rate) limits the amount of information available in the signal. More information is not always better, as higher bandwidth signals require more processing power to classify.

For our application we chose a 47.8 Hz sample rate with a resolution of one byte per axis. The resolution was limited by the sensor, and the sampling rate was chosen to put the Nyquist frequency well above what we thought were the fundamental frequencies associated with walking and running.

### **4.3.5 Data Collection**

Once the sensor is chosen and positioned and the sampling rate and precision are set, the next step is to gather labeled training data. This data should be gathered under circumstances as close as possible to the actual real-world activities one wishes to model. For example, running on a treadmill might make it easy to collect running motion or physiological data, but there is no guarantee the data will look like data from the same subject under less controlled conditions, such as running outdoors, in the snow or mud, *etc.* A running model based only on treadmill data is unlikely to generalize beyond the laboratory.

For our application we used a MITHril 2003 wearable[14] based on the Sharp Zaurus SL-5500 PDA to log the accelerometer data. The apparatus used to gather and label the data has important implications for the reliability and human factors of the experiment, to say nothing of the sanity of the experimenter; There is little that is more frustrating than dealing with an unreliable data acquisition system while trying to run an experimental protocol.

### **Annotation Techniques**

The choice of annotation technique is important. A structured protocol makes labeling the data easy. For example, to develop the three-class model we asked our subject to walk for 2 minutes, run for 2 minutes, and rest for 2 minutes. However, this approach may not work for less controlled, more spontaneous activities.

Other alternatives include self-annotation by subjects, observation by a third party, and audio/video recording. We have used all of these techniques for different experimental designs.

### **4.3.6 Feature Selection**

The first step in feature selection is to graph the data with the labels. By visualizing the data in different ways good features can be discovered. A good feature is one that has obvious correlation with the labels; Selecting good features makes model development and implementation easier. Selecting bad features may make model development difficult or even impossible.

For the three-class model we chose power-spectrum features because there was a clearly visible correlation between our classes and the features, as can be seen in Figure 4-3.

Having chosen the type of feature it is still necessary to choose the specific feature parameters. To reduce the computational burden on the wearable we chose a comparatively small 64 element FFT window (spanning 1.34 seconds) with 32 samples of overlap, producing one 32 element power-spectrum every 0.67 seconds. The zero frequency component represented a large, constant DC bias across classes (gravity) and was removed.

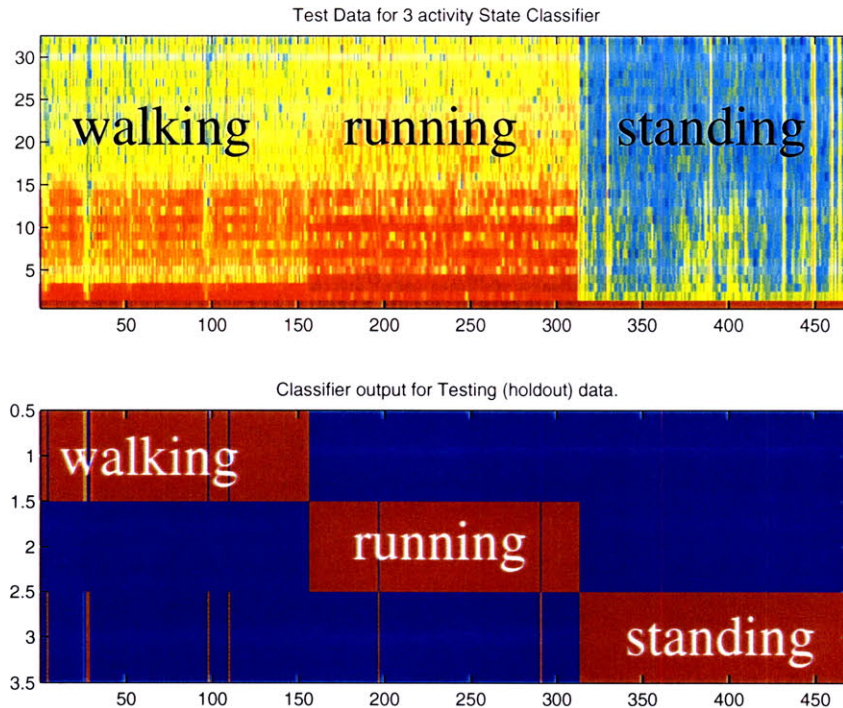


Figure 4-3: Three State Gaussian Activity Classifier: Spectral Features Plus Classifier Evaluation on Testing Data.

### 4.3.7 Model Selection

There are many possible model types to choose from, including Support Vector Machines, Hidden Markov Models, and Gaussian Mixture Models to name only a few. In general, models should be as simple as possible, but no simpler. A full discussion of model selection can be found in [18].

We chose a simple Gaussian model for our three-class activity classifier. The reason we chose this model is: (1) The number of activity classes was small. (2) Visual inspection of the feature data suggested that the classes were easily discriminable using the features and that no higher-order Markov statistics would be required. (3) The Gaussian model is simple, statistically principled, and can be estimated directly from training data. Had the simple Gaussian model proved insufficient (by failing to perform well enough for our purposes on the testing data) a Gaussian mixture model would be a reasonable next choice, starting with two mixture components per class, and increasing the number of mixture components until either the model was performing well enough or the accuracy on the hold-out data started

to decline.

In this model, the distribution of features for each activity class is modeled by a single 31-dimensional Gaussian, which is specified by a 31 dimensional mean  $\mu$  and  $31 \times 31$  covariance matrix  $\Sigma$ . The probability of an observation (feature vector)  $x$  given mean  $\mu$  and covariance  $\Sigma$  is given by the multivariate Gaussian formula

$$P(x|\mu, \Sigma) = \frac{1}{(2\pi)^{d/2} |\Sigma|^{1/2}} e^{-\frac{1}{2}(x-\mu)^T \Sigma^{-1} (x-\mu)}, \quad (4.1)$$

where  $d$  is the dimensionality of the feature vector  $x$ .

Thus, to classify an observation  $x$ , one calculates the posterior probability  $p_n = P(x|\mu_n, \Sigma_n)$  for each class  $n$  and chooses the class with the largest  $p_n$ .

### 4.3.8 Model Training

In the case of a simple Gaussian model, we can estimate the model parameters directly from the training data. We start by numbering the model classes  $1, 2, \dots, N$ , where  $N$  is the number of classes in the model: walking = 1, running = 2, standing = 3 ( $N = 3$ ). For each model class  $1 \leq n \leq N$  there is a training set  $X_n$  that contains  $j_n$  observations (in this case, 31-dimensional spectrum features). We estimate the model parameters  $\mu_n$  and  $\Sigma_n$  separately for each class  $n$  as follows:

$$\mu_n = \frac{1}{j_n} \sum_{x \in X_n} x \quad (4.2)$$

$$\Sigma_n = \frac{1}{j_n} \sum_{x \in X_n} (x - \mu_n)(x - \mu_n)^T \quad (4.3)$$

For other types of models, an iterative maximum-likelihood estimation algorithm such as EM[10] may be used to learn the model parameters from the training data.

### 4.3.9 Model Evaluation

It is important to test the performance of the model using different data than was used to train the model. Each class  $n$  has its own set of hold-out observations,  $Y_n$ . For each class  $n$

and observation  $y \in Y_n$  we test our classifier by calculating  $p_i(y), 1 \leq i \leq N$ . For correct classifications,  $p_n(y) > p_i(y), i \neq n$ . We count the total number of correct classifications and divide by the total number of hold-out observations to produce an estimate of the accuracy of the classifier.

Figure 4-3 shows the training set  $Y_n$  for each of the three classes at the top, and the classifier output for each of the three classes at the bottom. In the classifier output the regions of lighter color on the diagonals are accurate classifications, the occasional lighter-colored line off the diagonal represents an error. As shown here, our model is 99% accurate on the testing data<sup>1</sup>

### 4.3.10 Real-Time Implementation

Unfortunately, Matlab is not a suitable environment for implementing real-time classifiers on light-weight hardware. After the model parameters are learned, they are exported from the Matlab environment and imported into the real-time system as plain ASCII text files. The MITHril Context Engine provides C/C++ based feature extraction and modeling tools that provide real-time performance on modest embedded Linux hardware, including the Sharp Zaurus PDA. For more information on these tools, see [14] and the Enchantment web site, <http://www.media.mit.edu/wearables/mithril/enchant>.

### 4.3.11 Other Models

We have implemented other models with the MITHril Real-Time Context Engine tools, including an accelerometer-based head nodding/shaking classifier, a five-class single-accelerometer activity classifier employing a hybrid Gaussian/Markov model<sup>2</sup>, and an audio-based whistling

---

<sup>1</sup>The model was “trained” once (parameters estimated as described) and then formally evaluated on all holdout data. This is simple and fast, but requires that a significant amount of labeled data be hidden from the model training process. Because this model is simple it requires comparatively little data to train. However, if the model were more complex more training data would be required. Under such circumstances other evaluation techniques might be more appropriate, such as iterative leave-one-out cross-validation. It may also be appropriate under some circumstances to use all available labeled data for training and to employ more theoretical methods for predicting generalization error. For more discussion of model evaluation see [18].

<sup>2</sup>The activity states were sitting, walking, running, riding an bicycle, and riding the Boston subway while sitting, with a garbage “everything else” class. This model proved to be accurate and could reliably distinguish

classifier (so you can whistle commands to your computer)<sup>3</sup>. Biometric classifiers based on EKG, EMG, and GSR signals are also being developed.

### 4.3.12 Conclusions

The MIThril Real-Time Context Engine has proven itself as a useful methodology and set of tools for the development of these models. We have made our methodology, tools, and hardware designs freely available in the hope that this framework may be of use to other researchers who are investigating the applications of context classification in wearable, portable, and mobile interaction,

<http://www.media.mit.edu/wearables/mithril/context>.

## 4.4 Beyond the Memory Glasses: Other Applications

The following is a brief, incomplete survey of the use of the Enchantment middleware and the MIThril Real-Time Context Engine for research applications beyond memory support. The Enchantment tools and Context Engine have proven to have wide applicability, particularly in the domain of social network mapping and real-time physiological monitoring.

### 4.4.1 Group Interaction Research using the Enchantment Middleware

The Enchantment Whiteboard and Signal system have been used for a range of social networks and group dynamics research projects:

Ivan Chardin used the Enchantment middleware as the “cornerstone” of his software architecture for his masters thesis, *Spatial Aspects of Mobile Ad Hoc Collaboration*[7].

---

between the “sitting” and the “riding the subway seated” classes, but with a long latency (> 5 minutes) due to the dual-layer structure and the need for relatively long Gaussian mixture model classification strings. A better approach might be to train a HMM directly, but the dual-layer model also allowed for immediate use of the intermediate classification information.

<sup>3</sup>I think this is a really interesting idea. Unfortunately, I never went beyond the Matlab stage of modeling this system. Also, the whistle model that I trained would be unlikely to generalize to others; It turns out that I have perfect pitch production when whistling, and always whistle the same base octave between 1000 and 2000 Hz. As a result, my classifier was tuned for particular notes, whereas a more general system would use pitch change (up or down). This work has not been published.

Jonathan Gips used the Enchantment whiteboard as the basis of the “Opinionmetrics” project, a PDA-based classroom study in which Sloan school students used a wireless PDA during a lecture to provide real-time feedback and comments to the lecturer.

Nathan Eagle used the Enchantment Signal system for his “Reality Mining” project, in which a classroom of students were instrumented with PDA-based close-capture microphones and engaged in lecture and discussion. Each student’s audio was timestamped and wirelessly streamed to a central sever using the Signal system.

Michael Sung has used the Enchantment Signal system to capture and stream ECG and GSR data for visualization to provide lecturers real-time biofeedback for managing stress during presentations.

#### **4.4.2 Biomedical Applications of the MITHril Real-Time Context Engine**

The activity classification capabilities of the MITHril Real-Time Context Engine are not limited to context awareness for memory support. The system has also been used for soldier medicine and physiological monitoring research.

##### **Natick Labs/USARIEM Soldier Medicine Research**

We were invited to collaborate with the United States Army Institute of Environmental Medicine (USARIEM) based at Natick Labs on soldier physiology research due to our expertise with wearables and the clear structure of the MITHril Real-Time Context Engine.

In this project we collaborated with USARIEM physiologists to instrument Army Ranger volunteers participating in a hypothermia study. The protocol employed accelerometers, an R-wave (heart beat) detector, and skin/core temperature sensors. The data was gathered through a combination of MITHril 2003/Enchantment middleware and conventional physiology monitoring hardware and National Instruments Labview. We used the MITHril Enchantment middleware and Real-Time Context Engine to develop a real-time shiver detector and core-body-temperature classifier using accelerometer data and HMMs. We were also able to correlate changes in heart-rate variability with shivering, suggesting that a

predictive model for shivering could be created using inter-beat interval (IBI) and core temperature features.

### **Lucy Dunn's Haptics Stress Feedback Research**

Lucy Dunn, a researcher at Cornell, is using MITHril 2003 hardware and the Enchantment middlewear to investigate the use of a wearable haptic feedback system for managing stress. Lucy is extending her previous work with haptic interfaces[81] to integrate ECG and EMG sensors and real-time physiology classification into clothing. The goal of the research is to create a wearable, clothing integrated biofeedback system with haptic output.



# Chapter 5

## Wearable Human Computer Interaction for Memory Support

### 5.1 Introduction

Unlike desktop HCI, in which the interaction designer can rely on high-bandwidth interaction techniques with the expectation of a safe environment and few competing stimuli, interaction design in the wearable computing domain must take into account potentially hazardous environments, limited interaction resources, and other tasks and stimuli competing for the user's attention.

Wearable computing applications are frequently intended to provide support for complex real-world activities, from food inspection[47] to service and maintenance[80] to military command and control (SAAB WISE project, US Army Objective Force Warrior program). Unfortunately, these task-support applications of wearable information systems present special interaction difficulties because they divide the user's attention between the real-world task and the wearable interaction.

The Memory Glasses is a task-support system intended to support the user's memory with just-in-time cues. As such, issues of divided attention must be addressed. Further, the probabilistic nature of context classification means that reminders must be delivered in the absence of certainty. And even when reminders are delivered correctly, Ockerman's work suggests that over-reliance problems can result.[50]

In this chapter I describe the personal experiences and research that lead me to develop the minimalist, cue-based memory support interaction strategy for the Memory Glasses, and the lessons learned along the way that shaped my thinking about wearable interaction in general.

## 5.2 From Command Line to Cue

When I started using the Lizzy[77] in 1999, the command line was the primary interface. Although it was possible to run X-windows applications (a conventional pointer-based direct-manipulation graphical interface), the consensus among the MIT 'borg was that due to the limitations of the monochrome display and a less-than-optimal tilt-sensor based pointing device in the Twiddler, this wasn't worth the trouble.

Because the Lizzy was a Linux-based machine supporting primarily text-mode applications and I was a skilled Linux user, command-line and text mode interactions suited me quite well. For me the primary challenge in adapting to the use of the wearable lay in learning the Twiddler one-handed chording keyboard. I achieved basic alphanumeric touch-typing within weeks, and was fluent in most common characters and frequently-used macros within months.

When I began to use the Lizzy I did not have any reason to believe that the Lizzy's interface was in any way optimal (or even particularly well-suited) for a wearable. It was comfortable enough for my purposes, but also clearly a holdover from a desktop interaction. Further, it was an interface that was obviously intended for experts, highly motivated users who would spend the effort to learn the somewhat arcane syntax of the bash shell and Emacs text editor.

Many members of the wearables group believed that wearables were the next big thing, and would eventually be used by everyone<sup>1</sup>. However, I felt it unlikely that bash shell and Emacs editor would become the vernacular interface for wearable computing. Some sort of graphical direct-manipulation "desktop" with context-driven interface features would

---

<sup>1</sup>Given the large and growing popularity of cell phones and PDAs, this prediction has largely come to pass, though not in the way that many of us were expecting.

surely evolve to make wearable computing accessible to the masses, just as Moore's law would make the hardware small, light, and cheap<sup>2</sup> As time went on and I worked to build the context-driven graphical desktop we all knew was coming, I continued to depend on the command-line and screen-based text applications. Eventually I came to realize that there were important reasons why the much-maligned command-line interface was working, and (as importantly) the conventional direct-manipulation interface was not.

### 5.2.1 Direct Manipulation, Distraction, and Memory

I initially assumed that the difficulties I had with direct manipulation on the Lizzy were the result of the limitations of the interaction hardware. However, as I evolved the hardware platform these limitations became less of an issue. Even with a full-color display and improved pointing device my experiments with a direct-manipulation graphical interface on the wearable proved to be awkward at best.

I began to realize that my dislike of the direct-manipulation interface for wearable interaction wasn't simply due to the limitations of the Lizzy's display or the Twiddler's pointer. It had to do with the role of the wearable interaction as a secondary task, something that could be interrupted at any time, and the nature of the direct-manipulation interface itself.

When typing a command on the Twiddler, an interruption wasn't terribly costly. I could shift my attention away from the interface and back again, and easily pick up where I left off. The partially typed command, still visible on the head mounted display, provided important information as to what I was doing when I was interrupted. Likewise, if my attention were diverted for an extended period of time, the shell's command history provided a kind of episodic memory support that allowed me to reconstruct the interrupted task. Partially typed notes or other text in an Emacs buffer served much the same purpose.

However, when using a direct manipulation interface much less visual state was encoded in the interface. Even for an operation as simple as a "drag and drop," an interruption would often require me to start over from scratch. First I would have to find the mouse

---

<sup>2</sup>As it turned out, most chipmakers were interested in making their processors faster and more power-hungry, rather than smaller and lighter. Just because a problem can be solved by Moore's Law doesn't mean that it will.

pointer again, then remember which icon or widget I was interacting with, and only then could I continue the operation. To make matters more difficult, few applications encoded much in the way of history in the graphical state of the interface. This was much the opposite of the command line or text buffer, and made answering the “what as I just doing” question significantly harder to answer.

In addition, I found that the hand-eye coordination task of pointer manipulation incompatible with many other tasks, even something as simple as walking. By comparison typing on the Twiddler was easy, and appeared to interfere only with talking or other linguistic production tasks. By the time I was fluent in typing on the Twiddler, I could simultaneously walk, pay attention to a conversation, take notes with the Twiddler, and generally function in a normal fashion<sup>3</sup>.

The direct-manipulation interference effect wasn’t related to the complexity of the task. A comparatively simple “drag and drop” operation in the direct-manipulation interface would literally stop me cold, whereas a more complex operation involving the command line or Emacs text editor was comparatively easy to manage.

In order to understand this effect, I began to research the question of wearable interaction, starting with the more familiar literature of wearable HCI, proceeding to the literature of avionics systems design and driver attention studies, and eventually ending up in the cognitive science literature of perception and attention.

A summary of this research can be found in Appendix A.

## **5.2.2 Rethinking the Interaction**

My work developing the MITHril hardware platform began in large measure as an attempt to address some of the grosser physical human factors problems relating to wearable interaction. However, as time went on and I became more familiar with the problem I became convinced that the core problems were cognitive and perceptual.

Using the same head mounted display and Twiddler keyboard, I could construct divided

---

<sup>3</sup>Friends who are familiar with me in my “distracted cyborg” mode may beg to differ, but there is still a substantial difference between “distracted cyborg” and “incapacitated cyborg,” which was the state resulting from a direct-manipulation interaction.

attention interactions that were either trivial to manage or essentially impossible, depending on what was being asked of the user and the style of interaction.

Although it was clear that conventional pointer-based direct manipulation interfaces were some of the most difficult to manage in a divided attention condition, it was also true that they had features that were positive (easy to learn, clear mappings between controls and operations, already familiar to users of desktop and hand-held computers *etc.*). This raised the question of whether it would be possible to create an style of interface that would capture the positive features of both direct manipulation and the command line, and to make this work in divided attention interactions.

How to create a more effective interface was not an idle question for my Memory Glasses research, since I wanted an interaction framework for scheduling and delivering reminders that would be easy to learn, easy to use, and not unduly interfere with the user's other "real world" tasks. Thus I began a process of rethinking wearable interaction that began with my discomfort with conventional direct-manipulation and command line interfaces, was informed by a deepening understanding of perception and cognitive science, and eventually led to the development of the modified HCI models and the five principles of low-attention wearable interaction I describe below.

### **5.3 Wearable Interaction: Extending Existing Models**

The problems I experienced with conventional direct-manipulation interfaces for wearable applications could be framed in light of a straightforward competing-tasks[26] hypothesis: The complexity of the visual and manual dexterity tasks required to use a pointer-based direct manipulation interface was interfering with other tasks I needed to perform in the real world, and vice versa.

My own experience with wearable interfaces supported this hypothesis. In addition, there is a substantial body of research to suggest that the competing task of cell phone use by drivers (or, indeed any divided attention state) has a measurable impact on driving performance[35, 43].

### 5.3.1 The Seven Stages of Action

In “The Design of Everyday Things[49],” Donald Norman proposes the Seven Stages of Action model of human-system interaction. In this model, the user and the world (including the system with which the user is interacting) interact through a process of execution and evaluation. To quote Norman:

The basic idea is simple. To get something done, you have to start with some notion of what is wanted — the goal that is to be achieved. Then, you have to do something to the world, that is, take action to move yourself or manipulate someone or something. Finally, you check to see that your goal was made. So there are four different things to consider: the goal, what is done to the world, the world itself, and the check of the world. The action itself has two major aspects: doing something and checking. Call these *execution* and *evaluation*.

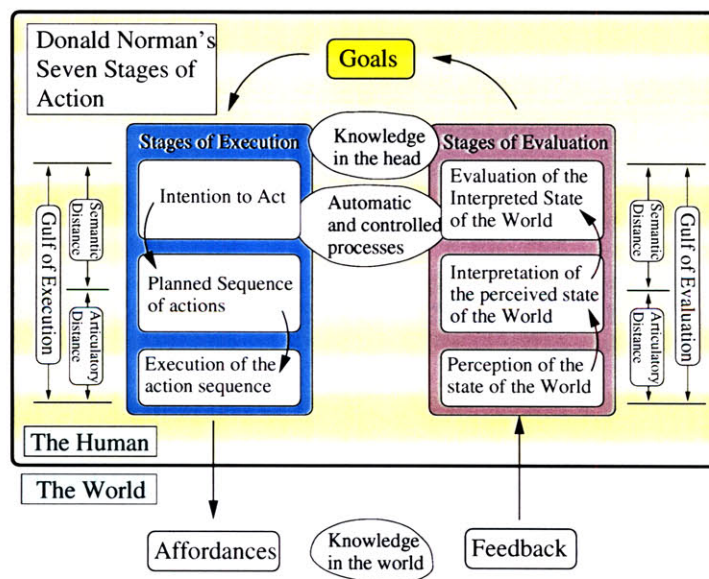


Figure 5-1: Donald Norman’s Seven Stages of Action model of human-system interaction.

Norman’s model is schematically illustrated in Figure 5-1. Between the user’s goals and intentions and the real-world system to be manipulated are two gulfs: the gulf of evaluation and the gulf of execution. In Norman’s model, good interfaces bridge these gulfs by making

evaluation and execution easy. This is the goal of the classic WIMP (Windows, Icons, Mouse, Pointer) direct-manipulation interface for desktop computing.

### **5.3.2 Seven Stages of Action in “Wearables World”**

Norman’s model is an excellent analysis of the cognitive and perceptual factors that influence the usability of systems. However, it implicitly assumes the human is engaged in one primary task, which is to say that the execution/evaluation feedback loop will remain unbroken. This is not a valid assumption when designing interactions for wearable systems, where the user can be distracted by real-world stimuli at any time. Further, these interruptions may force a task switch, where the user must abandon a partly-formed evaluation or execution process in order to deal with more pressing issues, only to return after some significant amount of time has passed.

Second, it assumes that there is nothing mediating between the human and the feedback and affordances of the real-world system. This assumption is appropriate for most desktop computing systems, but is not valid for users of a range of wearable technologies, from head mounted displays to the gear more commonly worn by fire fighters, divers, astronauts on EVA, workers in cold environments, *etc.*.

I have extended Norman’s model to take into account the divided attention and mediation effects resulting from the use of wearable technology. Although this analysis is intended primarily for the type of proactive/interactive wearable technology envisioned by this thesis, it could be equally well applied to interactions with other types of encumbered users, such as SCUBA divers and fire fighters. This model is schematically represented in Figure 5-2.

The mediation effect is the result of two factors: One, the baseline physical and perceptual encumbrance of wearing the system in a passive state, *e.g.* most existing head-mounted display technologies physically encumber the user and restrict their vision even when turned off. Two, the additional physical and perceptual encumbrance resulting from system interaction — such as the loss in manual dexterity resulting from using one hand to type on a chording keyboard, or attempting to dial a cell phone while driving.

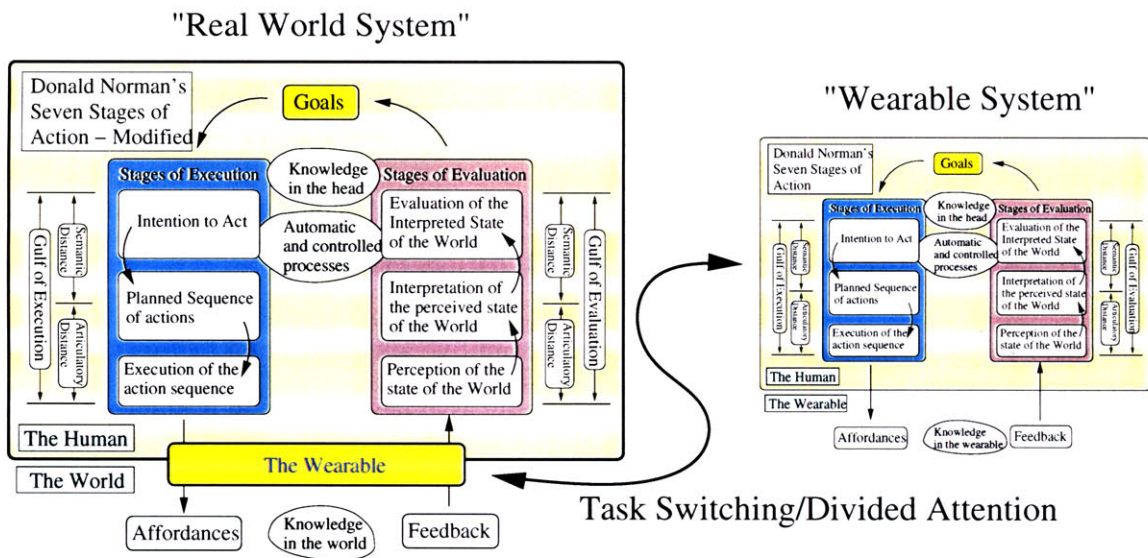


Figure 5-2: Modified Seven Stages of Action model representing both the mediation and the divided attention state produced by the use of an interactive wearable computing system.

### 5.3.3 Implications of the Modified Norman Model for Wearable Interaction Design

There are several implications of the modified Seven Stages of Action model that I propose.

1. Assume the encumbrances of the wearable system will interfere with the user's ability to evaluate and execute real-world tasks. Therefore, minimize the physical and perceptual encumbrances imposed on the user by the wearable system in its passive state.
2. Assume the user is engaged with a primary real-world task that takes priority over the wearable interaction, and can interrupt them at any time. Minimize the duration and frequency of interaction with the wearable system, so as to minimize divided attention and task-switches.
3. Bridging the Gulf of Evaluation: Assume the user can be interrupted in the evaluation process, and that the longer evaluation takes, the more likely this is to occur. Make it as fast as possible for the user to evaluate the state of the wearable system. Maximize the clarity, timeliness, and egocentricity<sup>4</sup> of the information presented to the user<sup>4</sup>.

<sup>4</sup>Egocentricity in this context means presenting information "from the point of view" of the user. For

Minimize the amount of visual search and interpretation required to evaluate the state of the system.

4. Bridging the Gulf of Execution: Assume that the user can be interrupted in the execution process, and that the longer execution takes, the more likely this is to occur. Maximize the clarity of interface affordances, provide simple mappings between goals and actions, and make actions sequences short, ideally single-step. If action sequences must be multi-step, assume they will be interrupted in mid-sequence and make them recoverable by providing adequate interface state.

The final point about making action-sequences recoverable in the face of interruption explains why command-line interfaces may have a higher usability for expert users than direct-manipulation interfaces in a high-distraction environment; Partly typed command lines (and command histories) make for recoverable action sequences, whereas the lack of visual state in many direct-manipulation interfaces puts more burden on the user's memory and makes action-sequence recovery more difficult.

### 5.3.4 GOMS Models

GOMS Models are a class of interaction model first proposed by Card, Moran, and Newall in 1983[5]. GOMS stands for Goals, Operators, Methods, and Selection Rules. GOMS models are distinguished from more general, cognitive models (such as Norman's Seven Stages of Action) by being engineering models that can be used to make specific predictions about an interaction task. The four elements of a GOMS model are:

**Goals** A goal is something the user wants to accomplish.

**Operators** An operator is an action that can be performed in the service of a goal.

**Methods** A method is a sequence of operations that accomplish a goal.

---

example, Wickens[89] shows that users take less time evaluating moving-map displays if the user's vehicle is represented as a fixed delta about which the map rotates, vs. presenting the same information as a map of fixed orientation with a rotating delta to represent the heading of the vehicle.

**Selection Rules** If more than one method is available, a selection rule is used to choose among them.

Because methods can have subgoals, GOMS models are often hierarchical. For example, a goal might be to eat breakfast. A method might be to order breakfast at Sound Bites, a restaurant in Davis Square, Somerville. However, before one can order breakfast, one must first have sufficient cash. Getting cash is a subgoal of the eating breakfast at Sound Bites method, for which there are a variety of plausible methods.

### **5.3.5 GOMS Models in “Wearables World”**

GOMS models are intended to represent specific interaction tasks, but one can usefully apply the GOMS methodology to the analysis of the divided-attention, wearable-mediated interaction model I proposed in the context of the Seven Stages of Action. From this standpoint, wearable interaction goals are subgoals of the “real world” system interaction goals. This has several important implications:

1. Because wearable goals are subgoals of a larger “real world” interaction system, this implies that the introduction of a wearable will generally complicate (rather than simplify) a real-world interaction task.
2. By extension of the previous point, wearable interaction goals should be few, clear, and relevant to “real world” goals.
3. Because the wearable interaction is already part of a larger and more complex interaction, operators should be few, simple, and ideally context-driven.
4. Likewise, methods (action sequences) should be as simple as possible, and contain few subgoals.
5. Selection among methods should be easy. Methods should be as few as possible, and method selection should ideally be context-driven and automated to the extent appropriate.

One way to introduce a wearable system into the larger “real world” interaction model and not complicate the larger task is to create a wearable interaction in which all operators and method selections are context-driven. This is the GOMS interaction model for reminder delivery in the Memory Glasses, where the user interacts with the world about them as normal, and these real-world interactions provide the context for the automatic delivery of reminders.

## **5.4 The Five Principles of Low-Attention Interaction for Wearable Computing**

Based on the modified Seven Stages of Interaction model and the GOMS model analysis above, I propose the following interaction strategy for wearable applications. The overall goal is to minimize the wearable’s physical, perceptual, and cognitive burdens on the user while still providing effective task support.

I formulated this strategy as follows:

1. Design wearable hardware, peripherals, and interfaces that minimize the physical and perceptual encumbrances on the user while still providing appropriate affordances and feedback for the application.
2. Design systems that initiate interactions with the user only when necessary. Minimize the frequency and duration of these interactions so as to minimize the likelihood of distracting the user from important “real world” interaction tasks.

Make appropriate use of context awareness to reduce the need for explicit user interaction. Do not ask the user to explicitly provide what can be easily classified from sensor data or inferred from past history or a user profile.

3. Design interfaces that are easy and quick to evaluate. Do not assume that the user has time to perform complex visual search tasks or remembers where they are in an action sequence. Make all relevant interaction state, including intermediate action-sequence state, immediately accessible.

Maximize the relevance, timeliness, and clarity of displayed information. Minimize the need for interpretation wherever possible. Maximize the egocentricity of displayed information — always strive to present information from the user’s perspective. Leverage awareness of context to make this possible.

4. Design interfaces that simplify and discretize operations, that make affordances clear, that provide clear mappings and represent all relevant intermediate interaction state. Strive for the fewest appropriate operations and methods, but do not over simplify. “Easy things should be easy, hard things should be possible.” Wherever appropriate, leverage awareness of context to drive method selection and operations. Expect the user to be interrupted mid-operation and make graceful recovery possible.
5. Avoid pointers. Avoid hidden interface state. Avoid non-salient interface changes. Never assume the wearable interface has the user’s undivided attention.

In my own work, the first principle of this strategy is addressed in the human-factors driven design choices of the MIThril 2000 hardware , described in Chapter 3. The second principle, that of avoiding interaction when possible and limiting the duration of interaction, is addressed in the Memory Glasses by driving all reminder delivery through context, and requiring the user to provide nothing beyond the initial scheduling of the reminder. This principle relates to the design of the prototype Memory Glasses applications, discussed below.

The third principle relates to the gulf of evaluation, and this is addressed in the design of the prototype Memory Glasses applications and in the Anduin window manager, described below. The fourth principle relates to the gulf of execution, and this is primarily addressed in this work by the design of the Anduin window manager and the context-driven behavior of the Memory Glasses application.

The fifth principle is a list of things to be avoided. Perhaps the only one that needs explanation is the avoidance of pointers; Pointers usually require a visual search task to locate, and encode very little interaction state. Manipulating pointers usually involves a complex hand-eye coordination task that interferes with other visual and manual dexterity tasks, and is generally to be avoided in a wearable context.

## 5.5 Early Memory Glasses Prototypes

A number of Memory Glasses prototypes were developed during my doctoral research, running on a variety of wearable platforms. The earliest versions were simple proactive information delivery systems, demo applications in which the user's context triggered the delivery of an image or short multimedia presentation. These were not really intended for memory support, and were little different in structure from the Wearable City of News system I helped design for SIGGRAPH '99[73].

### 5.5.1 Medical Applications

With the development of the first MIThril 2000 prototypes, I began work on the first real Memory Glasses application. This prototype differed from previous proactive information delivery systems in that it was specifically intended to support the user's memory. It was also my first attempt to address some of the low-attention design issues outlined above.

At the time we were collaborating with the Rochester University Medical Smart Home, and my advisor encouraged me to consider the medical applications of the Memory Glasses. Medical applications of memory support make sense, since those who's lives could benefit the most from effective proactive memory support are people with substantial memory (amnesia) or recognition (agnosia) problems. I identified five classes of possible users, ranging from the unimpaired to the significantly cognitively or perceptually impaired user.

#### Unimpaired users

This population is comprised of people without medically significant deficiencies in memory or recognition. This is the easiest group to study (for reasons ranging from IRB<sup>5</sup> approval to subject recruiting) though with least innate need for memory support. The most valuable applications for this group are likely to be in specialized task-support areas.

---

<sup>5</sup>IRB stands for Institutional Review Board, or the human-subjects research review committee that is required by longstanding academic convention and U.S. federal law.

## **“Normal” older users**

The learning and recall of information may become slightly harder with age[54], but the higher-level cognitive and memory functions of this group are comparable to the unimpaired group. Interaction design for this group of users may need to take into account somewhat decreased acuity in vision and hearing.

## **Amnesia patients**

Amnesia is loss of memory resulting from brain damage (physical trauma or disease) or severe emotional trauma. Amnesia may occur suddenly or slowly, may effect short-term or long term memory, and may be permanent or temporary depending on the cause. Effective treatment of amnesia depends almost entirely on the cause; amnesia resulting from some types of physical trauma or disease, such as amnesia resulting from the pressure of tumors or swelling brain tissue, may be effectively treated surgically or with drugs. Amnesia resulting from emotional trauma, usually related to a specific event (anterograde amnesia), may be effectively treated by addressing the underlying emotional cause of the trauma. Other types of amnesia, such as that resulting from Alzheimer’s disease (also known as primary degenerative dementia) are progressive and incurable[51, 54].

Patients with chronic short-term memory problems, such as those resulting from certain types of head injuries and the early stages of Alzheimer’s, might benefit from a Memory Glasses-style memory support system. However, the high-level cognitive and perceptual deficits that often accompany amnesia symptoms pose significant design problems, and may make effective use of the technology difficult.

## **Agnosia Patients**

Agnosia is the inability to recognize objects through the senses, ”a normal perception stripped of its meaning.” Agnosia may be subdivided into two types: associative agnosia is a failure of recognition (association with memory) given a properly integrated stimulus, and apperceptive agnosia is a disturbance of the integration of otherwise normally perceived components of a stimulus (disturbances in perceived orientation, inability to make

comparisons and matches, etc.). Agnosia typically effects only a single sensory modality, vision being the most common. A particularly famous (and clinically important) form of visual Agnosia is prosopagnosia or “face blindness” which is the inability to recognize faces. To be properly called agnosia, the recognition problems must be associated with normal or near-normal functioning in other aspects of high-level cognition and perception.

Agnosia patients in general and prosopagnosia patients in particular present an interesting application class because the impairment is so specific. A reliable face recognizer could almost literally replace the damaged brain functionality of the prosopagnosia patient, perhaps allowing the patient to lead a normal or near-normal life.

The modality and type of agnosia has interesting implications for the Memory Glasses interaction design, since the role of the application becomes the recognition of and translation from an unrecognizable perception into a recognizable one. For instance, an implementation for associative prosopagnosia might present the user with both a picture of the recognized face and the name, so that the user might compare the picture with the person and confirm recognition through visual comparison. However, an implementation for aperceptive prosopagnosia might present the user with name and a recorded sample of the recognized person’s voice, so that the user could confirm the identification aurally.

## **5.5.2 Amnesia/Prosopagnosia Memory Glasses Prototype**

Using MIThril 2000 as the hardware platform, I constructed an active-tag based amnesia/agnosia Memory Glasses implementation using the Crystal Tag IR active tag system. This application used a glasses-mounted tag detector, with a “look up” detector for receiving data from overhead location tags and a “look forward” detector that tracked the user’s gaze, receiving data from object or person ID tags.

This version of the Memory glasses was aware of time, location, and social context, though the classification of this context was based on a simple “last seen tag” algorithm that failed to take into account the passage of time. It used the MicroOptical head-mounted display to present the user with image and textual information. An example “screen shot” is shown in Figure 5-3.



Figure 5-3: HMD display from the Amnesia/Agnosia Memory Glasses prototype, showing information about a location in the Borglab (the dev-tty1 workstation) and a person (Sandy Pentland).

The application uses a simple split-window design, where information relevant to location is displayed on the left and information relevant to tagged people or objects is displayed on the right. By default, a picture and name is displayed for every recognized person and location, allowing the user to visually confirm the identification. Additional information (in the form of text) can be associated with a place or person, either to be delivered once or whenever the reminder's distinguished context is matched. Text reminders can also be associated with time.

The goal of this application is to allow a user with significant amnesia or agnosia difficulties to determine where they are and who they are talking to, in addition to providing a basic location-, social-, and time-aware proactive organizer function.

The Amnesia/Prosopagnosia Memory Glasses prototype only addressed the interaction problem of delivering reminder information. Scheduling reminders, mapping tags, and associating images with locations and people was done using a conventional Unix text editor and simple text configuration files.

### 5.5.3 Evaluation

Although the system was intended to be used in a clinical setting (such as the Rochester University Medical Smart Home) the complexities of competing agendas and IRB approval,

as well as the limited availability of MITHril hardware, prevented us from actually running the study.

I experimented with this Memory Glasses prototype as a reminder system during my workdays in the Borglab. I found the simplistic location and social context classification to be a major limitation of the system.

The Crystal Tag system itself worked well enough, but the way the Memory Glasses prototype dealt with the tags was overly simplistic. Not all locations were tagged, nor were all people. The Memory Glasses would acquire tag IDs well enough but the likelihood of location classifications did not decay with time. As a result, the classified location was always the last valid location tag ID received, even if that ID was received hours ago. Likewise, the classified social context was always the last valid person tag detected. This “stale context” problem left the display cluttered with old reminders, and generally made the system less useful than it might otherwise have been.

Another problem was that the Amnesia/Prosopagnosia Memory Glasses prototype monopolized the head-mounted display. While the Memory Glasses application was running, the display couldn't be used for other purposes. For example, although the Memory Glasses prototype was time-aware, in order to display the time of day on the head mounted display it was necessary to first stop the Memory Glasses and run a clock program.

These limitations in context classification and interaction led me to consider what a more correct, general solution might be. To address the context classification problems we developed the Enchantment middleware and the Enchantment Real-Time Context Engine, described in the previous chapter.

I decided that the solution to the interaction problems was to develop a multitasking interaction environment that would support flexible reminder delivery, as well as interaction with more conventional applications. Only with such an environment would it be possible to develop a Memory Glasses system that could support the user's memory without conflicting with other user-support applications (such as a clock, battery monitor, navigation system, *etc.*).

## 5.6 Anduin Window Manager

Anduin<sup>6</sup> was developed under my direction to provide the proper interaction environment for the next-generation Memory Glasses prototype as well as other research applications. The Anduin window manager is a pointerless X-windows window manager designed for use with a small head mounted display. Anduin provides a general-purpose wearable computing interface that would support legacy conventional X-windows and text-based applications as well as context-enabled, wearable-specific applications like the Memory Glasses.

Anduin embodies most of the interaction principles I outlined above, and differs from conventional window managers and desktops in several important ways:

1. Anduin is designed for a small (quarter VGA) display. At the time we started Anduin development, there were no comparable open-source, small-screen, general-purpose window managers suitable for our use, and arguably there are still none<sup>7</sup>.
2. Anduin is a pointerless desktop. There is no need for a continuous pointing device, and all operations that would traditionally require such a pointing device are available through simple button-press interactions.
3. There are no overlapping windows on the Anduin desktop. Instead, Anduin employs a large, central focus area and separate application and applet bars.
4. The Anduin desktop explicitly supports context-driven interaction, and the Anduin window manager is an Enchantment Whiteboard client. All major window manager functionality is available through the Whiteboard interface.

Although Anduin differs significantly in function and behavior from other X-windows window managers in appearance and interaction, it is in all fundamental respects a fully compliant X-windows window manager. This means that Anduin can be run on any X-windows capable display, and supports a range of legacy X applications. Anduin is also

---

<sup>6</sup>The Anduin window manager was written under my direction by Steve Dunn, and is inspired by [33].

<sup>7</sup>The QTopia desktop, developed for Linux PDAs and other small-screen mobile devices, would make a reasonable candidate for evaluation were this project to be re-engineered, but does not support the indirect rendering model needed for the efficient support of the field-sequential-color Kopin display.

configurable, and although intended to run on small displays, can be configured to make good use of much larger displays when available.

### 5.6.1 Evaluation

Anduin has not been formally evaluated in a user study. There are two long-term users of Anduin in a wearable context, David Kaplowitz and myself. Here is what David has to say about Anduin in a recent email:

Well ... it [Anduin] is extremely useful. I had to convince the X server to turn off the silly X pointer for the (not used) mouse, and change the colors, key mappings, and alerts to be something more useful to me, all of which are user-configurable, so quite reasonable, but beyond that, it worked just fine. Reliable, solid, a little slow, but that might have been the interface with enchantment (it worked faster with a local enchantment [Whiteboard] server, as opposed to one, occasionally available, over the net) my default state was one window open to each of the boxes (network, server, and interface) an emacs window with my mail in it open on the server, a second emacs window open for random editing and notes (occasionally running Bradly's [Remembrance Agent] code) a time window (because I hate analog clocks) that had "alerts" in it and a bus-schedule window (another x-term on the server) when I was working on other machines, they would often be in the network box window, not for any other reason, but less overhead.

What I would like (and not everyone agrees) would be better resolution, so I could have more on the screen, and have it still visible (I'd like to be able to do 75 -> 80 columns of text and 24 lines at least more would be delightful) but I don't know if that is doable or feasible. I would also enjoy more context-awareness, switching fonts if I am sitting/walking/running et al. And the memory glasses/reminder, based on face recognition might be nice as well, but ... if we are asking for the moon ... \*grin\*

Early on in its development, Anduin became my daily-use interaction environment for wearable computing and has remained so to this day. Anduin's structured display provides the necessary simplicity to effectively use a variety of applications on the wearable under divided-attention circumstances without undue distraction. It's pointerless, key-driven window management functions allow for simple task switching, and the applet bar is perfect for "ambient" applications like battery monitors and clocks that always need to be present, even though they may rarely be the central focus of attention. The applet and application bar provide the user with all of the visual state needed to find a running application, and any applet or application can be pulled into the central focus with a small number of key presses. The ticker bar at the bottom serves its purpose by providing providing a useful status display, although at present messages posted to the ticker persist until replaced, which can result in stale messages.

I feel that the primary limitations of the Anduin window manager are twofold. First, it has a "flat," TWM-ish visual style that looks out-of-date by comparison to most modern window managers, and at least some users (in a demo context) have said they find this ugly. This is a cosmetic issue, but such issues are of significant importance in usability.

Second, the context-driven features of the window manager, such as the ticker bar and context-driven focus shifting, require cooperative behavior on the part of applications. If applications do not coordinate, race conditions can result. For example, two applications in short succession may send messages to the ticker, resulting in the first message is almost immediately being replaced by the second. Likewise, two applications requesting central focus in short succession will result in the first application almost immediately being shifted out of focus by the second.

The proper solution to this problem is a top-level attention management system that brokers interaction events between applications and the window manager. Such a system could prioritize messages, suppress undesirable activity, and provide higher-level context awareness in the interaction environment without requiring that all applications be individually context-aware and coordinated.

The Enchantment middlewear and Context Engine would provide an excellent foundation for the attention manager. However, attention management system has yet to be



Figure 5-4: A screen grab of the Anduin-based Memory Glasses system delivering an overt cue – the name and project information for the author.

written.

## 5.7 Anduin/Enchantment-based Memory Glasses

With the development of the Enchantment middleware, the MIThril Real-Time Context Engine, and the Anduin window manager, it became possible to construct a much more principled, modular Memory Glasses prototype utilizing the window manager's context-enable features. The core of this new Memory Glasses implementation was the principled context-classification framework and modular IPC system provided by the Enchantment middleware and the Real-Time Context Engine, described in Chapter 4. This Memory Glasses prototype was aware of the following contexts:

1. **Social Context.** Like the previous prototype, this version used the Crystal Tag system and a glasses-mounted tag detector to determine social context. Unlike the previous prototype, a principled tag-classifier was employed that made probabilistic assessments of the user's social context based on the recency of ID tag data.
2. **Location Context.** As with social context, this prototype used the new tag classifier to make a principled classification of the user's location based on the recency of location tag data. This scheme also supported the use of other types of location classifiers,

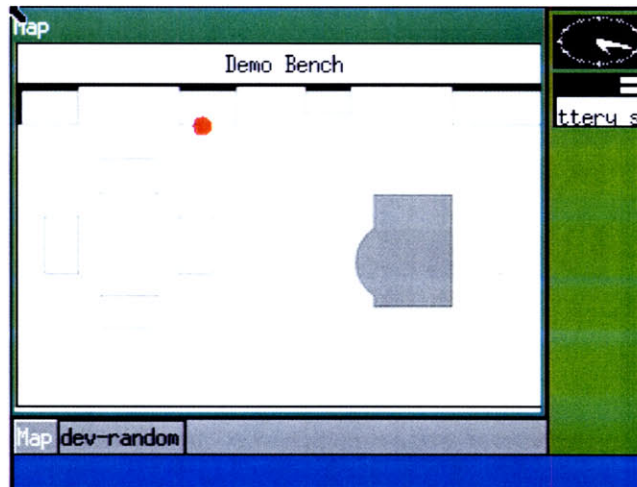


Figure 5-5: A screen grab of the Anduin-based Memory Glasses system showing the user's position on room-level map. The delivery of this information is triggered by the user's action context — walking.

such as a GPS-based location system.

3. Action Context. This prototype used the MIThril Real-Time Context Engine to support the multi-state classification of user activity, for example walking vs. stationary.
4. Time Context. The Enchantment Whiteboard server's clock provided a distinguished time reference for triggering the delivery of time-context reminders

Sitting on top of this framework was the Notifier application. The Notifier triggered the delivery of reminders by interpreted a text-file representation of reminders and matching the reminder contexts against the user's current context as represented on the Enchantment Whiteboard.

The Notifier supported the following overt reminder actions:

1. Raising a window in the Anduin central focus with text or static image content.
2. Displaying text or static image content in the Notifier window, but not raising it in central focus.
3. Sending a text message to the ticker bar, the continuously present text area at the bottom of the Anduin window manager.

4. Executing arbitrary scripts or running other programs, thus providing for the extension of the Notifier's reminder-delivery capabilities.

The Notifier's reminder syntax allowed for qualifications on the delivery of reminders, such as one-time delivery, repeated delivery with timeouts, and multi-modal delivery.

In addition to the Notifier application, a Map application could also be run. Like the Notifier, the behavior of the map was driven by the user's context. The action context of walking would cause the map to be raised, with the user's present location depicted as a red circle on the map. The Notifier and Map applications would cooperate in managing the user's attention, with the Map taking precedence based on the assumption that while walking the user's primary concern is navigation. Only when stationary would the Notifier's window be raised (although other actions, such as the delivery of ticker messages and subliminal cues, were not suppressed).

The cooperation of the Notifier and Map applications represent the first steps towards a more general attention management system (described above) that would prioritize all demands on the user's attention.

### **5.7.1 Evaluation**

This prototype has been successfully used in many Media Lab demos, but no formal user studies have been conducted. I have used the Anduin/Enchantment Memory Glasses prototype for well over a year, and find that it largely meets the goals I set for it. One of the most important features of this system is its seamless integration with the Anduin desktop. I don't have to choose between running the Memory Glasses and other applications, and like the well-behaved low-attention application that it is, it stays out of the way unless delivering a reminder.

I find that the reminder delivery options are adequate, although the visual reminder capabilities (delivering pictures *vs.* text or other types of information) are less useful in my daily interactions than they are when giving demos of the system. In most cases, short text messages displayed on the ticker bar<sup>8</sup> are sufficient, although may not be sufficiently

---

<sup>8</sup>I am embarrassed to admit that I find messages like "Eat" and "Go Home" to be useful reminders while I am working.

attention-getting for high-importance information.

In contrast, I find that raising the Notifier window to central focus to deliver a picture or text-page reminder is a very salient and attention-getting action. This is appropriate for urgent or high-importance information. A third alternative is to send a message short to the ticker and display further content in the Notifier window, but not to raise the Notifier to central focus. This third alternative is appropriate for a wide range of reminders, but at present requires both actions (the Notifier window content and the ticker text message) to be scheduled separately.

Raising the Notifier window to central focus (or any other context-driven window-raise) is not only attention-getting, it is all but guaranteed to interrupt any other ongoing interaction within Anduin. This can be quite annoying. A reasonable solution to this problem would be to construct a simple user-interaction classifier, and to suppress certain types of activity (such as automatic window raises) while the user is typing. Although it would not be hard to build this capability into the Notifier application itself, it should properly be part of the attention manager, and as such has not yet been implemented.

## **5.8 The Limitations of Low-Attention Interaction Strategies**

The design of the overt reminder delivery aspects of the Anduin/Enchantment Memory Glasses was intended to minimize the demands on the user's attention while providing effective memory support.

Reducing attentive demands on the user is a worthy goal, but contemporary theories of attention suggest that any demands made on the user's attention will result in task interference. This is unfortunate, because it implies that task interference is an inevitable part of memory support. Or is it?

My interest in minimizing the attentive demands on the user led me to wonder whether it might be possible to deliver reminders in a way that would not involve the user's conscious attention. In principle such an approach would address the divided attention prob-

lem, because the user’s attention would not be required<sup>9</sup>. And without the user’s conscious attention, they would at least not be distracted by the delivery of reminders triggered by incorrect context classifications, mitigating the reliability problem. Further, if the user were not aware of the support, it was plausible that the over-reliance problems described by Ockerman[50] might be mitigated.

To investigate this possibility, I added support for an additional reminder action: the delivery of extra-short-duration (1/180th second) video frames with cue information.

In the full Anduin/Enchantment Memory Glasses prototype, this capability was present but not well integrated. I was able to demo this functionality<sup>10</sup>, but nearly all users reported perceiving the cues as a flicker or flash on the display, and many could read the cue content.

I believed that with further engineering I could make the short duration cuing truly subliminal, but this did not address the question of whether it would actually work to improve memory recall. Further, although the semantic priming effects of unattended perception were well established, the application to memory support appeared to be almost completely unexplored in the cognitive science or HCI literature.

This led to my research in cuing for memory support, including my investigations of masked short-duration video cues and masked superliminal audio cues. In the next section I present a summary of my subliminal cuing results. More detail can be found in Appendix A, where I provide survey of the relevant cognitive science literature, and Appendix B where I present my experimental work in some detail.

## 5.9 Subliminal UI

In order to investigate the potential usefulness of subliminal cuing for just-in-time memory support, I devised a series of experiments that would allow me to investigate the effects of different types of cues. These experiments shared the same procedure, differing only in the method by which cues were given.

---

<sup>9</sup>This is not to say that other cognitive resource would not be required. However, there are significant bottlenecks associated with conscious thought and perception that do not apply to unconscious perception and cognition. See Appendix A for a discussion of the cognitive science of unattended perception.

<sup>10</sup>...most famously for Alan Alda, on the “You Can Make it On Your Own” episode of Scientific American Frontiers on PBS.

### 5.9.1 Summary of Cuing Experiment Design

In these experiments I investigated three cuing methods: subliminal visual cues, overt visual cues, and masked audio cues. In each experiment I had three experimental conditions: correct cues, misleading cues, and absent cues. There were also three subject conditions (assigning each memorization target to a different experimental condition) to balance the design.

The visual cuing experiments were conducted using the MicroOptical QVGA clip-on display and the MITHril 2000 video driver board as the cue delivery hardware. The audio cuing experiment used a standard mono headphone headset for cue delivery.

The experimental scenario and procedure was common to all three experiments. The subject was given two minutes to memorize 21 names and faces, presented simultaneously in a 7x3 grid on the flat panel display. The faces were unremarkable women's faces chosen from the FERET[55] face-recognition database. The names were chosen from a list of the most popular American women's names of the last 100 years. The name-face pairing were the same for all subjects, but the three subject conditions assigned each pairing to a different experimental condition, a different location in the memorization grid, and a different ordering in the memory trials.

After the memorization period and a brief delay, the subject was presented with 21 memory trials consisting of being shown a single face and having 10 seconds to type the correct name. The subjects were told that they were allowed to change their answer during this 10 second period, but only the last answer would be scored.

Each trial was pre-assigned (based on subject condition) to one of the three experimental conditions. Depending on the condition, the subject would receive correct cues, misleading cues, or no cues. The nature of the cuing depended on the experiment: either 10 masked short-duration video cues, a 10-second long overt video cue identical to the masked subliminal cue, or a white-noise masked superliminal audio cue.

Each subject received seven trials in each of the three experimental conditions, allowing within-subjects analysis for each experiment. Following the memory trials, each subject was given a questionnaire to complete, that included questions about detectability of cues

(in the case of the subliminal cuing experiments), the distraction associated with the cuing system, and the overall difficulty of the task.

This experimental procedure is described in detail in Section B.2.

### 5.9.2 Summary of Results: the Subliminal Visual Cuing Experiment

My hypothesis for the subliminal visual cuing experiment were that the effect of correct cues would be positive, which is to say to improve performance on the memory test, and that the effect of misleading cues would be slightly negative. Further, I hypothesized that although the effect of the misleading cues would be negative, the subject would be no more likely to choose the misleading cue name than any other name in the experiment<sup>11</sup>.

What I found was that correct cues significantly improved performance (a 58% improvement,  $p = 0.003$  single-tail T-test, within-subjects analysis), but that misleading cues did not significantly decrease performance. Indeed, the effect of misleading cues was a borderline significant increase in performance. This unexpected effect could be explained by a spreading activation or associative theory of memory, since the misleading cues were names used elsewhere in the memorization set. Further, I found that for the misleading condition trials subjects were very unlikely to choose the misleading cue.

cond.	mean	effect	effect $\sigma^2$	Student T $p$
“n”	1.71			
“C”	2.71	1.00	2.20	(one-tail) 0.003
“X”	2.19	0.48	1.56	(two-tail) 0.09

Table 5.1: Within-subjects hypothesis testing for masked short-duration visual cues. The “n” condition is the control condition. The “C” condition is the correctly cued condition. The “X” condition is the miscued condition. For further explanation, see Appendix B.

This experiment and the results were published in [13]. Within-subjects hypothesis testing, the magnitude of the effects, and  $p$  values are shown in Table 5.1. This experiment is described in detail in Appendix B.3.

---

<sup>11</sup>See Section B.3 for a discussion of why I expected this.

### 5.9.3 Summary of Results: the Overt Visual Cuing Experiment

My hypothesis for the overt visual cuing experiment was that the effect of correct cues would be positive, and the effect of incorrect cues would be negative. Further, I hypothesized that for misleading cue condition trials, subjects would choose the incorrect cue as the answer more often than other names used in the experiment.

What I found was that correct cues did significantly improve performance (246% improvement,  $p = 0.000000008$ , one-tail T test, within-subjects analysis). However, much to my surprise misleading cues did not significantly decrease performance. And although misleading cues did not decrease performance, subjects were vastly more likely to choose the incorrect cue as the answer than other names when answering incorrectly on a miscued trial. Within-subjects hypothesis testing, the magnitude of the effects, and  $p$  values are shown in Table 5.2 This experiment and its results are describe in detail in Appendix B.4.

cond.	mean	effect	effect $\sigma^2$	Student T $p$
“n”	1.88			
“C”	4.63	2.75	2.54	(one-tail) $8.3E^{-9}$
“X”	2.08	0.21	1.82	(two-tail) 0.22

Table 5.2: Within-subjects hypothesis testing for overt visual cues. The “n” condition is the control condition. The “C” condition is the correctly cued condition. The “X” condition is the miscued condition. For further explanation, see Appendix B.

### 5.9.4 Summary of Results: the Masked Superliminal Audio Cues Experiment

I expected similar results for my masked superliminal audio cuing experiment as for the subliminal visual cues experiment. I hypothesized that correct cues would improve performance and that misleading cues would have a slightly negative effect. Further, I hypothesized that, as in the visual cues experiment, misleading cues would not result in subjects being more likely to choose the misleading cue when answering incorrectly on a misleading cue condition trial.

What I found was that correct cues did not significantly improve performance, nor did misleading cues significantly decrease performance. The incidence of choosing misleading

cues on misleadingly cued trials was at chance levels. Within-subjects hypothesis testing for this experiment is shown in Table 5.3. This experiment and the results are described in detail in Appendix B.5.

cond.	mean	effect	effect $\sigma^2$	Student T $p$
“n”	2.63			
“C”	2.40	-0.22	2.26	(one-tail) 0.76
“X”	2.26	-0.37	2.40	(two-tail) 0.11

Table 5.3: Hypothesis testing for masked superliminal audio cues. The “n” condition is the control condition. The “C” condition is the correctly cued condition. The “X” condition is the miscued condition. For further explanation, see Appendix B.

### 5.9.5 Larger Experiment Analysis

Since the procedure for all three experiments was the same (and no subjects were run in more than one experiment) they can be regarded as one larger experiment, with the cuing method as the experimental condition.

What I found was that subjects in the overt visual cues experiment were approximately 30 times more likely to choose the misleading cue name than subjects of either of the masked cuing experiments.

Likewise, I noticed that while the mean on the “n” control condition was very similar for the two visual cuing experiments (1.71 on the subliminal visual cues and 1.88 on the overt visual cues) it was substantially larger on the masked superliminal audio cues experiment: 2.63. One explanation is that the subjects that I ran in the audio cues experiment simply had a better memory than the subjects in the overt cues experiment. However, these subjects were recruited from the same population and in the same way as the previous two experiments, so this explanation seems somewhat unlikely. Another possibility is that the masked superliminal audio cues did have an effect, but that the effect “spilled over” into the control condition trials. This would explain why this group’s performance on the control condition is similar to the performance of the other subjects on the correctly cued condition in the masked visual cues experiment. This may be possible due to the difference in decay rates of visual and auditory short-term memory – see Appendix B.6 for more discussion.

## 5.9.6 Implications

None of my experimental results suggested that cuing of any kind significantly interferes with the memory recall task. This suggests that if the only consideration is accuracy, it is safe to deliver masked or overt cues even when the classification results are less than certain.

Both subliminal and overt visual cues appear to significantly improve performance, with the possible advantage that subliminal cues may distract the user less and are vastly less likely to mislead them when they aren't sure of the answer.

Although misleading overt cues appear not to negatively impact accuracy, it is clear that users are much more likely to guess the name suggested by a miscue than other plausible names. This supports the idea that although users that are sure of their own memory won't be fooled by a faulty overt cue, unsure users may very well be misled. Although different from the missing-item-on-the-checklist phenomenon described by Ockerman, this is another type of over-reliance effect.

In general, these experimental results suggest that both overt and subliminal visual cues may be good choices for just-in-time memory support. Subliminal cues may be preferable in situations in which minimizing distraction is important, or in which the confidence in the classification that would trigger cue delivery is low.

Another interesting aspect of subliminal cuing is that it appears to support the user's memory by a different mechanism than overt cues. A Memory Glasses system delivering an overt cue is a memory surrogate. A Memory Glasses system delivering a subliminal cue may be functioning more as a memory aid. The user must still recall the information, the subliminal cue just makes it more likely that the recall will succeed.

The failure of the masked superliminal audio cuing experiment to show significant results for either improving or interfering with memory is an interesting contrast to the subliminal visual cuing experiment. One possible explanation is that the combination of cue and mask was simply ineffective, but modality effects may have also come into play. The memorization targets were presented visually, and the cues were delivered aurally. A revised experimental protocol could easily be developed to address this question. Another

possibility is the “spill over” effect that I described above, where the cues were having an effect but that the effect wasn’t confined to the experimental condition trials. Again, a revised protocol could be developed to investigate this effect.



# Chapter 6

## Intellectual Contributions

The full scope of work documented in this thesis includes important contributions from other researchers. Many thousands of engineering hours of work and many good ideas have been contributed by dozens of others, many of whom are referred to by name throughout this document. The scale of the engineering undertaking involved in the MIThril, Enchantment, and Memory Glasses work would not have been possible without this help. I have been extremely fortunate to work with such a talented and motivated group of collaborators.

In general, I have played a leading role in the work described here, with the exception of specific projects in which I have worked with others who are identified by name for their contributions. However, because this work was a collaborative effort, it is necessary to identify those areas of specific intellectual contribution that I claim as my own in this thesis research. This chapter summarizes this contribution, with references to the detailed descriptions elsewhere in this document.

### **6.1 Hardware Architectures for Wearable Computing: MIThril 2000/2003**

My first major area of contribution is in hardware architectures for wearable computing. As the lead architect of the MIThril system, I led the development of the first modular, distributed, clothing-integrated multi-computer architecture for wearable computing research.

In addition to being the architect and lead designer, I also did much of the actual hardware engineering for MITHril 2000, including the design of the Kopin video driver — the most complex single piece of custom hardware developed for MITHril thus far.

Previous commercial and research wearables were monolithic designs in hard packages. I hypothesized that substantial improvements in wearability could be achieved by distributing the computing, interaction, and sensing infrastructure through the clothing itself. Further, I hypothesized that this was possible by integrating existing printed circuit board and connector technology with conventional textiles, and that resorting to exotic textile-based computing[57] was not necessary to achieve this goal.

Likewise, the monolithic design of previous commercial and existing wearables provided limited flexibility and scalability. I hypothesized that substantial gains in flexibility and scalability could be achieved by creating a wearable multi-computing architecture in which decentralized computation, sensing, and interaction resources could be combined as-needed for particular applications.

Further, I hypothesized I could design a smaller, lighter-weight, lower-power video driver for the MicroOptical QVGA that would substantially reduce the size and power-consumption of the overall MITHril 2000 architecture for configurations requiring a head-mounted display.

### **6.1.1 Clothing Integration without Textile-based Computing Hypothesis**

The engineering of the MITHril 2000 prototypes proves that clothing integration of complex, full-featured, wearable computing, networking, sensing, and interaction systems is possible using conventional electronics, cable, and connector fabrication techniques. This is important, because true textile-based computing (in which textiles are used as printed circuit boards, wiring, or display and sensing elements) is still exotic technology with limited applications at present. In particular, textile-based power buses[45] and sensors (such as capacitive touch/proximity sensors[57] and piezoelectric elements[19]) are possible, but fully-integrated textile “printed circuit boards” and textile-compatible chip packages are

not yet available.

### 6.1.2 Wearability Hypothesis

The clothing-integrated design of MITHril improved wearability over the Lizzy and other monolithic computer designs in several important ways. First, the clothing-integrated aspect of MITHril 2000 makes it compatible with a range of conventional business and casual attire. I was able to wear and use packaged MITHril 2000 systems unobtrusively and comfortably in a range of social circumstances that would not have been possible with a conventional industrial wearable (such as the Xybernaut) or research wearable like the Lizzy. For example, I was able to wear my MITHril system with business attire to meetings and formal dinners without violating the implicit dress code of these events and without attracting undue attention.

Second, the MITHril 2000 clothing-integrated design is compatible with a range of daily activities, from sitting, standing, and walking, to wearing a backpack, running (for short distances), and lying down — without significant reconfiguration or shifting the gear, without awkward bouncing during athletic activities (a problem with satchel computers like the Lizzy), and without the contact or motion-range discomfort of a monolithic “brick” worn on the belt. This property is the result of the low-aspect ratio of the individual MITHril 2000 components and the placement of these components on the body, which are kept in place by the fabric “chassis.”

Third, the packaged MITHril 2000 system is compatible with a range of clothing and conventional body-worn gear, such as pagers, cell phones, *etc.*, that are worn on the belt or carried in a backpack. Conventionally packaged monolithic wearables (such as the Xybernaut, the Via, *etc.*) are typically worn on the belt, which can be incompatible with tool belts or other belt-worn equipment. Likewise the Lizzy satchel packaging can make wearing a backpack more difficult, since the shoulder strap must either be worn over the back pack straps (requiring that the Lizzy be taken off to remove the backpack) or underneath, which makes shifting the Lizzy to a more comfortable position for sitting or walking difficult. Put another way, the clothing integrated design is compatible with other types of gear that are

intended to be worn over or with conventional clothing.

Fourth, the clothing packaging and positioning of the MITHril 2000 gear provides compatibility with conventional protective outer wear, such as winter coats and rain jackets. Although we have not tested MITHril 2000 with technical protective outer wear, it should be compatible with occupational protective gear as well, such as the thermally protective outer wear worn by fire fighters or body armor worn by soldiers. Compatibility with protective outer wear is a not a feature of conventional wearable computers, which typically must be worn over outer wear in order to be comfortable. I have worn my MITHril system in the rain, snow, and extremely cold weather using conventional, unmodified outer wear to protect both the system and myself.

And finally, although the individual MITHril 2000 components are not packaged for impact resistance, the positioning of the MITHril 2000 gear on the user's torso provides at least some protection for the system. This was demonstrated in an inadvertent experiment involving a bike crash. Because the gear and cables are located on parts of the torso that I instinctively protected from contact with the ground, the resulting fall and roll did not damage the wearable or interrupt data collection.

### **6.1.3 Scalability Hypothesis**

My hypothesis that a modular, distributed, sensing and computing wearable architecture would provide gains in flexibility and scalability has been demonstrated by the range of MITHril research applications. The same sensors and interconnects have been used in configurations ranging from microcontroller-based data-acquisition systems to wearable Linux computing clusters. This is an important feature of the MITHril architecture that is not present in conventional monolithic wearable computing designs.

### **6.1.4 The MITHril Body Bus and Body Network**

The MITHril body bus is an integrated multi-protocol power/data bus that provides interconnections between computing/logging resources and sensors/peripherals. The Body Bus provides the foundations for the scalability of the MITHril architecture, allowing the

same set of microcontroller-based I<sup>2</sup>C-enabled sensors and peripherals to be used in wearable configurations ranging from simple microcontroller data-acquisition systems to Linux-based wearable multicomputers. This scalability in configuration, from light-weight, inexpensive, “Sociometer-style” configuration to full-blown “Memory Glasses” configuration is not possible with a monolithic architecture.

The MIThril body network is a power-over-Ethernet wired network that makes for easy interconnection between computing nodes on the body through a low-power, peer-to-peer Internet Protocol 10 Mbit network. The MIThril body network enables a distributed multi-computing architecture, allowing the combination of heterogeneous computing nodes on the body tailored to specific tasks. This is in contrast to monolithic wearable computing designs that require a single computer to perform all on-body computing, networking, storage, and user-interaction functions.

MIThril’s highly modular architecture also makes it easier to troubleshoot and upgrade than a monolithic design, since potentially-defective or obsolete components can be replaced piecewise. Likewise, MIThril 2000 components can be upgraded incrementally without a total system redesign.

There is a cost to MIThril’s distributed multi-computing design, in that every interconnect has an energy cost that is proportional to the switching frequency and capacitance of the signal line[2] ( $P \propto f \cdot V^2C$ , where  $P$  is power consumption,  $f$  is switching frequency, and  $C$  is capacitance of the signaling line). Because MIThril components are distributed across the body, signal lines between components are on the order of 1 meter in length, as opposed to the centimeter lengths typical of more integrated designs, resulting in increased capacitance. Also, the copper cables and ruggedized mechanical connectors add to the total system weight.

However, both the Body Bus and Body Network featured a unified regulated 5V power bus, with the Body Bus additionally supporting a unified unregulated 8-16V rail. The unified MIThril power rail partly offsets the increased weight of the cables and connectors because it allows for the use of lighter, smaller, battery-free components and provides natural load-balancing across the system. The load balancing maximizes system uptime while minimizing battery weight due to the non-linear properties of Lithium-ion batteries[42].

## **Supporting Evidence**

Supporting evidence for the wearability and scalability of the MIThril architecture, and of its overall usefulness and importance, is provided by the success of its in a range of research projects here at the Media Lab, as well as at other academic and industrial research settings (Cornell, SAAB, and Rochester University, to name a few). For specific examples see Sections 3.6 and 4.4.

### **6.1.5 MIThril 2000 Kopin Video Driver Engineering**

I hypothesized that we could substantially reduce the weight and power consumption of the MicroOptical head-mounted display by reengineering the video driver hardware. Although this proved to be quite difficult, I was successful in designing a substantially smaller, lighter-weight, more flexible and better-performing video driver for the MicroOptical QVGA clip-on as part of my MIThril hardware engineering effort.

At the time I designed the MIThril 2000 Kopin driver, the only alternative for driving the MicroOptical head-mounted display was the NTSC/VGA signal-path reference display driver hardware produced by MicroOptical. The reference design burned nearly 3W of power (as measured on our bench supply) and provided complex analog to digital signal conversion path that produced display images with less sharpness and clarity than the underlying Kopin display could provide.

The MIThril 2000 Kopin driver, by contrast, is not an NTSC or VGA analog signal path display driver. The MIThril 2000 Kopin driver interfaces directly with the LCD driver provided by the MPC823 microprocessor on the BrightStar Engineering ipEngine1 single-board Linux computer. It utilizes the Altera FPGA on the BrightStar to synthesize digital display timing signals<sup>1</sup> and produces level-shifted, impedance-matched signals appropriate for the MicroOptical display's microcoaxial cable and display. The MIThril 2000 Kopin driver is a single-board design the size of the BrightStar ipEngine1 (with which it mates), adding very little to the size of the BrightStar. The reference MicroOptical driver is a multi-board design that is packaged in a box several inches on a side, significantly increasing the

---

<sup>1</sup>The FPGA core and video driver code was primarily written by Ed Keyes.

size of any wearable with which it is packaged. The MIThril 2000 Kopin driver burns approximately 300 mW as compared to the approximately 3W dissipated by the reference MicroOptical design.

The MIThril 2000 Kopin driver hardware also provided complete control over the timing of the MicroOptical display backlights. This allows the brightness of the display to be adjusted in software, and for the display to be shifted from 180 Hz grayscale to 60 Hz field-sequential-color with a single command. Software control of display brightness and color was not possible using the reference design hardware. The precise control over backlight timing also facilitated the generation of extra-short duration video fields, an important feature for my subliminal visual cuing research.

## 6.2 Software Architectures

My second major area of contribution is in software architectures for distributed, multi-node and multi-user (group) context-aware wearable computing applications, and in particular the Enchantment middleware and the MIThril Real-Time Context Engine. Although I am the architect and lead designer of these systems, most of the programming was done by others, notably Jonathan Gips, Steve Dunn, and Daniel Barkalow. These systems are described in detail in Chapter 4.

I hypothesized that a whiteboard-style system would be most appropriate architecture for network-transparent interprocess communications on the distributed computing MIThril architecture. The whiteboard communications model transforms an  $O(n^2)$  point-to-point communications problem into an  $O(n)$  problem, and provides an asynchronous and simplified publisher/subscriber communications interface. However, context-aware applications require the streaming and processing of sensor data in real time, which could potentially overwhelm a centralized communications model. Thus, I hypothesized the need for an additional peer-to-peer communications channel specifically for streaming sensor data and features.

Based on this interprocess communications architecture, I hypothesized that it would be possible to implement a sensor data classification system that was both principled and

capable of real-time performance on embedded hardware. This system would be modular, network transparent, and well-suited to the the types of context classification required for the Memory Glasses application.

### **6.2.1 Enchantment Middlewear**

Although there are examples of combined centralized and peer-to-peer communications systems<sup>2</sup> no such system existed in the domain of light-weight, structured, interprocess communications. To test the hypothesis that a combined whiteboard/peer-to-peer communications system would be most appropriate, I designed the Enchantment middlewear system. Enchantment is the first system to integrate structured whiteboard and peer-to-peer interprocess communications, and does so in an efficient, modular, portable, open-source implementation that is capable of real-time performance on modest embedded hardware.

### **6.2.2 Enchantment Whiteboard**

The Enchantment Whiteboard system is closely related to blackboard and whiteboard systems that have developed in artificial intelligence research over the last twenty-five years[37, 59, 56]. Unlike traditional inter-process communications systems (such as RMI Unix/BSD sockets, etc.) which are based on point-to-point communications, the Enchantment Whiteboard is based on a client/server model in which clients post and read structured information on a whiteboard server. This allows any client to exchange information with any other client without the attendant  $O(n^2)$  complexity in negotiating direct client-to-client communication; in fact, this is possible without knowing anything at all about the other clients. The Enchantment Whiteboard provides authentication and encryption using OpenSSL<sup>3</sup>.

With its small footprint, portable C/C++ implementation, and network-transparent architecture, The Enchantment Whiteboard has proven to be quite successful for MITHril interprocess communications applications. Dozens of client applications have been writ-

---

<sup>2</sup>...such as the Internet Relay Chat system, which utilizes a centralized client/server model for text chat communications and a peer-to-peer mechanism called DCC (Direct Client to Client) for file exchange. See <http://www.irchelp.org/irchelp/faq.html> for more information about the IRC system.

<sup>3</sup>OpenSSL is an open-source secure sockets[23] implementation. See <http://www.openssl.org/> for more information.

ten, ranging from specialized clients and servers to control interface behavior (such as the Anduin window manager’s use of the whiteboard for context awareness, or the “subliminal channel” DJ application that triggers the delivery of subliminal cues) to synchronous text-based instant messaging.

Perhaps more importantly, the network-transparent architecture that was originally proposed to address the problem of on-body distributed interprocess communication proves to be a nearly ideal match for multi-user applications, and mixed ubiquitous/wearable applications. This capability has proved particularly helpful for MITHril 2003, which typically utilizes a single on-body processor but is used primarily for group applications.

### **The Enchantment Signal System**

The Enchantment Signal system is an efficient point-to-point extension of the Whiteboard system. The Enchantment Signal system enables, light-weight, efficient, network-transparent, low-latency, high-bandwidth peer-to-peer interprocess communications. The Enchantment Signal system is an improvement over raw BSD-style sockets or RMI in that it transparently handles the negotiation and establishment of connections, provides authentication and encryption features, and greatly simplifies the task of writing robust one-to-one and one-to-many streaming communications code.

The Enchantment Signal system operates as seamless extension of the Enchantment Whiteboard. All Signal clients are also Whiteboard clients<sup>4</sup>. Signal generators (Signal clients that produce signal data streams) are represented by a Whiteboard signal handle abstraction, allowing Signal clients to discover each other. To facilitate communications across firewalls and NAT, Whiteboard servers transparently proxy signals from private to public networks — an important feature for bridging signal data from a private MITHril 2000 body network to the world at large. Because the Enchantment Signal system is intended for real-time signal processing and classification applications, a range of scalar, vector, and matrix data types are supported, and many useful Signal clients (such as signal mixers, loggers, feature calculators, *etc.*) are part of the open-source Enchantment Signal

---

<sup>4</sup>... although the reverse it not also true. Enchantment applications that are not involved in real-time signal processing or classification typically need only Whiteboard functionality.

system.

### **Supporting Evidence**

Our testing has shown that Enchantment Whiteboard servers running on modest embedded StrongARM hardware can handle hundreds of transactions a second without noticeably taxing system resources, and need only a few hundred KB of ram. On similar hardware, Signal clients can reliably handle multiple audio-bandwidth signals.

Our experience with the Enchantment Whiteboard and Signal system supports the hypothesis that the combined whiteboard and peer-to-peer interprocess communications capabilities provided by these systems are a good match for the problem of developing network-transparent, distributed, context-aware systems for individual and group applications.

Supporting evidence includes the use of the Enchantment middlewear use as the basis of Ivan Chardin's software for his Masters Thesis[7] and its use in other published research at the MIT Media Lab. It has also been used in soldier medicine physiology in collaboration with the US Army, and for a variety of other wearables research projects at the Media Lab. See Section 4.4 for specific examples.

### **6.2.3 MITHril Real-Time Context Engine**

I hypothesized that using the Enchantment middlewear I could develop a sensor data classification system that would be principled, flexible, and could achieve real-time performance on modest embedded hardware. This system would provide the capabilities necessary to support the Memory Glasses and other wearable context aware applications.

To test this hypothesis I designed the the MITHril Real-Time Context Engine. The architecture of the Real-Time Context Engine is based on the graphical model formalism of Jordan[31] and the classification process proposed by Duda *et al.*[18]. For model training the MITHril Real-Time Context Engine uses Kevin Murphy's open-source BNT[46] graphical models packages for Matlab.

Except for the Matlab-based model training code, the Real-Time Context Engine is implemented in C and C++ using the Enchantment middlewear. This provides real-time

performance for context classification on modest StrongARM embedded hardware.

### **Supporting Evidence**

Supporting evidence for the success of the Real-Time Context Engine includes the activity classifiers I developed for use with the Memory Glasses and physiological classifiers developed in collaboration with the US Army Research Institute on Environmental Medicine (USARIEM). The success of the MITHril Real-Time Context Engine is also supported by its selection for evaluation by USARIEM researchers for the Objective Force Warrior's Warfighter Physiological Monitoring System. See Section 4.4 for information about these example applications.

## **6.3 Wearable Human Computer Interaction**

My third major area of intellectual contribution is in the domain of human-computer interaction for wearable and mobile applications, and in particular in the area of low-attention, small-field-of-view visual interfaces for memory and decision support. I hypothesized that the problems I observed using conventional direct-manipulation interfaces for wearable memory support could be addressed on a theoretical level by extending existing models of human-computer interaction to take into account the divided attention and mediation effects of wearable technology. I further hypothesized that the extension of these models would produce design principles that could be used as the basis for Memory Glasses interaction.

To test the first hypothesis, I examined Donald Norman's Seven Stages of Action model and GOMS models. What I found was that there were straightforward extensions of both classes of models to account for wearable mediation and divided attention. The result is the "Seven Stages of Action in Wearables World" model, and the "GOMS models in Wearables World" analysis. Further, the implications of this extended model and analysis translate directly into the five principles of wearable human-computer interaction that I set out in Section 5.4, and can be summarized as follows:

1. Avoiding physically and perceptually encumbering the user.

2. Avoid unnecessary distractions in the interface.
3. Design interfaces that are quick to evaluate.
4. Simplify execution as much as possible, but no further.
5. Avoid pointers, hidden interface state, non-salient interface changes, and do not assume the user's undivided attention.

In order to test these principles of design as they relate to wearable memory support, I developed the prototype Memory Glasses memory support application developed for prosopagnosia and early-stage Parkinson's' patient applications, the Anduin window manager, and the more sophisticated Anduin/Enchantment Memory Glasses prototype.

Utilizing MITHril as the basis, the first memory glasses prototype embodies all five principles of wearable HCI. To address the first principle of wearable HCI, I developed the MITHril 2000 research platform. The second principle is addressed by having the display change only for reminder delivery. The third principle is addressed by having a simple, two-panel layout with consistently-located text and graphic elements. The fourth principle is addressed by eliminating explicit user interaction entirely, replacing it instead with context-triggered reminder delivery. No violations of the fifth principle are present in this simple design.

Limited though it is, this prototype demonstrates that a self-contained, light-weight, comfortable, proactive organizer application could deliver reminders based on time, location, and social context without requiring any explicit user interaction. This is an improvement over conventional organizers that are unaware of user social and location context and typically require the user to take further action (such as removing a PDA from a belt holster) to receive specific reminder details.

The Anduin window manager is a much more complex system, as it is an entire application management framework for wearable HCI. Beginning with MITHril's human factors to address the first wearable HCI principle, it addresses the remaining principles in the following way:

1. The first principle, that of avoiding physical and perceptual encumbrance, is addressed by the human factors of MIThril hardware platform.
2. The second principle, which may be summarized as distraction avoidance, is addressed as follows: Unlike other window managers, Anduin manages the interaction demands of applications by restricting interaction to a single application at a time in central focus. In other window managers, it is necessary for the user to not only select, but position application windows. Anduin completely controls window positioning, thus reducing what would otherwise be a complex pointer-based continuous-positioning and selection task to a discrete keypress-based task. Further, Anduin provides the ticker bar as a unified “out of band” notification channel usable by any application, even those not in central focus, thus reducing the necessity for user-driven or context-driven application focus switching.
3. The third principle relates to speeding interface evaluation. By completely controlling the size and layout of all windows on screen, Anduin imposes a consistent, easily-evaluated structure on all applications and applets, thus reducing the need for visual search. Anduin’s state is completely driven by discrete keypresses, and each salient keypress produces a corresponding discrete change in window manager state. There is no possibility of overlapping or hidden application windows. None of these features are found in any other X-Windows window manager.
4. The fourth principle relates to simplifying interaction execution. Anduin is the only window manager designed to be operated solely with key presses. Further, Anduin provides a limited number of operations, which are bound to single user-assigned function keys. Due to screen real-estate limitations, these mappings are not visible, which imposes on the users’ memory and violates the fourth principle. However, the user need learn only five function key mappings to access all important window manager functions, and these five mappings can easily be displayed by a help-screen application started at run-time by the window manager. Further, other window managers provide no mechanism for context-driven focus change. Anduin explicitly provides this mechanism by tight integration with the Enchantment Whiteboard and

MIThril Real-Time Context Engine.

5. Anduin avoids pointers (and the visual search and continuous positioning tasks that are associated with them), hidden interface state, non-salient interface changes, and does not assume the user's full attention.

The Anduin/Enchantment Memory Glasses prototype extends the functionality of the first Memory Glasses prototype into a multi-application framework. Like the first prototype, the design of the Anduin/Enchantment Memory Glasses prototype addresses all five principles of low-attention wearable HCI, but includes action-context awareness more options for reminder delivery, including the delivery of subliminal cues.

### **Supporting Evidence**

Further support for the effectiveness of the low-attention wearable HCI strategy that I developed can be found in its influence on Steve Schwartz's WearSAT[6] project. WearSAT, which was developed after MIThril 2000, stands for Wearable Situation Awareness Terminal, and is a wearable computing prototype designed to be worn inside a flight-qualified EVA suit. The WearSAT system is one of the only EVA-qualified in-suit systems to be developed in the last 20 years, and the prototype interface design is strongly influenced by the Anduin window manager. Further, the interaction model proposed for the WearSAT is strongly reminiscent of the Memory Glasses. Both applications are based on the just-in-time delivery of supporting information. Instead of generalized memory support, WearSAT is designed for specialized EVA task support, including memory, navigation, and communications support. Instead of context-triggered reminder delivery, information delivery in WearSAT is triggered by ground controllers in two-way voice communication with the EVA astronaut. And like MIThril 2000, WearSAT components are distributed in a thin layer across the body and intended to be worn beneath "clothing" — in this case a Russian or American EVA space suit. As proposed, the WearSAT design embodies all five principles of low-attention interaction set forth in this thesis.

### 6.3.1 Cuing Research

A particularly interesting and (I believe) potentially important contribution I have made in the domain of wearable HCI is cuing research investigating different types of cues for just-in-time memory support, and particular my preliminary work investigating the use of subliminal visual and superliminal masked auditory cuing for just-in-time memory support.

I hypothesized that I could use the semantic-priming effects of subliminal perception, well-known to cognitive scientists, as a means of providing just-in-time memory support. In order to formally test this hypothesis, I ran a series of experiments comparing the effects of different types of just-in-time cue delivery on memory recall. The cognitive basis for this work is outlined in Appendix A and the experiments are described in detail in Appendix B.

I found that short-duration masked visual cues did have a strong, significant positive effect on memory recall, improving my subjects' performance by an average of 58% over the control condition, significant at  $p = 0.003$ . These results suggests that it is possible to provide effective memory support using just-in-time subliminal cues without the user's conscious awareness or attention. If true, this interaction technique could dramatically reduce the distraction associated with memory support.

Further, my cuing research suggests that misleading subliminal cues do not decrease recall accuracy or mislead the user. Indeed, if anything my results suggest that in at least some cases incorrect (but related) masked short-duration visual cues may have a slight positive effect on recall. Thus, there appears to be very little cost to incorrect subliminal cue delivery — a significant consideration for real-world memory support applications that must operate with less than 100% certainty.

These results are preliminary, but if replicated by others could open the door to an entire new field of subliminal HCI for low-distraction applications.



# Chapter 7

## Conclusions and Future Work

### 7.1 MIThril Hardware Conclusions

The MIThril hardware development efforts have provided a useful platform for the Memory Glasses research, substantially addressing previously unaddressed human factors limitations in previous wearable research platforms and providing the necessary flexibility for a variety of interesting wearable computing research.

The many applications of MIThril hardware to a variety of research projects, both inside the Media Lab and outside MIT, speak to the relevance of this work. MIThril continues to be developed in its PDA-centric MIThril 2003 incarnation, and is supporting a range of research projects ranging from soldier medicine to group coordination and social networks mapping.

The MIThril 2000 platform succeeded in most of the goals we set for it, but proved too complex and expensive in the full, multicomputing vest format, for widespread use. This has largely been remedied in the simpler, PDA-based MIThril 2003 architecture.

### 7.2 MIThril Software Conclusions

The Enchantment middleware has proven its worth in many ways, providing a basis for a range of interesting research projects and is under active development today. The Enchantment middleware and MIThril Real-Time Context Engine have evolved along with

the MITHril hardware platform, and are now used in multi-user, group collaboration and social networks research applications.

Given the strong interest in the use of this software within the Media Lab and at other institutions (including USARIEM), the future of the Enchantment middleware and MITHril Real-Time context engine is bright. Its open-source license and efficient, portable implementation make it a good candidate for tackling a range of problems in real-time sensor data communications, processing, and analysis.

Recently, the Enchantment middleware and MITHril Real-Time Context Engine were ported to the Windows XP environment, and commercial support of this open-source software by AWARE Technologies, Inc. is on the horizon.

### **7.3 Human Computer Interaction Conclusions**

My work in wearable HCI has several dimensions, each of which have been important in this thesis research. The first dimension is my six-year daily use personal experience with a range of prototype wearable computing interface software and hardware, much of which I reengineered, designed, or otherwise shaped to my own needs. This experience provided me with invaluable insight into the fundamental human factors and cognitive challenges of wearable computing.

This experience, in turn, led me to reevaluate conventional HCI models in light of the divided attention and user/world mediation effects of wearable computing, resulting in the modified Seven Stages of Action model and the GOMS model analysis put forward in Chapter 5. Based on this analysis, I proposed five principles of low-attention wearable HCI, and embodied these principles in the design and engineering of the Memory Glasses prototypes and the Anduin window manager.

My research into low-attention interaction strategies provided me the background, justification, and opportunity to investigate the use of just-in-time subliminal for memory support, as well as other methods of cue delivery. The cuing research described here is only a beginning; If my results can be replicated, an entire field of wearable subliminal user interaction is waiting to be explored. I hope that my preliminary experimental work

in this area and the positive results of the subliminal cuing work will encourage others to explore this area and attempt to replicate my results.

## 7.4 Future Work

It is hard to write about directions for future work at the end of a doctoral research process. I am acutely aware of just how much more could be done in all of the areas that I have investigated, and how fundamentally incomplete this research is. As Neil Gershenfeld once said to me, “Good thesis research projects aren’t completed, they are abandoned.” The time has come for me to let go of many of the projects I’ve worked on over the last six years — far too many to name here — which I’ve told myself that I will return to and complete given a spare week or two.

At the same time, the next steps in this research agenda could not be more clear. I believe that research in low-attention interaction for secondary task applications will only become more important as the prevalence of these applications (cell phones, PDAs, mobile IM/email devices, *etc.*) grow. The potential for this technology to positively benefit humanity huge, and at the same time we are all participating in a large-scale, and largely unconsidered, experiment in which the technology is changing our lives, and not always for the better. As the creators of new technology, we can never foresee all ends, but we can take responsibility for investigating the reasonably foreseeable consequences of our actions.

I am less worried by the prospect that somehow the next generation of wearable/mobile devices will damage our society in some irreparable way than that an unconsidered approach will delay or even prevent us from realizing the full potential of this technology. Present high-attention interaction strategies sharply limit the number and range of wearable/mobile applications that are manageable by a single user. Until we take a different approach to the design of this technology, and realize that the ultimate limits are cognitive and not MIPS, Mbits, or megapixels, we are unlikely to move much beyond the present world of PDAs, text messengers, and cell phones.

I would very much like to continue my research on cuing, and at the same time I am also very interested in commercializing the Memory Glasses and supporting technologies.

In the future I hope to pursue a dual academic/commercial strategy to achieve these ends.

# Appendix A

## Attention and Wearable Interaction

### A.1 Introduction

This is a thesis about wearable computing for memory support, including experimental investigation of different cuing methods for just-in-time memory support. In this appendix I provide a brief overview of the HCI, cognitive, and perceptual science literature as it relates to wearable human-computer interaction for memory support. This review focuses mostly on issues surrounding the delivery of visual cues, both subliminal and overt, but issues related audio cuing are also discussed.

Because there is no established theory of subliminal HCI, it is necessary for me to cite primary cognitive and perceptual science sources to justify some of my research and design choices. Indeed, low-attention interaction in general appears to be a subject little studied outside of the avionics and driver distraction literature.

The literature search presented in this appendix is not intended to be comprehensive with regards to any of the subjects that I am surveying. For example, entire books are devoted to the survey of issues of attention, such as Pashler's excellent reference[53]. The purpose of this appendix is to provide reasonable justification for my design choices and point the interested reader in the right direction to learn more about these issues.

## A.2 Over reliance and Memory Support

Memory support is a common application of wearable task-support systems. Much effort has gone into designing wearables (such as CMU's VuMan series) that are memory surrogates, providing checklist-style memory support[71]. Jennifer Ockerman refers to these systems as *task guidance systems* and describes over-reliance problems that can result from their use.

For example, Ockerman describes a study in which experienced pilots performed aircraft inspections, some of whom were asked to use a hands-free wearable inspection guidance application. The pilots who used the wearable did about as well as the pilots who did not for tasks on the wearable's inspection list, but did much worse in performing tasks not on the list[50].

Although the Memory Glasses is not targeted at checklist-style task guidance *per se*, over reliance is a concern, particularly with respect to the possibility of misleading the user.

### A.2.1 Probabilistic Context Classification and Misleading Users

Although a time-aware system can know the time of day with high confidence (assuming the clock has been set properly), social context classification and activity context classification are generally more difficult, and likely to result in lower confidence classification. To operate effectively in the real world, a context-aware reminder system must deal gracefully with low-confidence context classification and classification error.

One response to this problem is deliver only reminders triggered with very high confidence. Unfortunately, high confidence does not mean certainty, and some degree of error is all but inevitable. As responsible designers we must consider the potential problems of unreliable context classification and the delivery of incorrect reminders. If at all possible, we should explore the effects of incorrect cuing and attempt to design for graceful degradation as accuracy declines. The likelihood of context classification error is the justification for including a misleading cue condition in the experiments described in Appendix B.

### A.3 Attention and Task Interference for Wearable HCI

Distraction is an important concern in designing wearable interfaces. Early on in my work with wearable interaction I became convinced that managing the wearable's demands on the user's attention was critical for designing successful task-support applications, including memory support. This led me to research the cognitive science of attention and perception.

Contemporary theories of attention began with Broadbent's "Early Selection Theory" in 1958[3]. These theories are based on finite resources of perception and cognition. If the user is engaged in two tasks that require conscious attention, the resulting resource competition will cause task interference, which in turn results in reduced performance on both tasks. The extent to which these tasks interfere with each other is the extent to which they involve competing cognitive and perceptual resources (as well as the user's familiarity with the tasks). Task interference results in slowed reaction time and decreased performance on both tasks[26].

The problem and consequences of task interference are particularly problematic for wearable task-support applications: As the complexity of the real-world task increases, the need for task support increase. However, as the complexity of the real-world task increases the capacity of the user to attend to attend to the task support application *decreases*. This makes effective support hardest when the user needs it most.

Task interference can be mitigated by reducing the perceptual or cognitive demands of the wearable task support application. One important technique for reducing interaction complexity is *context awareness*, or the use of non-explicit user and environmental input, typically acquired through sensors[52, 78, 75]. Context awareness can reduce, or in some cases eliminate, the need for explicit user input for some applications.

However, at best context awareness only addresses half of the interaction problem; Even when the use of contextual information drastically reduces the need for explicit user input, interactive applications must still present information to the user, with the attendant perceptual and cognitive demands.

Likewise, the perceptual and cognitive demands of presenting information to the user can be lowered by reducing the complexity of information displays and reducing the amount

of interpretation required by the user [89, 88]. However, any interaction strategy that requires conscious perception will necessarily require controlled (conscious) processing and interfere with other controlled processing tasks.

This is unfortunate, because it implies that task interference is an inevitable consequence of task support, unless it is possible to bypass the user's conscious perception and still provide meaningful task support. This possibility is explored in the next section.

## **A.4 Subliminal Perception and Wearable HCI**

In this section I provide a survey of the literature of unattended perception and unconscious cognition as it relates to the use of just-in-time subliminal cuing for memory support. This, in turn, provides the basis for the experimental work described in Appendix B.

The reason I cite the cognitive and perceptual science literature and not an established literature of subliminal HCI is that there isn't one<sup>1</sup>. Prior to my work in this area there has only been one other publication of note, and it dealt with desktop HCI and was published in 1991 (F. Layne Wallace, *et al* "The Effect of Subliminal Help Presentations on Learning a Text Editor,[83]).

### **A.4.1 Unattended Stimuli**

Broadbent's "Early Selection Theory[3]" proposed that while all stimuli are analyzed at a physical (nonsemantic) level, only attended stimuli are processed further. "Late-selection" models, such as the one proposed by Deutsch and Deutsch[11], were developed in response to mounting evidence of high-level semantic interpretation of unattended stimuli, such as the ability to accurately shadow a message that jumped back and forth between ears in a dichotic listening task based on semantic content[22].

Modern theories of attention have moved beyond the early- vs. late-selection debate. The evidence now generally supports a compromise "attenuation" theory[82], in which

---

<sup>1</sup>This is not to say that there has been no research on the effects of semantic priming or unattended perception for a range of other purposes, including applied research on the use of these cues for advertising. However, there is no established literature exploring the applications of semantic priming (or other unattended perception effects) to human-machine interaction in the conventional interactive sense.

filtering attenuates, but does not prevent, the processing of unattended stimuli. Current research focuses on the fine-grained attentional aspects of different perceptual modes and tasks — see [53] for a discussion of contemporary attention research.

### **A.4.2 Unconscious Cognition**

“Unconscious cognition” is essentially the processing of information by the brain in ways that do not require or provoke the awareness that the processing is occurring.

In “A New Look: 3,” Greenwald observes that unconscious cognition is solidly established in empirical research, though in a form substantially different from (and simpler than) that put forward by classic psychoanalytic theory[24].

High-level semantic processing of unattended stimuli provides strong evidence for unconscious cognition. Further evidence lies in the phenomenon of automatic processing[70]. Highly practiced skills, such as word reading in literate adults, appear to be automatic and parallel, in that little demand is placed on attention or short-term memory. (Other examples include many aspects of driving, or playing a musical instrument.) Controlled processing is just the opposite — it places demands on attention and short term memory, and is highly serial[26].

Once a processing task becomes automatic, it may also be inevitable, in that it will always occur when an appropriate stimulus is presented. This is illustrated by the Stroop effect[79], *e.g.* when subjects are shown a color word (for example the word “green”) printed in ink of a different color (for example, blue ink), they find it very difficult to say the name for the ink color aloud (“Blue.”) rather than reading the word (“Green.”). Processing the word as written language is inevitable, even as it interferes with the consciously-selected controlled processing task of naming the color of ink.

### **A.4.3 Subliminal Perception**

Unattended perception is a type of automatic process

...in which behavior or physiological response are shown to be influenced by stimuli which fall below the subject’s threshold of awareness. This is achieved

by presenting stimuli of very low intensity or or very short duration, or at frequencies beyond the range of conscious perception, by visual or auditory masking, or by the use of unattended channels as in dichotic listening.[26]

Stimuli can be divided into three categories, based on their availability to attention:

1. the consciously-perceived,
2. the perceptible-but-unattended, and
3. the subliminal.

For the purposes of this thesis, I draw a distinction between the more general case of an unattended stimulus (such as that resulting from a dichotic listening task) and a subliminal stimulus, which *cannot* be attended to. Into this category I also place stimuli that under other circumstances might be perceptible, but are masked in such a way that the subject is unable to perceive them.

An important aspect of subliminal stimuli is that they *cannot* be attended to; they are not available for conscious perception. If a person can't consciously perceive something, they cannot choose to direct their attention towards it.

But the fact that such stimuli cannot be attended to does not mean that they do not get processed by the brain in ways that affect subject performance. Deutsch and Deutsch[11] and Treisman[82] observed high-level semantic processing of unattended stimuli. Von Wright et. all[90] and Govier and Pitts[21] found that subjects were capable of discriminating between alternative meanings of unattended stimuli in a dichotic listening task. Further, Dixon[16] and Groeger[25] have shown that subliminal stimuli may be subject to high-level semantic, memory, and emotional processing, as in the case of “perceptual defense” — the selective inhibition of conscious perception of disturbing or embarrassing material. Marcel[40, 41] used masked visual stimuli to show that subliminal stimuli have semantic-level priming effects.

#### **A.4.4 Subliminal Cuing for Wearable and Mobile Applications**

My operational definition of a *cue* is a stimulus that is intended to affect the user's performance on a memory task. An *overt cue* is a cue that can be consciously perceived and reported by the user. A *subliminal cue* is a cue that cannot be consciously perceived or reported. Both overt and subliminal cues may produce measurable effects on task performance. In another axis, cues may also be *correct* relative to the task at hand, or they may be *miscues* that are contextually inappropriate.

In 1991, Wallace showed that subliminal cuing decreased reliance on online help in a desktop-computer text editing task application[83]. In essence, this was a use of subliminal cues in support of a memory task. Subliminal cues were shown to affect the subjects' conscious use of overt cues.

Since Wallace, little work has been done since to explore the use of subliminal cuing in the domain of human computer interaction. This lack of research may be because the common use of modern desktop computers, with their high-resolution displays and their users seated on chairs and concentrating upon the screen, provide a very large quantity of available interaction bandwidth. In such a context, there is little need to reduce the attentive demands of the interaction.

However, the situation is very different in wearable and mobile computing environments, in which display resources are limited and the user is often engaged in complex tasks imposing significant attentional and cognitive demands, such as piloting a vehicle, conversing with others, *etc.*

A more effective task-support system would support the user's memory of the procedure without interfering with the task, potentially reducing or eliminating over-reliance problems. I began investigating this possibility with my subliminal visual cuing research[13], which is described in detail in Appendix B.

#### **A.4.5 Subliminal Cuing and Attention**

The automatic nature of subliminal perception means that the processing of subliminal cues should not distract the user's attention, nor interrupt any consciously-controlled cognitive

processing tasks. Therefore, subliminal cues should have a minimal effect on the user's overall cognitive load.

The rapid, parallel nature of automatic processing and lack of conscious distraction associated with subliminal cuing leads me to believe that the use of a "subliminal channel" for memory support could be of great value in circumstances where overt cues would be distracting, inappropriate, or otherwise undesirable.

#### **A.4.6 Miscues and Subliminal Perception**

The existence of Stroop effects demonstrates that automatic cognitive processes can interfere with consciously-controlled processes. (The classical example involves presenting a subject with a word like "green" written in blue ink, and asking them to report the ink color[79].) Can inappropriate cues, be they subliminal or consciously perceived, interfere with memory retrieval or other cognitive processing? This is one of the questions my research addresses through the use of a misleading cue condition.

#### **A.4.7 Masked Superliminal Audio Cues**

Much of the evidence for high-level semantic interpretation of unattended stimuli comes from dichotic listening experiments. For example, Gray and Wedderburn's work in 1960 presented evidence that subjects were able to accurately shadow a message that jumped back and forth between ears in a dichotic listening task based on semantic content[22]. More recently, Von Wright et. al[90] and Govier and Pitts[21] found that subjects were capable of discriminating between alternative meanings of unattended stimuli that the subjects were not consciously aware of.

This suggests that an unattended audio stimulus could be used to provide memory support. However, to be useful as a low-attention interaction strategy for memory support, it is necessary that the cues be not simply unattended, but actually subliminal. Which is to say, below the threshold of comprehension of the user. This raises the question of how this might be plausibly achieved, given the engineering constraints of a realistic wearable interface.

Some research has been done with white-noise masked superliminal stimuli, such as that done by Merikle investigating the effects of subliminal audio messages[44]. White-noise masked stimuli are interesting from the standpoint of this application, because they are easy to generate and do not require artificially directing the subject's attention as is necessary in a dichotic listening task.

Unfortunately, This work has largely shown negative results. However, Merikle points out that it is entirely possible that the reasons these results were negative is that a minimal stimulus condition was not met: there was simply not enough information perceived to influence higher-level cognitive processes[44]. Subsequently, other researchers have rephrased this in terms of signal-to-noise (S/N) ratios, suggesting that higher S/N ratios may be necessary for semantic priming[27].

As a result, I have chosen to investigate the effect of white-noise masked superliminal audio cues that have a high enough S/N ratio to be detected, but are just below the threshold of comprehension. Further, to ensure that this condition is met, I designed an experimental procedure where the amplitude of the cues is calibrated individually for each subject, as described in Section B.5.



# Appendix B

## Memory Support Experiments

### B.1 Introduction

In Section 5.9 I summarize a series of experiments I conducted to explore the effects of different types of cues for just-in-time memory support. All three experiments shared a common procedure, differing only in the way cues were delivered.

In this appendix I describe these experiments in detail, beginning with the common procedure and then presenting the details of each of the three experiments. In this section I present a detailed analysis of the results of the experiments, individually and taken together as a single larger experiment.

### B.2 Experimental Procedure

In this section I describe the experimental procedure used in all three cuing experiments. The experimental procedure was designed to be simple enough to support meaningful analysis, but close enough to a “real world” task that the applicability of the results to real-world memory support would be clear.

### **B.2.1 “Social Memory Assistant” Scenario**

Remembering names and faces is a common task in social settings, and one many people find difficult. The experimental scenario I chose is an application of the Memory Glasses to a social setting, with the goal of supporting the user’s memory of face/name associations.

An important feature of this scenario is that I assume the social context classification isn’t completely reliable, and that the Memory Glasses might confuse one conversational partner for another; a reasonable assumption given the limitations of current social-context classification technology, such as face recognition<sup>1</sup>.

Rather than introducing my subjects to a relatively large group of unknown people (which would need to be the same for all subjects) in a social situation, in this “Social Memory Assistant” scenario I chose to use a desktop computer as a stand-in for actual social interaction.

By using a desktop computer for presenting memorization and testing stimuli, I was much better able to control for non-experimental conditions than would have been possible given a more “real world” setting for this task. However, an obvious limitation of this design is that its highly structured nature and limited information is not necessarily a good match for the realities of attempting to learn names at a cocktail party.

### **B.2.2 Design Overview**

In collaboration with Vicka Corey of Harvard Medical, I designed an balanced experimental procedure that would allow for within-subjects analysis. Each experiment employed three experimental conditions: uncued, correct cues, and misleading cues. For each of the three conditions, the subject was presented with seven memory trials, resulting in a total of 21 memory trials per subject.

The inclusion of a misleading cue condition was important for three reasons. First, I assumed that any real memory support application would have to operate with imperfect information, and errors are inevitable. Second, I expected subject performance on the

---

<sup>1</sup>In fact the Memory Glasses prototype I constructed uses an IR-active-tag based classifier that provides relatively high-confidence classifications, but I do not expect that a tag-based solution to social context classification will always be possible.

miscued trials to provide evidence on the extent to which subjects relied on the cues rather than their own memory. And third, in the case of the subliminal cuing experiments, I expected subject performance on miscued trials to provided corroborating evidence for conscious (non-subliminal) perception of cues, as was the case in the subliminal visual cuing pilot study[12].

I knew that some names and faces would be more memorable than others. To balance the design, I ensured that each face/name pair would appear in each condition for equal numbers of my experimental subjects. The way in which the balancing was accomplished is described below in Section B.2.2.

Because I used a within-subjects design, I was able to examine the effect of the experimental conditions on a particular subject's performance. For example, I can say that a particular subject did two questions better on correctly cued memory trials than uncued memory trials. More importantly, I can analyze this within-subject effect across subjects, allowing me to control for individual differences in memorization skills.

### **B.2.3 Procedure**

#### **Orientation**

Each subject was assigned to a subject condition (see below). Subjects were informed of the nature of the memorization task and the memory test. Subjects were given an experiment-specific orientation, described below. Subjects were oriented on the testing system, and asked to type their own name to familiarize themselves with the interface. Subjects were instructed to do as well as possible on the memory test, and to guess if they were not sure.

#### **Memorization**

Each subject was given two minutes to memorize twenty-one name-and-face pairs. The name/face pairs were presented as a web-page in a seven-by-three grid. The duration was timed automatically.

The names were chosen at random from a list of historically popular American women's names, and pictures of women were chosen at random from the FERET[55] face database.

(Women were chosen for stimuli because there is more variation in common American women's names than in men's.)

## **Memory Trials**

After the memorization period there was a brief pause, during which the subject indicated that they were ready to proceed. Then the memory trials began. In each trial, the subject was shown one face from the memorization set, and attempted to produce (type) the associated name. Each memory trial was ten seconds in length, and the subjects could answer anytime during that period<sup>2</sup>

Depending on the experimental condition of the trial, the subject would receive either a correct cue, a misleading cue, or no cue. After all trials were completed, the subject was given a questionnaire to complete and the protocol was finished.

### **B.2.4 Memory Trials**

The trials were divided into three conditions of seven trials each per subject:

Condition "n": uncued condition (control),

Condition "C": correct cue condition,

Condition "X": misleading cue cuing.

In "n" trials, no cuing occurred<sup>3</sup>. This was the control condition. In "C" condition trials, correct cues were provided using the experiment-specific cuing procedure. In "X" condition trials, misleading cues were provided using the same technique as the "C" condition trials.

To balance the design, each of the 21 faces was randomly assigned to one of three groups, labeled "I", "II", and "III", resulting in three groups of seven faces. I generated three randomized sequences of trials, labeled Sequence 1, Sequence 2, and Sequence 3. For each sequence, the face groups were assigned to each of the three experimental conditions

---

<sup>2</sup>At my defense, the question arose as to whether the length of name had any effect. It was not a factor I considered in assigning names. However, the balanced design ensures that if name length were a factor, it would be spread across all three conditions equally.

<sup>3</sup>For the subliminal cuing experiments, the mask stimulus was present even when no cue was delivered.

in rotation, i.e.: Sequence 1 n = I, C = II, X = III.

Sequence 2 n = II, C = III, X = I.

Sequence 3 n = III, C = I, X = II.

The result was three balanced sequences of 21 trials each. For each trial sequence, I generated a randomized memorization stimuli grid containing all 21 faces and names in random order. The combination of a sequence and its associated memorization grid defined a subject condition. The result was three subject conditions consisting of a memorization grid and a trials sequence.

Subjects were assigned a subject condition on a rotating basis, such that one third of the subjects experienced each subject condition. In this way, the experiment was balanced so that each face/name combination appeared in each experimental condition for a third of the subjects. Further, the different ordering of the trials sequences and memorization grid by subject condition provided some control for primacy and recency effects.

*Recency* is the tendency to remember better what one has most recently learned, such as the last items in a sequence. *Primacy* is the tendency to remember better the first first items in a sequence. The middle items of a sequence tend to be remembered less well than either the beginning or end[26]. By randomizing the ordering of faces in the memorization grids for each subject condition, I was able to control somewhat for primacy effects, since any uniform tendency to memorize in order (top to bottom, for instance, or left to right) would be spread across conditions. Likewise, by randomizing the order of conditions for each trials sequence I was somewhat able to control for recency effects — I expected my subjects to do somewhat better on the first memory trials of any sequence than the last (due to recency), so by mixing up the order I spread this effect across conditions.

### **B.3 Subliminal Visual Cuing Experiment**

The cues for the first experiment I ran were short-duration masked video frames presented at the transition between static images.

### **B.3.1 Hypotheses**

I hypothesized that my subjects would do better on the correctly cued memory trials than the uncued trials, *i.e.* I expected the effect of the correct cues would be positive. I further hypothesized that that subjects would do no better, and perhaps slightly worse, on the miscued trials, *i.e.* I expected the effect of the miscues to be zero, or slightly negative. Further, I hypothesized that my subjects would be no more likely to choose the cued answer on miscued trials than any other name used in the experiment.

### **B.3.2 Orientation**

Because I wanted to simulate the conditions of intentionally using a subliminal task support application, all subjects were told in advance that the experiment involved subliminal visual cuing. They were not told anything about the nature or content of the cues.

### **B.3.3 Experiment-specific Procedure**

After orientation and signing the consent form, subjects donned the monocular MicroOptical QVGA clip-on head mounted display (see below) and were instructed on its proper position and orientation. The display was oriented so that it would appear centered in the subject's field of view when they looked at the desktop computer display.

### **B.3.4 Equipment**

Subliminal cuing was provided by a head-mounted display worn on the subject's own eyeglasses, or blank optical-quality glasses for subjects who did not wear glasses. The head mounted display was worn during the memorization period as well as during the memory trials, but only active during the trials. The subliminal output device was was a MicroOptical QVGA clip-on head-mounted display driven by custom display hardware on the MIThril 2000 wearable computing system.

I designed this hardware, shown in Figure B-1, to provide complete control of all display parameters, including the timing and sequencing of backlights. This allowed me to

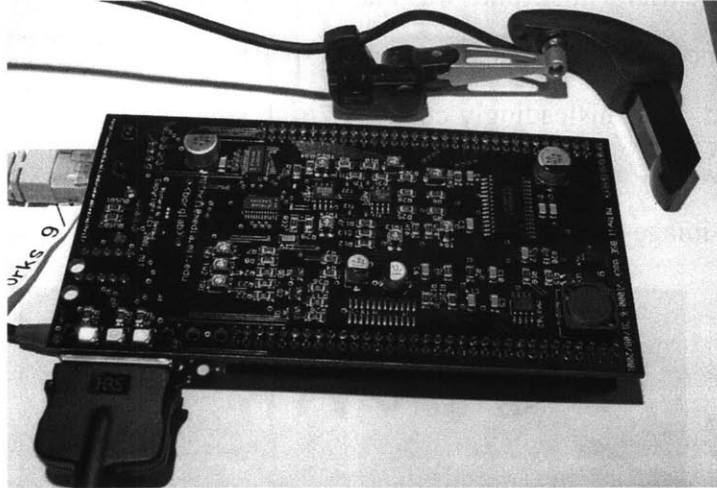


Figure B-1: MITHril 2000 BSE Dock video driver hardware and MicroOptical QVGA clip-on display

precisely control the duration of the subliminal cuing frames (described below) and to adjust this parameter independently of the duration and sequencing of other display information.

The detailed control provided by this video hardware and driver software allowed me to precisely calibrate the duration and intensity of subliminal cuing frames. I hypothesize that by adjusting these for individual subjects, I may be able to produce even stronger effects in future experiments. In order to allow replication of these experiments by others, the design for the driver hardware and source code for the software is available on the MITHril web site [64].

### **B.3.5 Cuing Strategy**

Subliminal cues, in the form of short-duration masked video frames, were displayed on the MicroOptical Head Mounted Display during the memory trials. The type of cue depended on the condition of the trial, which was unknown to subject or investigator during the running of the experiment.

During each trial the HMD cycled through ten static masking images, one every second. At the transition between images, a 1/180th second cue frame was inserted, resulting in 10 cues per memory trial.

For uncued “n” trials, this frame was blank (black). For correctly cued “C” trials, the cue frame contained the correct answer (name) spelled out in a large sans-serif font on a black background. For misleadingly cued “X” trials, the cue frame contained a plausible but incorrect answer — a name associated with a different face in the experiment. Example cue and masking images can be seen in Figure B-2

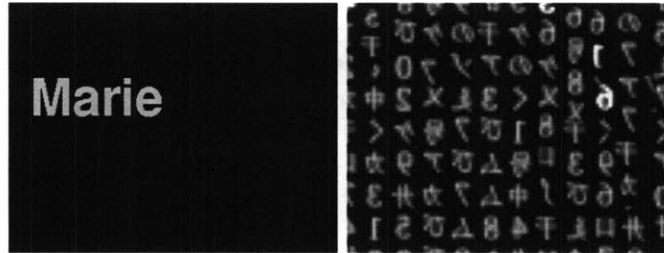


Figure B-2: Example Cue and Mask Image

### B.3.6 Results

A total of 28 subjects were run. The data from seven subjects were discarded for protocol violations, leaving 21 subjects in the analysis.

#### Within Subjects Analysis

The within-subjects analysis of the 21 subjects’ data is summarized in Table B.1 and Figure B-3. The *effect* was the difference in performance – that is, the number of correct names produced – between control and experimental conditions for each subject.

The mean effect of correct cuing (“C” trials) was found to be 1.00, or an improvement factor of 1.58 over the uncued “n” trials. The mean effect of misleading cuing (“X” trials) was found to be 0.48, or an improvement factor of 1.28 over the uncued trials.

I used a one-tailed T test to test the hypothesis that the effect of correct cuing (“C” condition) was positive. I found this to be acceptable at  $p = 0.003$ . I used a two-tailed test to test the hypothesis that the effect of misleading cuing (“X” trials) was non-zero; this was borderline significant at  $p = 0.09^4$ . The within-subjects hypothesis testing is

---

<sup>4</sup>Although it might have been possible to push the significance of the “X” condition effect by running more subjects, the proper approach would be to run another experiment to explicitly test the hypothesis that

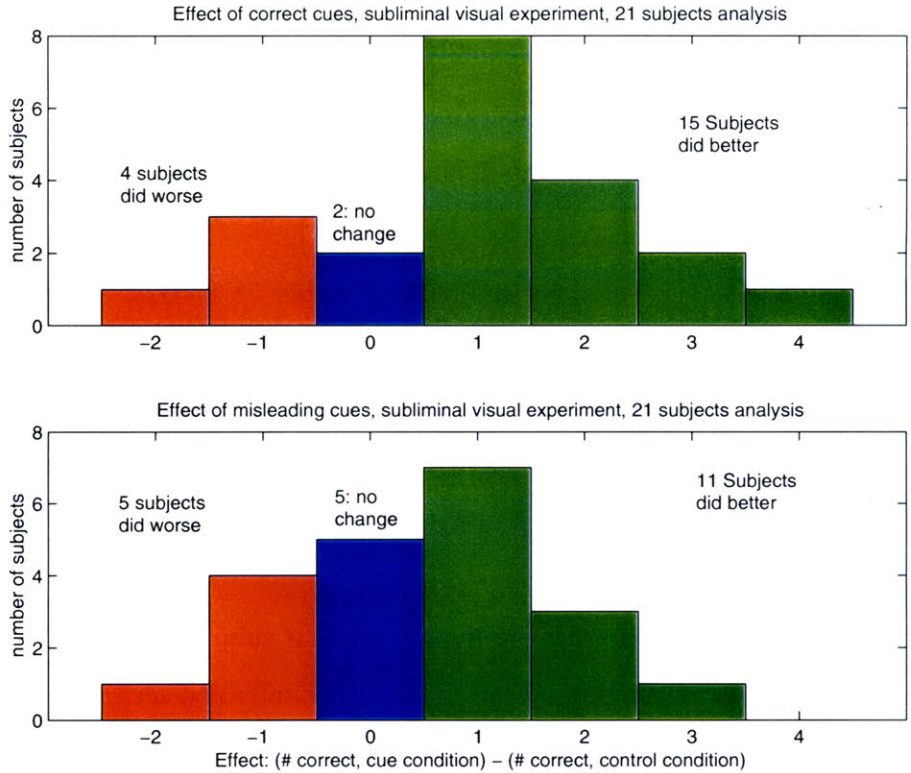


Figure B-3: Histograms comparing the effect of correct (“C”) and incorrect (“X”) masked short-duration visual cues.

cond.	mean	effect	effect $\sigma^2$	Student T $p$
“n”	1.71			
“C”	2.71	1.00	2.20	(one-tail) 0.003
“X”	2.19	0.48	1.56	(two-tail) 0.09

Table B.1: Within-subjects hypothesis testing for masked short-duration visual cues.

summarized in Table B.1

### Between Subjects Analysis

I designed this experiment for within-subjects analysis to better control for individual variations in memory skill, thus providing greater experimental power. However, the within-subjects analysis is conducted on the effect of a condition (a subject’s cued condition score minus their uncued condition score), not a direct comparison between subject condition

misleading subliminal visual cues might improve performance. This would be a one-tailed test vs. the present two-tailed test, and if similar results were observed to that of the present experiment the results would be significant at  $p = 0.045$ .

scores.

Calculating the effect creates a potential problem, since a low average score on the uncued condition would tend to bias effects in the positive direction. The mean on the “n” control condition for this experiment was 1.71 out of a possible 7 questions correct, so this bias cannot be ruled out.

To address this possible bias, one approach would be to analyze the effects using a Chi Square test or other non-parametric hypothesis test. However, a more conservative approach is to throw out the within-subject effects entirely and analyze the raw condition scores as a between-subjects design. This results in a loss of experimental power, and as one would expect, lower significance numbers are the result. However, it does eliminate any possible within-subjects bias.

I used a one-tailed T test to test the hypothesis that subjects did better on the “C” condition trials than the “n” condition trials. This was found to be acceptable at  $p = 0.01$ . I used a two-tailed test to test the hypothesis that subjects performed differently on the “X” condition trials than the “n” condition trials. This hypothesis was found not to be acceptable at  $p = 0.22$ . The results of this between-subjects hypothesis testing is summarized in Table B.2.

Since the results of the more conservative between-subjects hypothesis testing are consistent with the within-subjects hypothesis testing, I accept the hypothesis that correct masked short-duration visual cues improved my subjects’ performance on the memory trials.

cond.	mean	$\sigma^2$	Student T $p$
“n”	1.71	1.11	
“C”	2.71	2.91	(one-tail) 0.01
“X”	2.19	1.96	(two-tail) 0.22

Table B.2: Between-subjects hypothesis testing for masked short-duration visual cues.

### **B.3.7 Miscues and Spreading Activation**

As shown in Table B.1, the within-subjects analysis of the miscued condition showed a borderline-significant *increase* in performance. While I was not expecting significant task

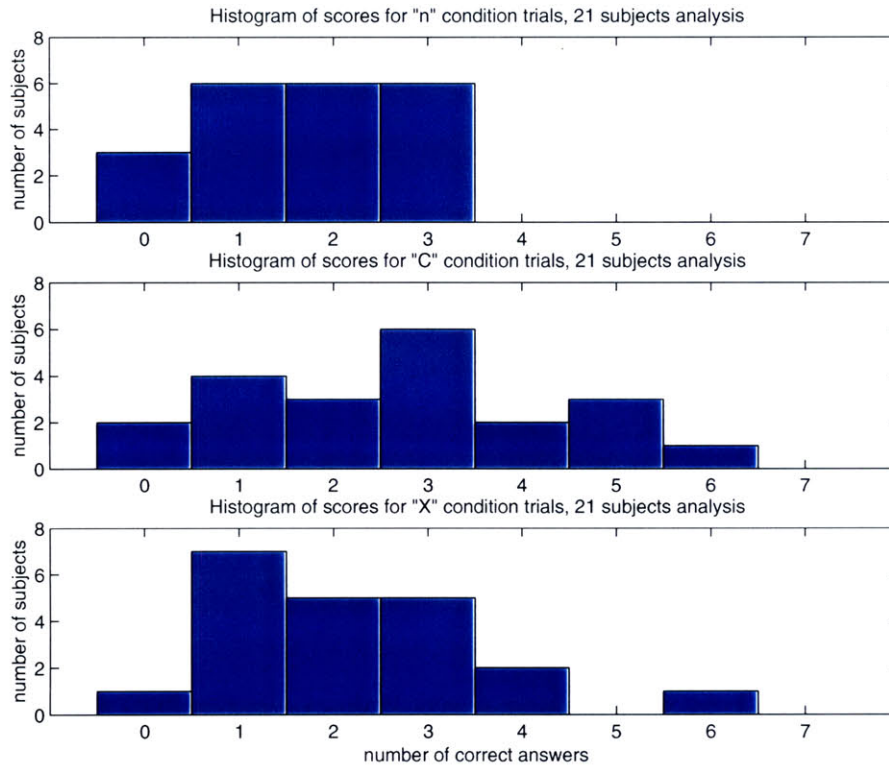


Figure B-4: Histograms showing subject performance on “n”, “C”, and “X” cue conditions for the masked short-duration visual cues experiment.

interference, the *improvement* in response to *incorrect* cuing was a surprise.

I suggest that an explanation for this counterintuitive effect lies in my choice of miscues. While the name shown in a miscued trial was incorrect for the particular face on the screen at the time, that name did come from the same memorization task, associated with some other face. So the slight increase in performance can be explained by an associative or spreading-activation theory of memory[1].

If the spreading-activation hypothesis is correct, I would not expect to see the same effect if the miscues were completely unrelated to the memorization set. Further study is needed to investigate the effect of different types of miscues.

## B.4 Overt Visual Cuing Experiment

In this experiment, the procedure of the previous experiment was replicated, but with the masking imagery replaced by the cues. The result was a procedure in which was identical

in all respects to the subliminal cuing experiment, except for the nature of the cues and a modified questionnaire and orientation, described below.

### **B.4.1 Hypotheses**

My hypothesis for the overt visual cuing experiment was that the effect of correct cues would be positive, and the effect of incorrect cues would be negative. Further, I hypothesized that for misleading cue condition trials, subjects would choose the incorrect cue as the answer more often than other names used in the experiment.

### **B.4.2 Orientation**

Subjects were told that the experiment involved testing the effects of a “potentially unreliable memory aid.” They were further advised that they were to do as well as they could on the memory test, regardless of what they might see on the head mounted display. As in the previous experiment, they were encouraged to guess if not sure.

### **B.4.3 Experiment-specific Procedure**

The procedure of the overt visual cues experiment was identical in all respects to the procedure for the subliminal visual cues experiment, with the exception of a modified questionnaire that omitted questions relating to subliminal cues.

### **B.4.4 Equipment**

The equipment used for this experiment was identical to that used for the subliminal visual cuing experiment.

### **B.4.5 Cuing Strategy**

In this experiment, the cuing strategy was identical to the previous experiment, except that the mask images were replaced with copies of the cue image. The effect was to cause the

cue to be clearly visible on the HMD for the full 10 second duration of each memory trial, or a blank screen for uncued trials.

### B.4.6 Results

As in the previous experiment, I performed a within-subjects and between subjects analysis, which yielded comparable results. What I found was that correct cues did improve performance, a large and highly significant effect,  $p = 0.000000008$  within subjects one-tail T test,  $p = 0.0000003$  between subjects one-tail T-test. However, much to my surprise misleading cues did not significantly decrease performance. See Figure B-5 and Table B.3 for the score histograms and between-subjects hypothesis testing, and Figure B-6 and Table B.4 for the effect histograms and within-subjects hypothesis testing.

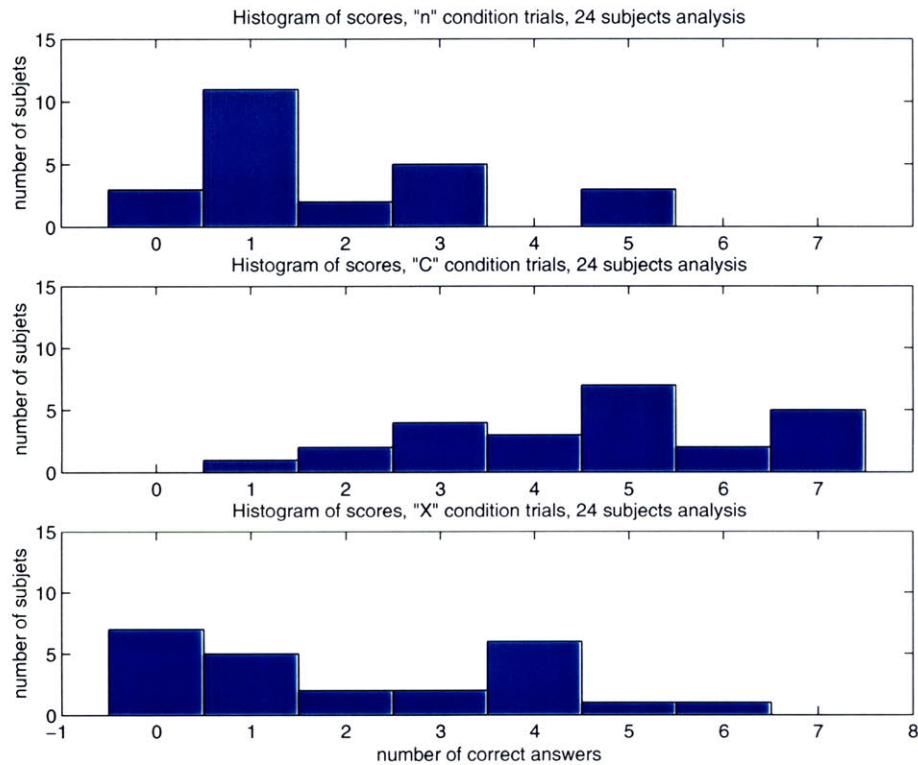


Figure B-5: Histograms showing subject performance on “n”, “C”, and “X” cue conditions for the overt visual cues experiment.

Although misleading cues did not decrease performance, subjects were vastly more likely to choose the incorrect cue as the answer than other names when answering incor-

cond.	mean	$\sigma^2$	Student T $p$
“n”	1.88	2.38	(one-tail) $3.4E^{-7}$ (two-tail) 0.34
“C”	4.63	3.11	
“X”	2.08	3.64	

Table B.3: Between-subjects hypothesis testing for overt visual cues.

rectly on a miscued trial.

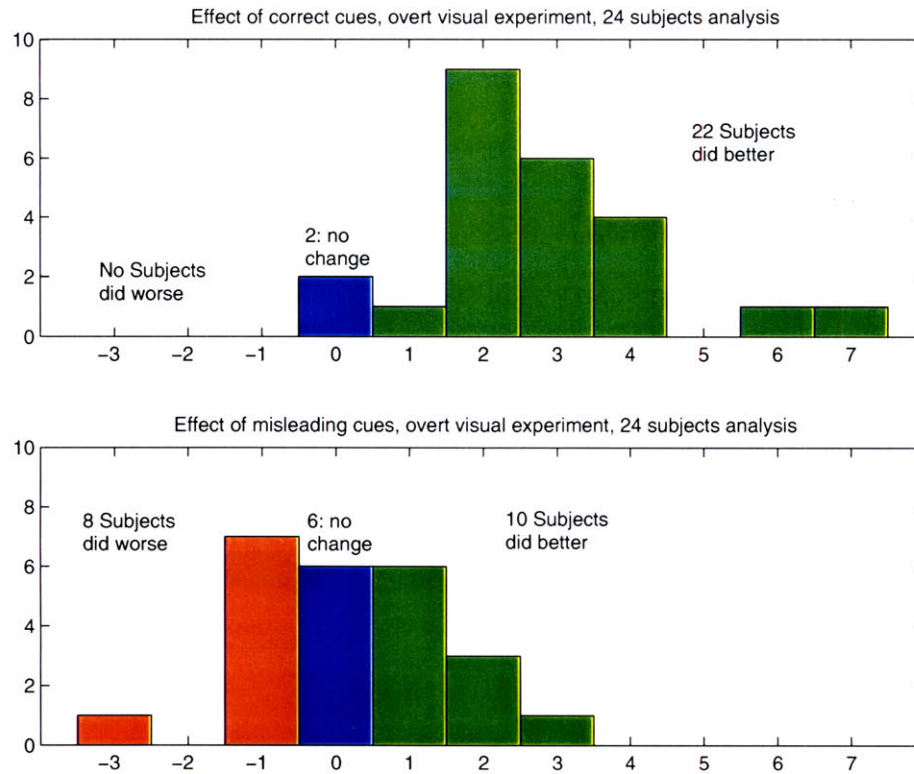


Figure B-6: Histograms comparing the effect of correct (“C”) and incorrect (“X”) overt visual cues.

## B.5 Masked Superliminal Audio Cues Experiment

In this experiment, the procedure of the subliminal visual cuing experiment was replicated, but with the masked visual cues replaced by white-noise masked superliminal audio cues delivered using a mono-aural (one ear only) headset. The post-trials questionnaire was modified to reflect the audio cuing apparatus.

cond.	mean	effect	effect $\sigma^2$	Student T $p$
“n”	1.88			
“C”	4.63	2.75	2.54	(one-tail) $8.3E^{-9}$
“X”	2.08	0.21	1.82	(two-tail) 0.22

Table B.4: Within-subjects hypothesis testing for overt visual cues.

### B.5.1 Hypotheses

I expected similar results for my masked superliminal audio cuing experiment as for the subliminal visual cues experiment. I hypothesized that correct cues would improve performance and that misleading cues would have a slightly negative effect. Further, I hypothesized that, as in the visual cues experiment, misleading cues would not result in subjects being more likely to choose the misleading cue when answering incorrectly on a misleading cue condition trial.

### B.5.2 Orientation

Subjects were told that the experiment involved testing the effects of subliminal cues on memory support, but were not told anything about the nature or content of the cues. They were further advised to do as well as they could on the memory test, regardless of what they might hear on the headset.

### B.5.3 Experiment-specific Procedure

#### Setting

The experiment took place in a somewhat controlled audio environment of a quiet office with the door closed. Background noise in this setting included fan noise from computers, ventilation noise, and typing noise caused by the subject themselves entering memory trials answers or other computer users in the office).

This setting was chosen to provide a balance between the desire for some control over the audio environment and the desire to test the cuing strategy in something like “real world” conditions.

## **Calibration**

After orientation, the subject was asked which hand was dominant (this was confirmed by experimenter observation of the subject's signing the consent form) and the headphone was placed on the opposite ear. Placing the headphone relative to hand dominance was intended to control for the effects of handedness on language processing in the brain[32].

The subject then underwent a hearing test procedure designed to calibrate the intensity (loudness) of the cues with respect to the mask noise. The goal was to pitch the level of the cue above the threshold of detection, but below the threshold of comprehension. (See the description of the calibration under "Cuing Strategy," below.

## **The Need for Calibration**

In the subliminal visual cuing experiment no attempt was made to adjust the duration or masking parameters to particular subjects, as the experience with the pilot[12] suggested that a single set of parameters would work well for the majority of subjects, which indeed it appeared to do.

However, research for the audio cuing experiment suggested that to be effective, cues would need to be pitched above the threshold of detection, but below the threshold of comprehension[44, 27].

In my pilot, I encountered a substantial variation in the sensitivity of my pilot subjects' hearing. Some subjects (myself included) could comprehend cues at very low signal-to-noise (S/N) ratios with respect to the masking noise level. Others could not detect cues (let alone comprehend them) except at much higher S/N ratios. Thus, in order to create the same subjective cuing experience for my subjects I decided that a calibration procedure was needed.

I chose to use the repetition of example cues rather than a pure-tone detection or other calibration procedure because the phenomenon of interest was the subject's ability to detect and comprehend the cues actually used in the experiment.

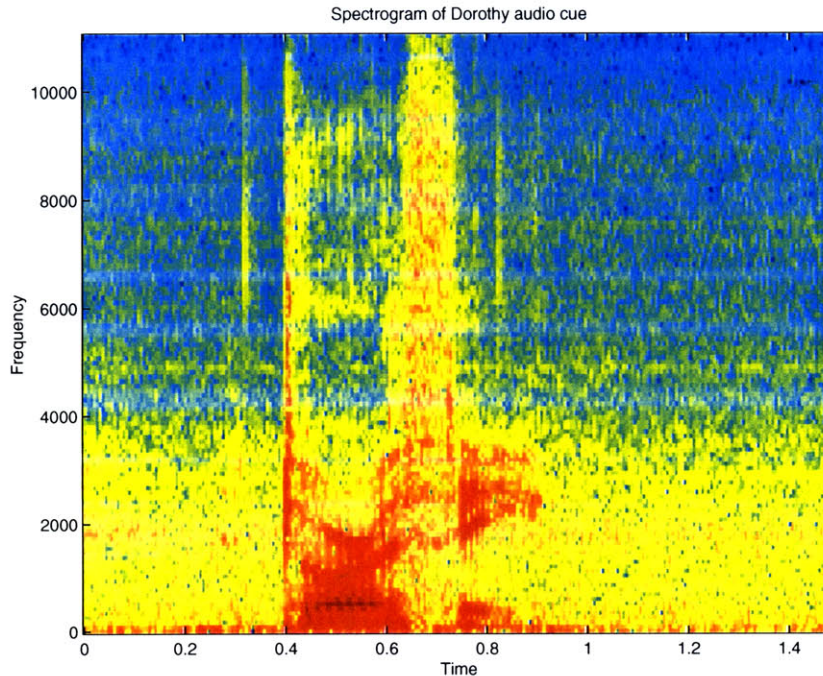


Figure B-7: Spectrogram of “Dorothy” audio cue used in masked superliminal audio cues experiment. The time axis is in seconds.

### B.5.4 Equipment

The equipment used for the audio cuing experiment consisted of a standard microphone/headphone headset of the kind used for hands-free phone use or computer-based audio chat. The audio signal was generated using a conventional built-in laptop audio system with headphone jack.

### B.5.5 Cuing Strategy

The cuing strategy for this experiment was a masked superliminal audio cue. These cues consisted of names spoken by the experimenter and recorded using a headset microphone, a laptop, and the Enchantment Signal system. The names were recorded in a single session, with the experimenter repeating the complete list several times. The recordings were digitized at 22kHz with 16bit resolution, and edited in Matlab to produce a collection of sound samples of approximately uniform volume and duration. See Figure B-7 for the spectrogram of an example cue.

The masking noise was produced in real-time by a 22kHz 16-bit random number gen-

erator, resulting in flat spectrum noise. The cues and masking stimuli were mixed in real time using the Enchantment Signal system, allowing the levels to be adjusted at run-time through an Enchantment Whiteboard interface. This real-time mixing capability afforded by the Enchantment system could be used to dynamically adapt cuing levels to ambient audio conditions, were the masked audio cuing process proven to be effective. However, for the experiment I conducted no attempt was made to adapt masking to ambient noise conditions after the initial calibration, described below.

### **Calibration Procedure**

The subject was asked to adjust the headphone until comfortable, and to confirm that they could clearly hear the quiet white noise mask “hiss.” The level of the white noise mask was chosen by the experimenter to be louder than the quiet office background noise, but substantially quieter than ordinary conversation.

To calibrate the level of the cues, the experimenter asked the subject to close their eyes and focus on the mask noise. The experimenter then played a sequence of example cues (cues identical in type to those used in the experiment, but for names not in the memorization set) at a fixed level with respect to the white noise mask, and asked subjects if they could hear anything other than the mask, and if so, to describe what they heard.

If the subjects failed to report hearing anything other than the mask, the loudness of the cuing was increased and the example cues repeated. The procedure was stopped when the subject reported being able to hear something voice like, but before being able to comprehend what might be said.

In this experiment, the cuing strategy was identical to the previous experiment, except that the mask images were replaced with copies of the cue image. The effect was to cause the cue to be clearly visible on the HMD for the full 10 second duration of each memory trial, or a blank screen for uncued trials.

### **Cue Delivery**

During the memory trials, subjects received five cues per trial, one masked cue every two seconds. As in the previous experiments, the cues were either a correct name, a misleading

name, or no cue (a sample of digitized quiet room noise comparable to the background noise of the cue recordings) depending on trial condition.

The number of cues per memory trial in this experiment was half the number in the subliminal visual cuing experiment because the longest recorded cue was approximately a second and a half long, and thus a delay of two seconds between the start of each cue was required.

### B.5.6 Results

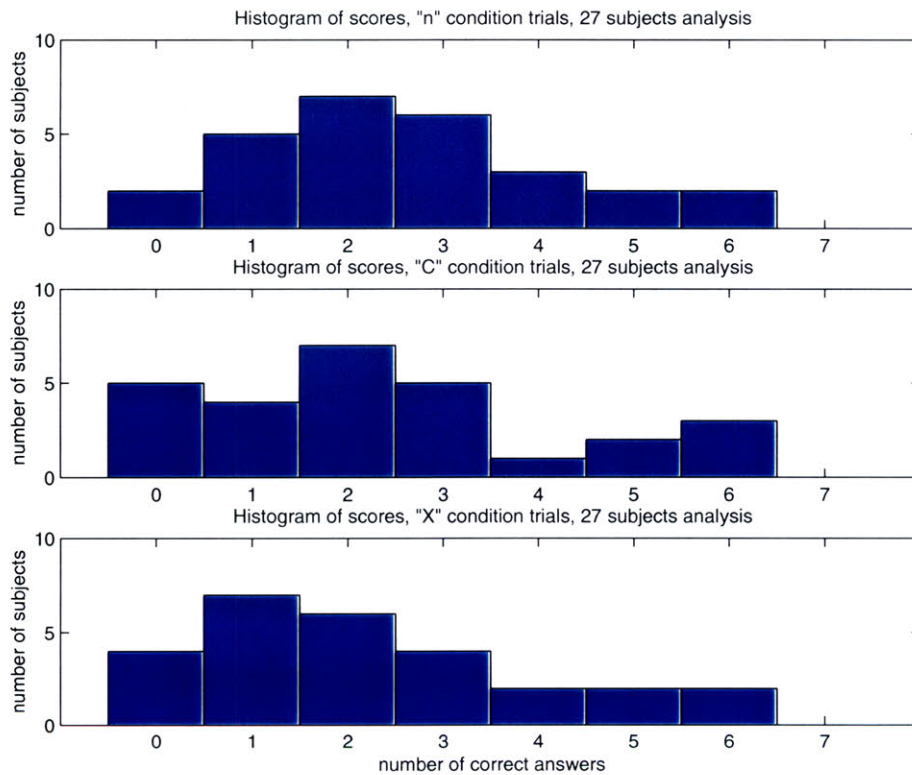


Figure B-8: Histograms showing subject performance on “n”, “C”, and “X” cue conditions for the masked superliminal audio cues experiment.

cond.	mean	$\sigma^2$	Student T <i>p</i>
“n”	2.63	2.70	
“C”	2.41	3.64	(one-tail) 0.68
“X”	2.26	3.20	(two-tail) 0.78

Table B.5: Between-subjects hypothesis testing for masked superliminal audio cues.

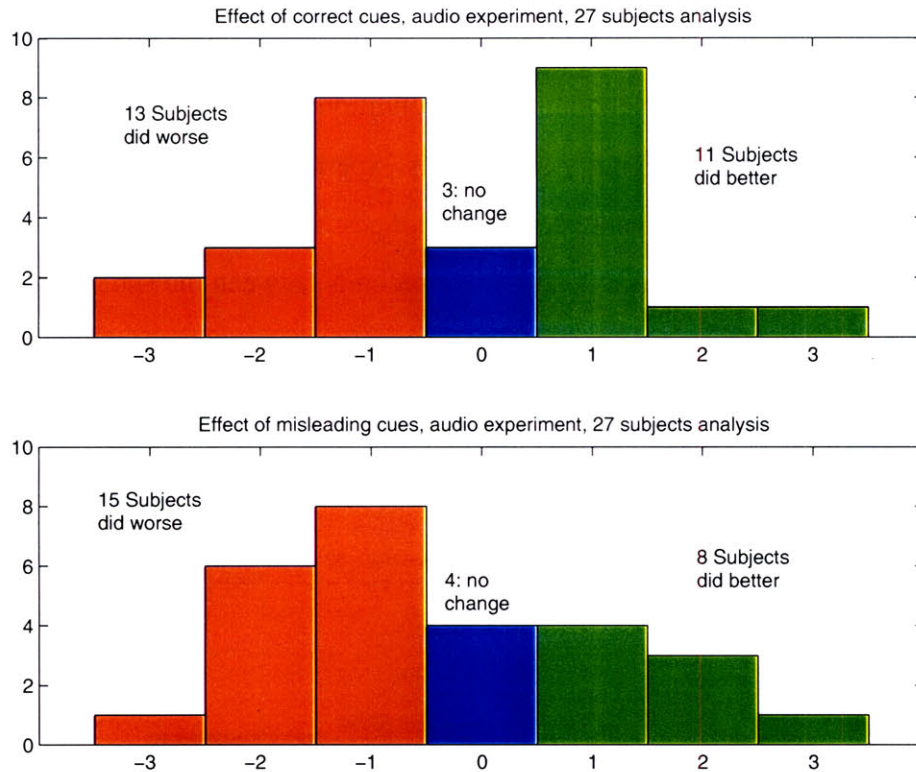


Figure B-9: Histograms comparing the effect of correct (“C”) and incorrect (“X”) masked superliminal audio cues.

cond.	mean	effect	effect $\sigma^2$	Student T $p$
“n”	2.63			
“C”	2.41	-0.22	2.26	(one-tail) 0.76
“X”	2.26	-0.37	2.40	(two-tail) 0.11

Table B.6: Hypothesis testing for masked superliminal audio cues.

As in the previous two experiments, I performed a within-subjects and between subjects analysis, which yielded comparable results. What I found was that correct cues did not significantly improve performance, nor did misleading cues significantly decrease performance. The incidence of choosing misleading cues on misleadingly cued trials was at chance levels.

## B.6 Overall Experimental Analysis

All three experiments shared the same procedure, with the only differences being the differing methods of cued delivery. As such, these three experiments can be considered a

single larger experiment with three cuing conditions: masked short-duration visual cues, overt visual cues, and mask superliminal audio cues.

One of the statistics gathered was the degree to which subjects were misled by miscues, which is to say the frequency of choosing the suggested miscue for the answer on an “X” condition trial. My hypothesis was that subjects would be more misled by overt cues than subliminal cues.

cond.	DOF	mean	$\sigma^2$
sub. vis.	20	0.05	0.05
overt vis.	23	1.79	2.87
sub. audio	26	0.07	0.07

Table B.7: Frequency of misled answers on “X” condition trials across cuing-method conditions.

The frequency of misled answers on “X” condition trials is presented in Table B.7. Although the differing variances between the overt visual cue condition and the other two conditions make this an inappropriate candidate for a three-way T-test, formal hypothesis testing is not really necessary here. Subjects in the overt visual cue condition were between 25 and 35 times more likely to be misled by overt visual cues than either of the two subliminal cuing methods.

In comparing the results from the three experiments, I noticed that while the mean of the “n” condition trials was similar for the two visual cuing experiments (1.71 on the subliminal visual cues and 1.88 on the overt visual cues) it was almost a full question larger on the masked superliminal audio cues experiment, 2.63, and in fact is more similar to subject performance on the “C” trials of the masked visual cues experiment. One explanation is that the subjects that I ran in the audio cues experiment simply had a better memory than the subjects in the overt cues experiment. However, these subjects were recruited from the same population and in the same way as the previous two experiments, so this explanation, while possible, seems somewhat unlikely. Another possibility is that the masked superliminal audio cues did have an effect, but that the effect “spilled over” into the uncued control condition trials. This would explain why this group’s performance on the control condition is similar to the performance of the other subjects on the correctly cued condition in the masked visual cues experiment.

Such a spill-over effect is plausible given that visual and auditory short-term memory decay at different rates: 10s of ms for visual short-term memory, and seconds for auditory memory[53]. As a result, it may be possible given the design I employed that cues could effect performance on a subsequent uncued trials through an associative memory effect. A new experimental design, possibly a between-subjects design, would be needed to test this hypothesis.

## **B.7 Experimental Conclusions**

### **B.7.1 Subliminal Visual Cuing Experiment Conclusions**

In the masked short-duration visual cues experiment, subjects did significantly better on the “C” trials than the “n” trials. The magnitude of the effect was 1.00, or roughly a factor of 1.58 times improvement compared to the control condition. This suggest that correct masked short-duration visual cues may be effective in supporting memory recall in a just-in-time cuing application.

The misleading cues in our “X” condition did not interfere with correct recall; indeed, a small (though borderline significant) improvement was observed, as discussed in Section B.3.7 above. In addition, subjects were very unlikely to answer the suggested miscue on “X” condition trials. Indeed, this happened only once in the entire experiment – a result consistent with guessing a common name at chance levels.

This is encouraging, as it suggests that incorrect masked short-duration visual cues do not interfere with recall, and may (under some circumstances) even support it. Further, while the intent of these cues was to be misleading, the results suggest that subjects are quite unlikely to be misled by masked short-duration cues.

### **B.7.2 Overt Visual Cuing Experiment Conclusions**

In the overt visual cuing experiment, subjects did significantly better on the “C” condition trials than the condition “n” trials, as expected. The magnitude of this effect was almost three times larger than the effect of correct cues in the masked short-duration visual cuing

experiment, suggesting (not surprisingly) that correct overt visual cues are more likely to aid recall than correct masked short-duration visual cues.

Subjects did not do significantly worse on the “X” condition trials than the “n” condition trials, which was a surprise. However, subjects were very likely to choose the suggested miscue when answering incorrectly on “X” trials, averaging 1.79 times out of a possible 7 per subject. (This compared to an average of 0.05 times out of 7 per subject on the masked short-duration visual cuing experiment, and an average of 0.07 times out of 7 on the superliminal masked audio cues experiment, as shown in Table B.7.

Subjects were told in advance that the memory support was potentially unreliable, and the data shows that they did not trust it completely — the effect of the misleading cues would have been significantly more negative if they had. Yet subjects were frequently misled when answering incorrectly. This data suggests a kind of over-reliance effect that interface designers should carefully consider when delivering overt cues with uncertainty.

One question this experiment does not address is not whether subjects would be more misled if they were told the memory support was reliable, or if the accuracy of the cuing were higher (perhaps perfect to begin with) so that subjects would be induced to trust it more. It would be straightforward to test these hypotheses with a new experimental design.

### **B.7.3 Masked Superliminal Audio Cuing Experiment Conclusions**

No significant effects on memory recall were shown by either “C” or “X” condition trials in the superliminal audio cuing experiment.

One possible explanation for this lack of effect is that the cues were not loud enough with respect to white noise mask. However, this seems unlikely, as the calibration procedure consistently resulted in cues that were above the level of detection, as reported by subjects on a post-test questionnaire. Indeed, if anything the calibration routine was too aggressive, since almost half the subjects run in this experiment reported being able to understand at least one cue, and were removed from the analysis as a result.

Another possible explanation is that the structure of the mask (white noise) inhibited semantic priming. A third possible explanation is a modality conflict — subjects learned

the material visually, yet were being cued aurally. And a fourth is that there was an effect, but that this effect “spilled over” into the control condition trials, as described above. It would be straightforward to design an experiment to test any of these hypotheses.

#### **B.7.4 Overall Experimental Conclusions**

Based on these results a Memory Glasses implementation could effectively use both masked short-duration visual cues and overt visual cues for just-in-time memory support. Surprisingly, plausible incorrect overt visual cues were not shown to decrease performance on the memory trials. However, there was a misleading effect of such cues, in that subjects who received such cues were vastly more likely to choose them as the answer when answering incorrectly than other plausible answers. The misleading effect of plausible miscues could be influenced by a range of factors that were not explored in this set of experiments.

No significant effect was shown for white-noise masked superliminal audio cues, but the disparity between control-condition performance on the superliminal audio cues experiment and the other experiments suggest that a cuing “spill over” effect is possible.

No attempt was made to measure user distraction in these experiments, as I was not confident I could effectively explore the effect of the cues and the distraction caused by them in the same experimental context. A speeded reaction task would need to be introduced into the design to explore the relative degree of distraction caused by different types of cuing.

All of these experimental results should be considered preliminary. My hope is that this work will inspire other researchers to attempt to replicate these results and further examine the effects of different types of cuing on memory for wearable computing memory support applications.

# Bibliography

- [1] John R. Anderson. A spreading activation theory of memory. *Journal of Verbal Learning and Verbal Behavior*, 1983.
- [2] Jan-Derk Bakker, Koen Langendoen, and Henk Sips. Lart: flexible, low-power building blocks for wearable computers. In *International Workshop on Smart Appliances and Wearable Computing (IWSAWC)*, Scottsdale, Arizona, April 2001.
- [3] Donald Eric Broadbent. *Perception and Communication*. Pergamon Press, London, 1958.
- [4] Peter J. Brown, John D. Bovey, and Xian Chen. Context-aware applications: From the laboratory to the marketplace. *IEEE Personal Communications*, 4(5):58–64, October 1997.
- [5] Stuart K. Card, Thomas P. Moran, and Allen Newell. *The Psychology of Human-Computer Interaction*. Lawrence Erlbaum Associates, 1983.
- [6] Christopher E. Carr, Steven J. Schwartz, and Ilia Rosenberg. A wearable computer for support of astronaut extravehicular activity. In *Proceedings of the 6th International Symposium on Wearable Computers* [28], pages 23 – 30.
- [7] Ivan Sergeyeovich Chardin. Spatial aspects of mobile ad hoc collaboration. Master’s thesis, MIT Media Lab, August 2003.
- [8] Tanzeem Choudhury and Alex Pentland. Sensing and modeling human networks using the sociometer. In *Proceedings of the 7th International Symposium on Wearable Computers* [29], pages 216–222.

- [9] Tanzeem Khalid Choudhury. *Sensing and Modeling Human Networks*. PhD thesis, MIT Media Laboratory, Cambridge, MA, 2004.
- [10] A. P. Dempster, N. M. Laird, and D. B. Rubin. Maximum likelihood from incomplete data via the EM algorithm. *JRSSB*, 39:1–38, 1977.
- [11] J. A. Deutsch and D. Deutsch. Comments on 'selective attention: perception or response?'. *Quarterly Journal of Experimental Psychology*, 19:362–363, 1967.
- [12] Richard DeVaul and Sandy Pentland. Toward the zero attention interface. In *Proceedings of the 6th International Symposium on Wearable Computers* [28], pages 141–142.
- [13] Richard W. DeVaul, Alex "Sandy" Pentland, and Vicka R. Corey. The memory glasses: Subliminal vs. overt memory support with imperfect information. In *Proceedings of the 7th International Symposium on Wearable Computers* [29].
- [14] Richard W. DeVaul, Michael Sung, Jonathan Gipps, and Sandy Pentland. MIThril 2003: Architecture and applications. In *Proceedings of the 7th International Symposium on Wearable Computers* [29], pages 4–11.
- [15] Richard Wayne DeVaul, Steven J Schwartz, and Alex "Sandy" Pentland. MIThril: Context-aware computing for daily life. unpublished white paper, 2001.
- [16] Norman F. Dixon. Perceptual and other related defenses. *Polish Psychological Bulletin*, 21(4):319–330, 1990.
- [17] John G. Dorsey and Daniel P. Siewiorek. Online power monitoring for wearable systems. In *Proceedings of the 6th International Symposium on Wearable Computers* [28], pages 137–138.
- [18] Richard O. Duda, Peter E. Hart, and David G. Stork. *Pattern Classification*. Wiley-Interscience, 2nd edition, 2000.

- [19] Joshua Edmison, Mark Jones, Zahi Nakad, and Tom Martin. Using piezoelectric materials for wearable electronic textiles. In *Proceedings of the 6th International Symposium on Wearable Computers* [28], pages 41–48.
- [20] F. Gemperle, C. Kashbach, J. Stivoric, M. Bauer, and R. Martin. Design for wearability. In *Proceedings of the 3rd International Symposium on Wearable Computers*, San Francisco, California, October 1999. IEEE Computer Society, IEEE Computer Society.
- [21] Ernest Govier and Marian Pitts. The contextual disambiguation of a polysemous word in an unattended message. *British Journal of Psychology*, 73(4):537–545, November 1982.
- [22] J. A. Gray and A. A. I. Wedderburn. Grouping strategies with simultaneous stimuli. *Quarterly Journal of Experimental Psychology*, 12:180–184, 1960.
- [23] David Greenfield. Lesson 173: Ssl and tls. *Network Magazine*, 17(12):26+28, December 2002. This appears to be the earliest SSL writup.
- [24] Anthony G. Greenwald. New look 3: Unconscious cognition reclaimed. *American Psychologist*, 47(6):766–779, 1992.
- [25] J. A. Groeger. Qualitatively different effects of undetected and unidentified auditory primes. *Quarterly Journal of Experimental Psychology*, 40(2):323–329, May 1988.
- [26] David Groome, Hazel Dewart, Anthony Esgate, Kevin Gurney, Richard Kemp, and Nicola Towell. *An Introduction to Cognitive Psychology: Processes and Disorders*, chapter 2.12. Psychology Press Ltd, 1999.
- [27] Jane L Harris, David Salus, Robert Rerecich, and Debra Larsen. Distinguishing detection from identification in subliminal auditory perception: A review and critique of merikle’s study. *The Journal of general psychology*, 123(1):41 – 50, 1996.
- [28] IEEE Computer Society. *Proceedings of the 6th International Symposium on Wearable Computers*, Seattle, Washington, October 2002. IEEE Computer Society.

- [29] IEEE Computer Society. *Proceedings of the 7th International Symposium on Wearable Computers*, White Plains, New York, October 2003. IEEE Computer Society.
- [30] Xybernaut Inc. Xybernaut mobile assistant. <http://www.xybernaut.com/>, 2001.
- [31] M. I. Jordan, editor. *Learning in Graphical Models*. Kluwer Academic Press, 1988.
- [32] Eric R. Kandel, James H. Schwartz, and Thomas M. Jessell, editors. *Principles of Neural Science*. Elsevier Science Publishing Co., Inc., third edition edition, 1991.
- [33] Edward Keyes. The Enchantment window manager: A wearable computing user interface, <http://albireo.mit.edu/enchantment.html>. Technical report, MIT, 2000.
- [34] Kristof Van Laerhoven, Albrecht Schmith, and Hans-Werner Gellersen. Multi-sensor context aware clothing. In *Proceedings of the 6th International Symposium on Wearable Computers* [28], pages 49–56.
- [35] D Lamble, T Kauranen, M Laakso, and H Summala. Cognitive load and detection thresholds in car following situations: safety implications for using mobile (cellular) telephones while driving. *Accident; analysis and prevention.*, 31(6):617–623, November 1999. ISSN: 0001-4575.
- [36] M. Lamming and M. Flynn. Forget-me-not: Intimate computing in support of human memory. In *FRIEND21: Inter. Symp. on Next Generation Human Interface*, pages 125–128, Meguro Gajoen, Japan, 1994.
- [37] V. R. Lesser, R. D. Fennell, L. D. Erman, and D. R. Reddy. Organization of the hearsay ii speech understanding system. In *IEEE Symposium on Speech Recognition Contributed Papers*. IEEE, IEEE, 15–19 April 1974.
- [38] Steve Litchfield. The history of psion (from 1980 to 1997). *Palmtop magazine*, 1997.
- [39] Gary Lynch. Memory enhancement: the search for mechanism-based drugs. *Nature Neuroscience*, 5 Supplement:1035 – 1038, November 2002.

- [40] Anthony J. Marcel. Conscious and unconscious perception: An approach to the relations between phenomenal experience and perceptual processes. *Cognitive Psychology*, 15(2):238–300, April 1983.
- [41] Anthony J. Marcel. Conscious and unconscious perception: Experiments on visual masking and word recognition. *Cognitive Psychology*, 15(2):197–237, April 1983.
- [42] T. L. Martin and D. P. Siewiorek. Non-ideal battery properties and low power operation in wearable computing. *Digest of Papers. Third International Symposium on Wearable Computers. IEEE Comput. Soc.*, pages 101–6, 1999. Inst. for Complex Eng. Syst., Carnegie Mellon Univ., Pittsburgh, PA, USA.
- [43] A J McKnight and A S McKnight. The effect of cellular phone use upon driver attention. *Accident; analysis and prevention.*, 25(3):259–365, June 1993.
- [44] Phillip M. Merikle. Subliminal auditory messages: An evaluation. *Psychology and Marketing*, 5:355–372, 1988.
- [45] Gorlick Mm. Electric suspenders: a fabric power bus and data network for wearable digital devices. *Digest of Papers. Third International Symposium on Wearable Computers. IEEE Comput. Soc.*, pages 114–21, 1999. Aerosp. Corp., El Segundo, CA, USA.
- [46] Kevin Murphy. The bayes net toolbox for matlab. *Computing Science and Statistics*, 33, 2001.
- [47] L. Najjar, C. Thompson, and J. Ockerman. A wearable computer for quality assurance inspectors in a food processing plant. In *IEEE Intl. Symp. on Wearable Computers*. IEEE Computer Society, 1997.
- [48] D. Norman. *The Design of Everyday Things*. Doubleday Currency, New York, 1988.
- [49] Donald A. Norman. Designing the future. *Scientific American*, 273:194, September 1995.

- [50] Jennifer Jo Ockerman. *Task guidance and procedure context: Aiding workers in appropriate procedure following*. PhD thesis, Georgia Institute of Technology, 2000.
- [51] Alan J. Parkin. *Memory And Amnesia: An Introduction*. Blackwell Publishers, Ltd., 1997.
- [52] Jason Pascoe. Adding generic contextual capabilities to wearable computers. In *Digest of Papers. Second International Symposium on Wearable Computers*, pages 92–99. IEEE, October 1998.
- [53] Harold Pashler, editor. *Attention*. Psychology Press, 1998.
- [54] Ronald C. Petersen, editor. *Mild Cognitive Impairment: Aging to Alzheimer's Disease*. Oxford University Press, 2003.
- [55] P. J. Phillips. The feret database and evaluation procedure for face-recognition algorithms. *Image and Vision Computing Journal*, 16(5):295–306, 1998.
- [56] G. P. Picco, A. L. Murphy, and G. C. Roman. Lime: Linda meets mobility. In *Proceedings of the 21st International Conference on Software Engineering (ICSE '99)*. IEEE Comput. Soc. Tech. Council on Software Eng.; ACM SIGSOFT; ACM SIGAda; ACM SIGPLAN, ACM, May 1999.
- [57] E. R. Post, M. Orth, P. R. Russo, and N. Gershenfeld. E-broidery: Design and fabrication of textile-based computing. *IBM Systems Journal*, 39(3 & 4), 2000.
- [58] Janet Rae-Dupree. Palmpilot is handing industry a big lesson. *KNIGHT-RIDDER NEWS SERVICE, Philadelphia Newspapers, Inc.*, 1997.
- [59] R. Rajkumar, M. Gagliardi, and Lui Sha. The real-time publisher/subscriber inter-process communication model for distributed real-time systems: design and implementation. In *Proceedings. Real-Time Technology and Applications Symposium*, pages 66–75. IEEE Comput. Soc. Tech. Committee on Real-Time Systems, May 1995.

- [60] B. Rhodes. The wearable remembrance agent: A system for augmented memory. *Personal Technologies*, 1(4):218–224, March 1997.
- [61] B. Rhodes. *Just-In-Time Information Retrieval*. PhD thesis, MIT Media Laboratory, Cambridge, MA, May 2000.
- [62] Brad Rhodes and Thad Starner. Remembrance agent: a continuously running automated information retrieval system. In *Proceedings of the First International Conference on the Practical Application of Intelligent Agents and Multi Agent Technology (PAAM '96)*, pages 487–495, London, April 1996.
- [63] Bradley Rhodes. The wearable remembrance agent: A system for augmented memory. *Personal Technologies Journal Special Issue on Wearable Computing*, 1:218–224, 1997.
- [64] The MIThril project website. <http://www.media.mit.edu/wearables/mithril>, 2001.
- [65] A Sahgal. A critique of the vasopressin-memory hypothesis. *Psychopharmacology (Berl)*, 83(3):215–28, 1984.
- [66] Daniel L. Schacter. *The Seven Sins of Memory*. Houghton Mifflin Company, 2001.
- [67] Steven J. Schwartz and Alex Pentland. The smart vest: Towards a next generation wearable computing platform. Technical report, MIT Media Lab, 1999.
- [68] Philips Semiconductors. The i<sup>2</sup>c-bus specification version 2.1. Technical report, 2001.
- [69] Claude E. Shannon and Warren Weaver. *The Mathematical Theory of Communication*. University of Illinois Press, 1949.
- [70] R. M. Shiffrin and W. Schneider. Controlled and automatic human information processing: II. perceptual learning, automatic attending, and a general theory. *Psychological Review*, 84:127–190, 1977.
- [71] A. Smailagic and D. Siewiorek. The CMU mobile computers: A new generation of computer systems. In *COMPCON '94*. IEEE Computer Society Press, February 1994.

- [72] Paul R. Solomon, Felicity Adams, Amanda Silver, Jill Zimmer, and Richard DeVaux. Ginkgo for memory enhancement: A randomized controlled trial. *JAMA*, 288(7):835–40, Aug 21 2002.
- [73] Flavia Sparacino, Rich DeVaul, Chris Wren, Ron MacNeil, Gloriana Davenport, and Alex Pentland. Wearable city of news. In *SIGGRAPH 99 Visual Proceedings, Emerging Technologies*, Los Angeles, 1999. ACM SIGGRAPH.
- [74] Jonathan D. Spence. *The Memory Palace of Matteo Ricci*. Penguin Books, 1985.
- [75] T. Starner. *Wearable Computing and Context Awareness*. PhD thesis, MIT Media Laboratory, Cambridge, MA, May 1999.
- [76] T. Starner, D. Kirsch, and S. Assefa. The locust swarm: An environmentally-powered, networkless location and messaging system. Technical Report 431, MIT Media Lab, Perceptual Computing Group, April 1997. Presented ISWC’97.
- [77] Thad Starner. Lizzy: MIT’s wearable computer design 2.0.5. <http://wearables.www.media.mit.edu/projects/wearables/lizzy/>, 1999.
- [78] Thad Starner, Bernt Schiele, and Alex Pentland. Visual context awareness in wearable computing. In *Digest of Papers. Second International Symposium on Wearable Computers*, pages 50–57. IEEE Computer Society, October 1998.
- [79] J. R. Stroop. Studies of interference in serial verbal reactions. *Journal of Experimental Psychology*, 18:643–662, 1935.
- [80] L. Sydiinheimo, M Salmimaa, J Vanhala, and M Kivikoski. Wearable and ubiquitous computer aided service, maintenance and overhaul. *IEEE International Conference on Communications*, 3(99CH36311):2012–17, 1999. Inst. of Electron., Tampere Univ. of Technol., Finland.
- [81] Aaron Toney, Lucy Dunne, Bruce H. Thomas, and Susan P. Ashdown. A shoulder pad insert vibrotactile display. In *Proceedings of the 7th International Symposium on Wearable Computers* [29], pages 34–44.

- [82] A. M. Treisman. Verbal cues, language and meaning in selective attention. *American Journal of Psychology*, 77:206–219, 1964.
- [83] F. Layne Wallace, J. Michael Flanery, and Gerald A. Knezek. The effect of subliminal help presentations on learning a text editor. *Information Processing and Management*, 27(2/3):211–218, 1991.
- [84] R. Want, W. Schilit, N. Adams, R. Gold, K. Petersen, D. Goldberg, J. Ellis, and M. Weiser. An overview of the parctab ubiquitous computing experiment. *IEEE Personal Communications*, 2(6), December 1995.
- [85] Roy Want, Andy Hopper, Veronica Falcao, and Jonathan Gibbons. The active badge location system. *ACM Transactions on Information Systems*, 10(1):91–102, January 1992.
- [86] Josh Weaver, Ari Y. Benbasat, and Richard W. DeVaul. The crystal tag system. <http://www.media.mit.edu/~rich/>, 1999.
- [87] Joshua A. Weaver. A wearable health monitor to aid parkinson disease treatment. Master’s thesis, Massachusetts Institute of Technology, 2003.
- [88] C. Wickens. Situation awareness: impact of automation and display technology. In *NATO AGARD Aerospace Medical Panel Symposium on Situation Awareness, AGARD CP-575*, Neuilly-Sur-Seine, France, 1996.
- [89] C. Wickens and T. Prevelt. Exploring the dimensions of egocentricity in aircraft navigation displays: influences on local guidance and global situation awareness. *J. of Experimental Psychology: Applied*, 1(2):110–135, 1995.
- [90] J. M. Von Wright, K. Anderson, and U. Stenman. Generalisation of conditioned GSRs in dichotic listening. In P. M. A. Rabbitt and S. Dornic, editors, *Attention and Performance*. Academic Press, London, 1985.
- [91] Frances A. Yates. *The Art of Memory*. University of Chicago Press, 1974.