Application Design for Wearable and Context-Aware Computers

Asim Smailagic and Daniel Siewiorek

Institute for Complex Engineered Systems and Human Computer Interaction Institute

Carnegie Mellon University, Pittsburgh, PA

Abstract This paper describes a taxonomy of problem solving capabilities for wearable and context aware computers developed from our iterative design methodology, with a wide variety of end users, mainly mobile workers. The taxonomy is illustrated by wearable systems evolving from basic stored information retrieval through synchronous /asynchronous collaboration within a team, to context aware platforms with a proactive assistant. Example evaluation methods illustrate how user testing can quantify the effectiveness of wearable systems.
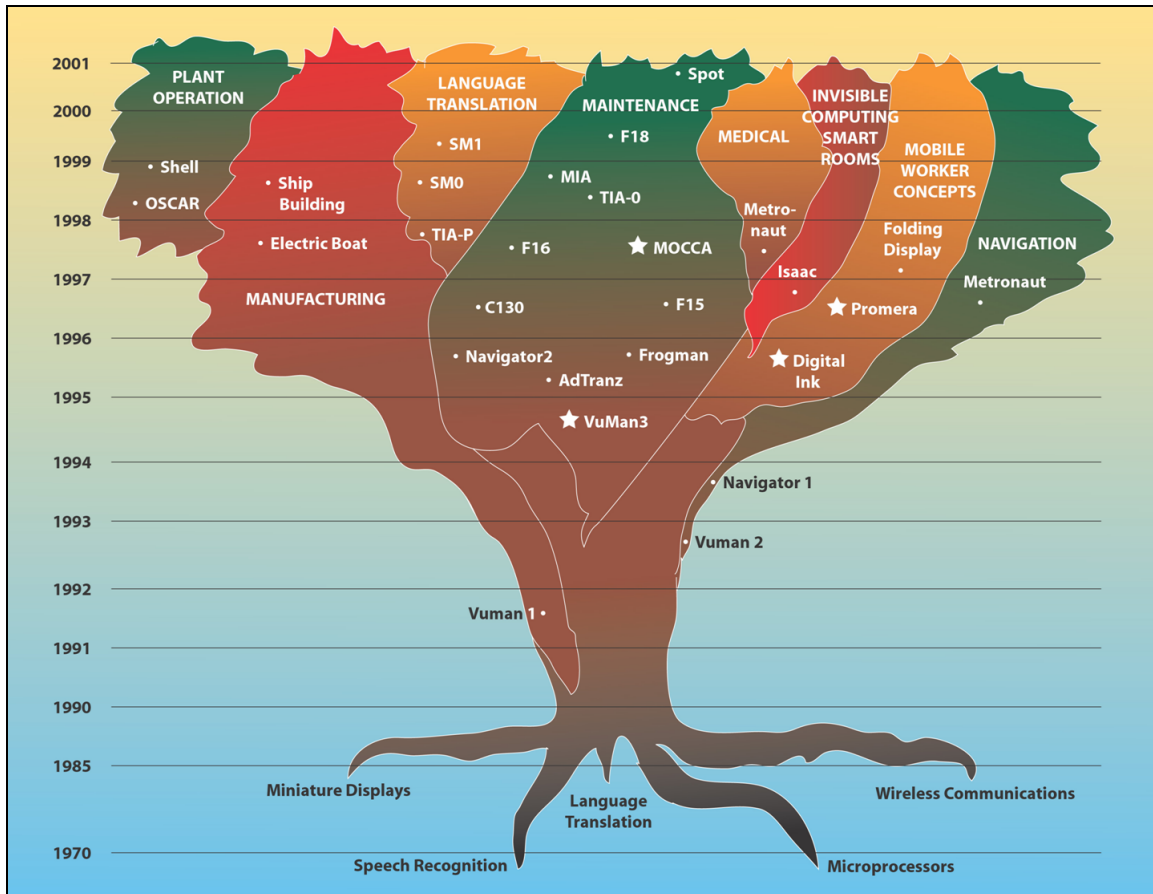
1. Introduction

Carnegie Mellon's Wearable Computers project is helping to define the future for not only computing technologies but also for the use of computers in daily activities. The goal is to develop a new class of computing systems with a small footprint that can be carried or worn by a person and be able to interact with computer-augmented environments [1]. Since users are an integral part of the system, techniques such as user centered design, rapid prototyping, and in-field evaluation are used to identify and refine user interface models that are useful across a wide spectrum of applications [2,3]. Over two dozen wearable computers have been designed and built over the last decade, with most tested in the field. The application domains range from inspection, maintenance, manufacturing, and navigation to on-the-move collaboration, position sensing, and real-time speech recognition and language translation. At the core of these paradigms is the notion that wearable computers should seek to merge the user's information with his or her work space. The wearable computer must blend seamlessly with existing work environments, providing as little distraction as possible. This requirement often leads to replacements for the traditional desktop paradigm, which generally require a fixed physical relationship between the user and devices such as a keyboard and mouse. Identifying effective interaction modalities for wearable computers, as well as accurately modeling user tasks in software, are among the most significant challenges in designing wearable systems.

The goals for this paper are: (1) present a map of wearable system functionality to application types and to (2) summarize examples of four user interface models.

2. Evolution of CMU Wearable Computers

The Family Tree of CMU Wearable Computers, as shown in Figure 1, classifies wearable computers into several application categories and presents their development over the last decade. The tree is modeled after the NSF "Family Tree" of early

computers circa 1955. Each name represents a distinct wearable computer placed under its corresponding application domain. The four wearable computer designs marked with a star have been awarded prestigious international design awards (VuMan 3, MoCCA, Digital Ink, and Promera).

Figure 1. Family Tree of CMU Wearable Computers

As shown in Figure 2, wearable computers at Carnegie Mellon have ranged from "proof of concept" to customer-driven systems design based on a task specification, to visionary design predicting the form and functionality of wearable computers of the future.

In order for pervasive/ubiquitous computing to reach its' potential, the average person should be able to take advantage of the information on or off the job. Even while at work, many people do not have desks and/or spend a large portion of their time away from a desk. Thus mobile access is the gating technology required to make information available at any place and at any time. In addition, the computing system should be aware of the user's context not only to be able to respond in an appropriate manner with respect to the user's cognitive and social state, but also to anticipate needs of the user.

Table 1 briefly summarizes the past forty years of user interface evolution. While technologies show doubling of capability every few years, it takes about 10 years for a

new user interface to become widely deployed. The extra time is required to working out technology bugs, reducing costs, and adapting applications to the new user interfaces. Currently speech and handwriting recognition are becoming common inputs.
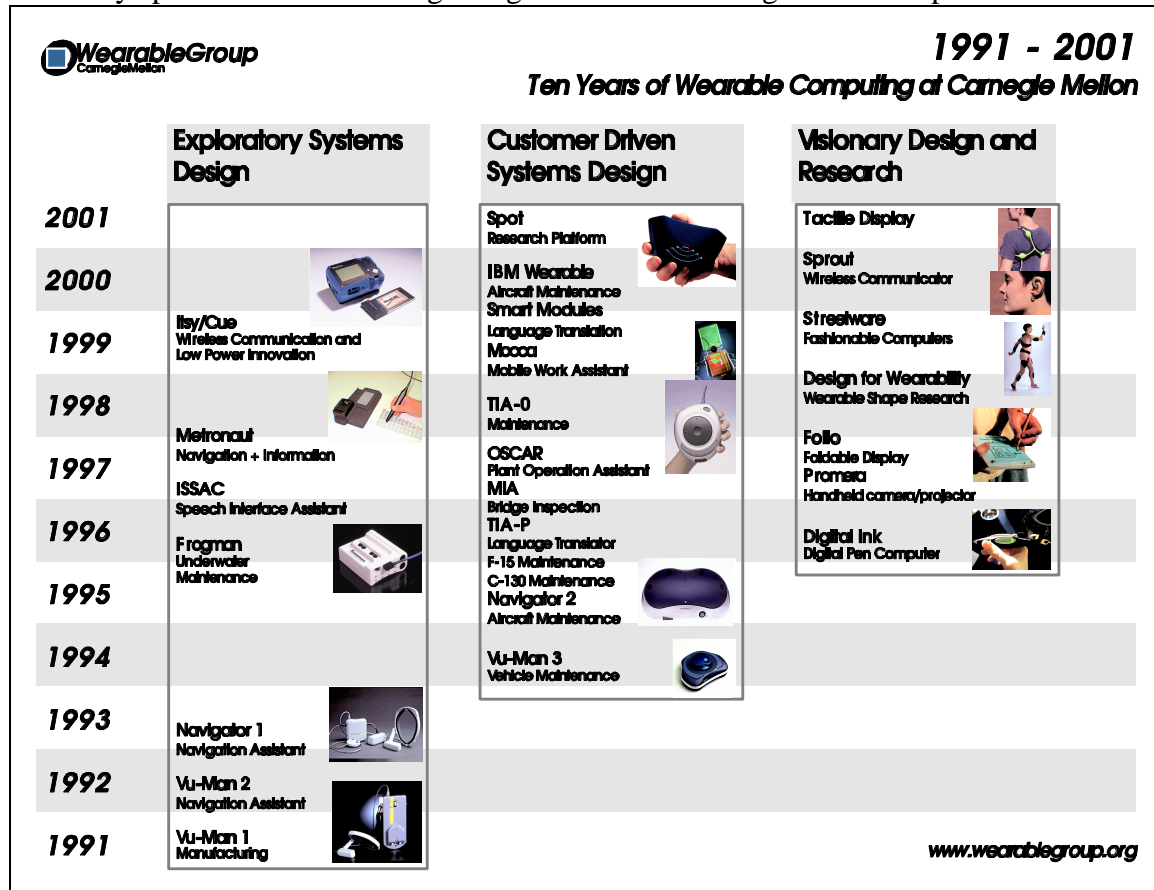

Figure 2. Ten Years of Wearable Computing at Carnegie Mellon

In the near future, position sensing, eye tracking, stereographic audio and visual output will be coupled with three dimensional virtual reality information.

Period of Introduction	Mainstream Acceptance	Input/Output/Information
1965	1975	Keyboard, alpha-numeric display, text
1980	1990	Keyboard/mouse, graphics display, icons
1995	2005 predicted	Handwriting/speech recognition, speech synthesis, multi-modal
2010 predicted	2020 predicted	Position sensing/eye tracking, stereo audio/video, 3D virtual reality

Table 1. User interface evolution

3. Issues in Wearable Computing

Today's computer systems distract a user in many explicit and implicit ways, thereby reducing their effectiveness. The systems can also overwhelm users with data leading to information overload. The challenge for human computer interaction design is to use advances in technology to preserve human attention and to avoid information saturation.

We have identified four design principles for mobile systems.

- User Interface models. What is the appropriate set of metaphors for providing mobile access to information (i.e., what is the next "desktop" or "spreadsheet")? These metaphors typically take over a decade to develop (i.e., the desktop metaphor started in early 1970's at Xerox PARC and required over a decade before it was widely available to consumers). Extensive experimentation working with end-user is required to define and refine these user interface models.

- Input/output modalities. While several modalities mimicking the input/output capabilities of the human brain have been the subject of computer science research for decades, the accuracy and ease of use (i.e., many current modalities require extensive training periods) are not yet acceptable. Inaccuracies produce user frustrations. In addition, most of these modalities require extensive computing resources which will not be available in light-weight, low-power wearable computers. There is room for new, easy-to-use input devices such as the dial developed at Carnegie Mellon University for list-oriented applications.

- Matched capability with application requirements. Many mobile systems attempt to pack as much capacity and performance in as small a package as possible. However, these capabilities are often unnecessary to complete an application. Enhancements such as full-color graphics not only requires substantial resources but also may actually compromise easy of use by generating information overload for the user. Interface design and evaluation should focus on the most effective means for information access and resist the temptation to provide extra capabilities simply because they are available.

- Quick interface evaluation methodology. Current approaches to evaluate a human computer interface require elaborate procedures with scores of subjects. Such an evaluation may take months and is not appropriate for reference during interface design. These evaluation techniques should especially focus on decreasing human errors and frustration.

Over the past decade, we have built wearable computers for over a dozen clients in diverse application areas. We have observed several functionalities that have proven useful across multiple applications. These functionalities form the basis for four User Interface Models, each with their unique user interface, input/output modality, and capability:

- Procedures – Text and Graphics. Maintenance and plant operation applications are characterized by a large volume of information that varies slowly over time. For example, even simple aircraft may have over a hundred thousand manual pages (like the aircraft manuals at US Airways hangar at Pittsburgh International Airport). But due to operational changes and upgrades, half of these pages are obsolete every six months for even very mature aircraft. Rather than distribute CD-ROMs for each maintenance person and running the risk of a maintenance procedure being performed on obsolete information, maintenance facilities usually maintain a centralized data base to which maintenance personnel make inquiries for the relevant manual sections on demand. A typical request consists of approximately ten pages of text and schematic drawings. Changes to the centralized information base can occur on a weekly basis. Furthermore, the trend is towards more customization in manufacturing. In aircraft manufacturing no two aircraft on an assembly line are identical. The aircraft may belong to different airlines or be configured for different missions. Customization extends to other industries. One leading manufacturer produces over 70,000 trucks per year, representing over 20,000 different configurations. The customer can select the transmission, the engine, and even the stereo system. In the near future trucks will be accompanied by their own documentation describing "as built", "as modified", and "as repaired". When personnel carrying out manufacturing or scheduled maintenance arrive for a days work they receive a list of job orders which describe the tasks and include documentation such as text and schematic drawings. Thus this information can change on a daily or even hourly basis.

- Master/Apprentice Help Desk. There are times, however, when an individual requires assistance from experienced personnel. Historically this assistance has been provided by an apprenticeship program wherein a novice observes and works with an experienced worker. A simple example of this is the "Help Desk" wherein an experienced person is contacted for audio and visual assistance in solving a problem.

- Team Maintenance/Collaboration. The Help Desk can service many people in the field simultaneously. Today, with down-sizing and productivity improvement goals, teams of people are geographically distributed yet are expected to pool their knowledge to solve immediate problems. An extension of Help Desk is a team of personnel such as field service engineers, police and firefighters, who are joining together to resolve an emergency situation. Information can be expected to change on a minute-by-minute and sometimes even second-by-second basis.

- Context Aware Collaboration – Proactive Synthetic Assistant. *Context-aware computing* describes the situation where a mobile computer is aware of its user's state and surroundings, and modifies its behavior based on this information. The system can monitor a user's state and act as a proactive assistant, linking information derived from many contexts, such as location and schedule. Distractions are even more of a problem in

mobile environments than desktop environments, since the user is often preoccupied with walking, driving or other real-world interactions. A ubiquitous computing environment that minimizes distraction should be context aware. If a human assistant were given such context, he or she would make decisions in a proactive fashion, anticipating user needs. The goal is to enable mobile computers to play an analogous role, exploiting context information to significantly reduce demands on human attention.

The remainder of the paper is organized as follows. Section 4 describes the four user interface models and gives examples of how they address the three design principles of user interface models, input/output modalities, and functional capability requirements. Section 5 provides examples of how systems can be evaluated for each of the four user interface models.

4. Example Systems

Four wearable computer user interface models will be illustrated using example CMU wearable computer systems. It is interesting to note that these user interface models can also be found in systems developed by other organizations. Figure 3 shows examples of wearable computer platforms, corresponding to each of the four user interface models. These examples include:

- Procedures – Text and Graphics
  - VuMan 3 text based inspection of U.S. Marines heavy military vehicles, CMU [4]
  - Navigator 2 graphical inspection of Boeing aircraft, CMU [1]
  - Georgia Tech wearable computer for quality assurance inspection in food processing plants [5]
- Master/Apprentice Help Desk
  - TIA-P (Tactical Information Assistant - Prototype) used for C-130 Help Desk, CMU [9]
  - NETMAN Enables Technicians in the field and office based experts to collaborate in realtime using audio and video, University of Oregon [10]
- Team Collaboration
  - MoCCA (Mobile Communication and Computing Architecture) to support collaboration of geographically distributed field engineers, CMU [8]
  - Land Warrior integrated infantry soldier system for close combat. Designed to avoid information overload [11]
- Synthetic Collaboration – Proactive Synthetic Assistant
  - GM/CMU Companion, a contextual car/driver interface that proactively assists the driver in managing information and communication
  - Touring Machine, combines the overlaid 3D graphics of augmented reality with the untethered freedom of mobile computing to support users in their everyday interactions with the world, Columbia University [12]

A synthetic assistant is a technology that allows a computer model of a human expert to interact conversationally, provide advice, read procedures and answer questions to a human. This technology is developed at CMU [9] [13].

Table 2 summarizes the four User Interface Models with respect to the first design principle, input/output modalities. The sources of knowledge for each of the models are also presented. The Knowledge Source and User Interface Models will be illustrated by system examples in Section 4.1 to 4.4.

Table 3 summarizes the third design principle, capability to fulfill requirements for the User Interface Models in Table 2. For example, the Master/Apprentice model employs static and synchronous expert functionality. The following subsections elaborate further and give examples of each problem solving capability. Figure 4 depicts the four Problem Solving capabilities in a state diagram. The system examples in the following four subsections will illustrate each capability in-turn.

User Interface Model	Data Representation	Knowledge Source	Input	Output
Pre-Stored Procedures: Text	Text	Task Specific/ Pre-stored Procedures/ Menu Selection Input	Buttons	Alphanumeric
Pre-Stored Procedures: Graphics	Bitmap		Mouse	Graphical User Interface
Master/ Apprentice Help Desk	Speech Synthesis	Archival	Speech	Multimedia
Team Collaboration	Pictures	Team "Corporate" Memory	Archival Data	Group Collaboration
Synthetic Collaboration	3D Animation	Real Time Physical & Social Context	Contextual Information	Proactive, Context Appropriate

Table 2. Input/Output modalities and information sources for the User Interface Models

User Interface Model	Problem Solving Capability			
	Static	Synchronous Expert	Asynchronous Expert	Proactive Assistant
Pre-Stored Procedures	X			
Master/Apprentice Help Desk	X	X		
Team Collaboration	X	X	X	
Synthetic	X	X	X	X

Collaboration				

Table 3. User Interface Models and required Problem Solving Capability
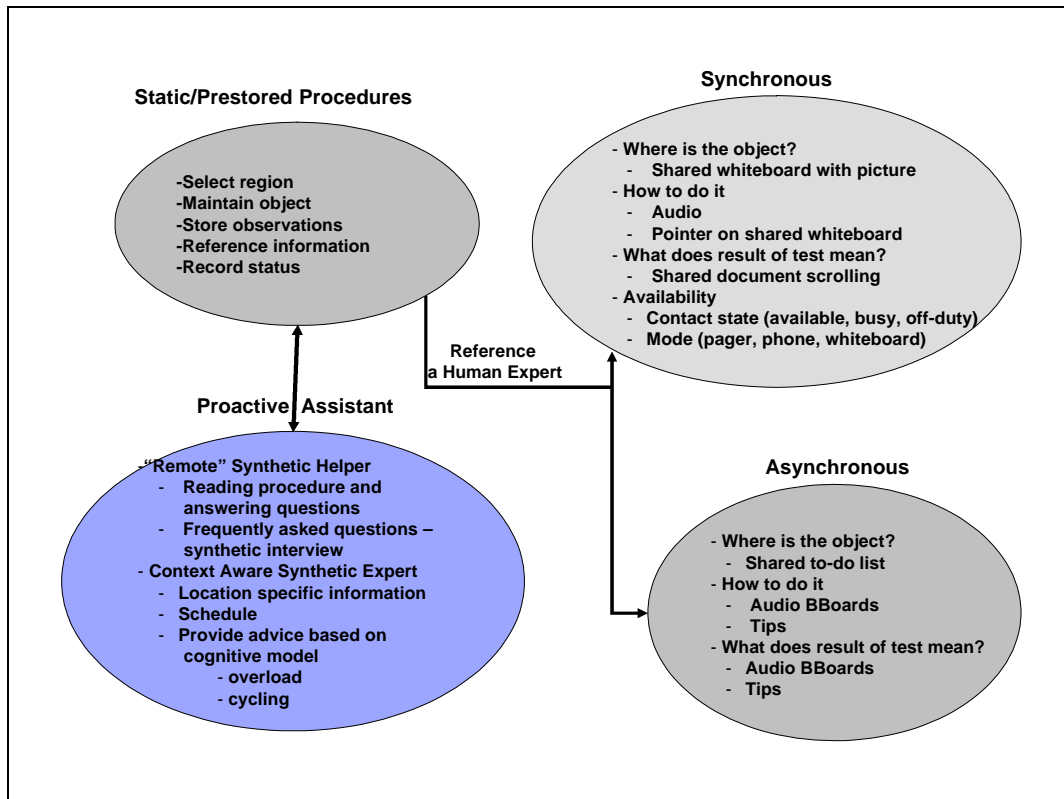

Figure 3. Examples of Wearable Computer Platforms
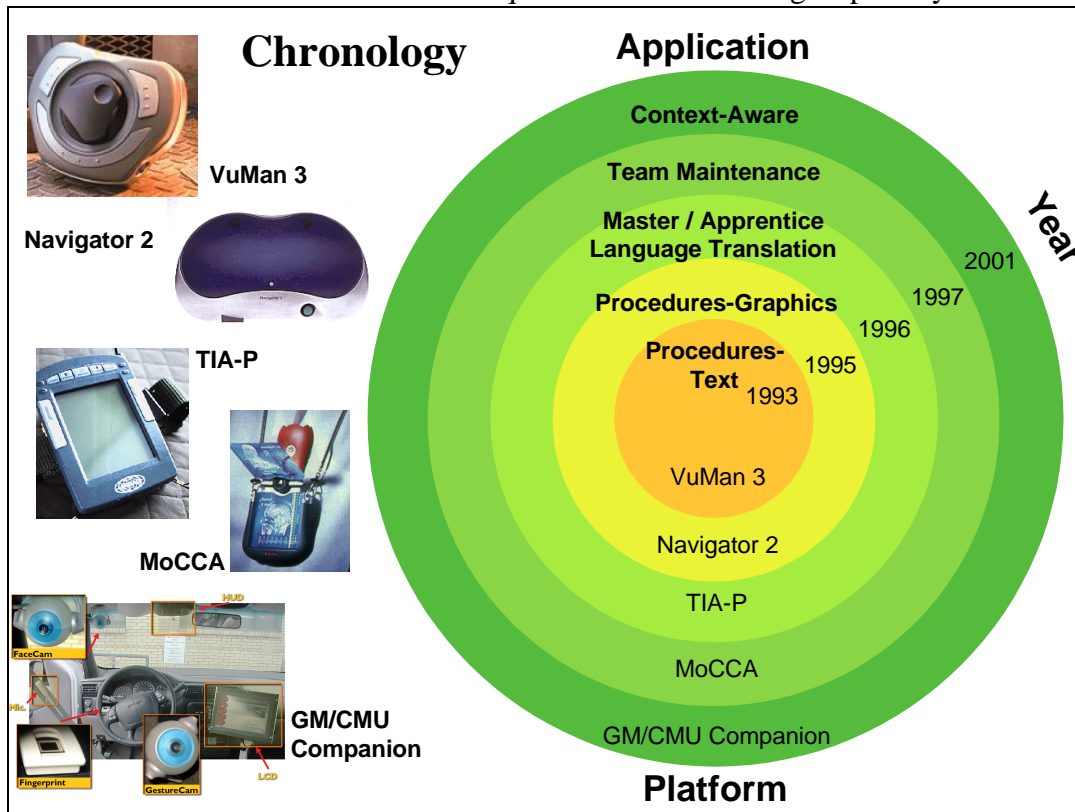
Figure 4. State diagram of Problem Solving Capabilities

4.1 Procedures with Static Pre-Stored Text/Graphics

The Pre-Stored capability is illustrated by the sheet metal inspection of a military aircraft. Approximately 100 defects are identified during an average 36 hour inspection. The inspection starts with selecting a region on the aircraft body, and proceeds with inspecting the object, referencing information from the manuals, archival storage of observations, and recording the status. Once the user chooses to begin an aircraft inspection, the field of interest is narrowed from major features (such as Left Wing, Right Tail) to more specific details (individual panes in the cockpit window glass, aircraft body polygons), as shown on Figure 5. The area covered by each defect as well as the type of defect, using a "How Malfunctioned" code such as Corroded, Cracked, or Missing, are recorded. To maximize usability, each item or control may be selected simply by speaking its name or, in the case of more complicated phonemes, a designated numeral. This two-dimensional selection method, in which defect locations are specified on a planar region, and overall user interface design have received very favorable feedback from the Boeing aircraft inspectors at McClellan Air Force Base, CA.
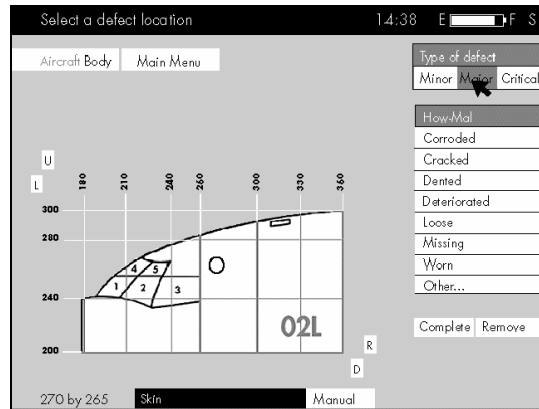
9

Figure 5. Example User Interface Screen for static/pre-stored information

4.2 Master/Apprentice (Live Expert) Help Desk

The C-130 project is designed to use collaboration to facilitate training and to increase the number of trainees per trainer. Inexperienced users are being trained to perform a cockpit inspection and the trainers are remotely located. The trainee loads the inspection procedures and performs the inspection. A desktop system manages the normal job order process and is used by the instructors to observe the trainee's behavior. In collaboration, the instructor looks over the shoulder (through a small video camera attached to the top of the trainee's head mounted display) and advises the trainee. In addition to a two way audio channel, the instructor can provide advice via a cursor for indicating areas on a captured video image which is being shared through a whiteboard. The instructor manages the sharing session and whiteboard. The trainee's use of the whiteboard is limited to observation.
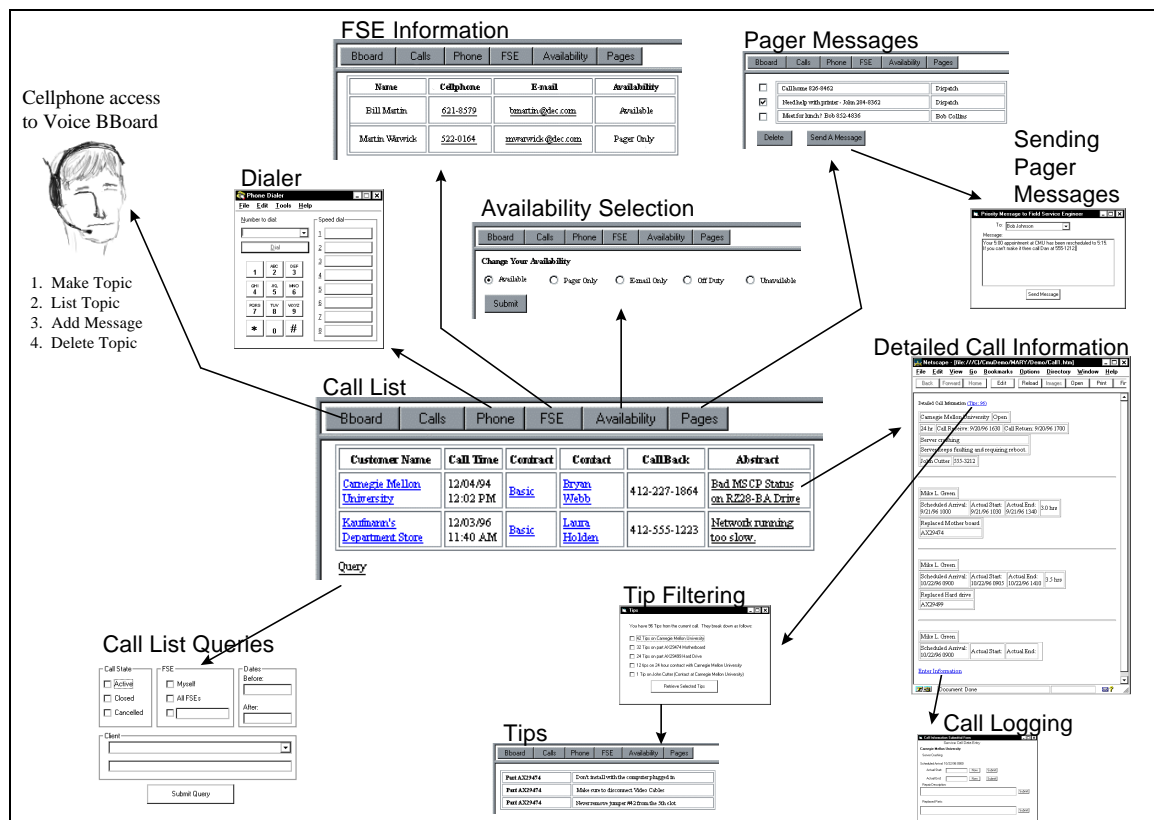
The capability for the Master/Apprentice paradigm is in the synchronous communication bubble in Figure 4. Synchronous communications facilitates answering questions such as: "Where is the object?" (drawing on captured image), "How to do it?" (audio guidance through prestored material), and "What does the result of the test mean?" (audio discussion). This model also uses the Static/Pre-Stored capability.

4.3 Team Collaboration

The Mobile Communication and Computing Architecture (MoCCA) [9] is designed to support a group of geographically distributed field service engineers (FSE). The FSEs spend up to 30% to 40% of their time in a car driving to customer sites. Half of what they service is third party equipment for which they may not have written documentation. The challenge was to provide a system that allowed the FSE's to access information and advice from other FSEs while on customer sites and while commuting between sites. Synchronous and asynchronous collaboration (Figure 4) are supported for both voice and digitized information.

An additional challenge arose from user interviews which suggested that the FSEs desired all of the functionality of a laptop computer including a larger color display with an operational cycle of at least eight hours. The system had to be very light, preferably

less than one pound, and required access to several Legacy databases. Further discussions with the FSEs indicated that the most frequently used databases were textually-oriented. Only on rare occasions is access to graphical databases required. A novel architecture combined a light weight alphanumeric satellite computer with the high functionality of a base unit included in the FSEs tool kit. The base unit can be carried into any customer site providing instant access to the global infrastructure.

The Team Problem Solving interface Asynchronous capability (Figure 4) includes Audio Bulletin Boards and tips for shared collaboration space between remote field service engineer and their colleagues. The concept of an Audio Bulletin Board is equivalent to a storehouse for audio clips describing the problems that the FSEs encounter while on the job. Each "trouble" topic contains a list of audio responses from other FSEs with the possible solution. Figure 6 shows the integrated user interface which starts with the Call List, List of available FSEs, and information about the incoming request for service.

Figure 6. MoCCA Integrated Interface

4.4 Context Aware Collaboration – Proactive Synthetic Assistant

A context aware cell phone has been designed which provides the remote caller feedback on the current context of the person being called. Time (e.g., calendar), location, and

audio environment sensing/interpretation are used to derive user context. In this scenario, location is derived from a GPS unit on the car (some newer cell phones from Erickson and Motorola also have GPS). We have focused on the callee being the driver of a car. The goal is for the caller to interact with the driver in a manner similar to that of a passenger in the car. For example, when there is particularly difficult driving situation that has a high cognitive load (e.g. passing a truck on a down hill curve at night in the rain) a passenger is sensitive to the situation and suspend their conversation until the driving situation has passed. With contemporary cell phones, however, the caller is unaware of the driver's context and will continue talking, perhaps causing the driver to enter a state of cognitive overload.

5 Evaluation

Evaluation of all four Applications was performed with laboratory prototypes. Metrics used in this evaluation include time on task and time/accuracy. These tests can be used as examples of how to evaluate wearable computer systems.

5.1 Pre-Stored Procedures

Field tests were performed at Digital Equipment Corporation facility in Forrest Hills, Pittsburgh, PA. Five Field Service Engineers (FSEs) participated in tests that included performing a set of typical operations related to troubleshooting and repair operations on computing equipment. Each of the FSEs performed all of these operations. The subject systems included printers, motherboards, and networks. The use of the prototype contributed to a significant saving of time (35 to 40 %), Figure 7. During these field tests, the FSEs used the system for the first time. A larger savings in time is expected with continued use. In addition, MoCCA allowed the FSEs to fix some problems immediately, which otherwise would have required return trips to find and bring back manuals.
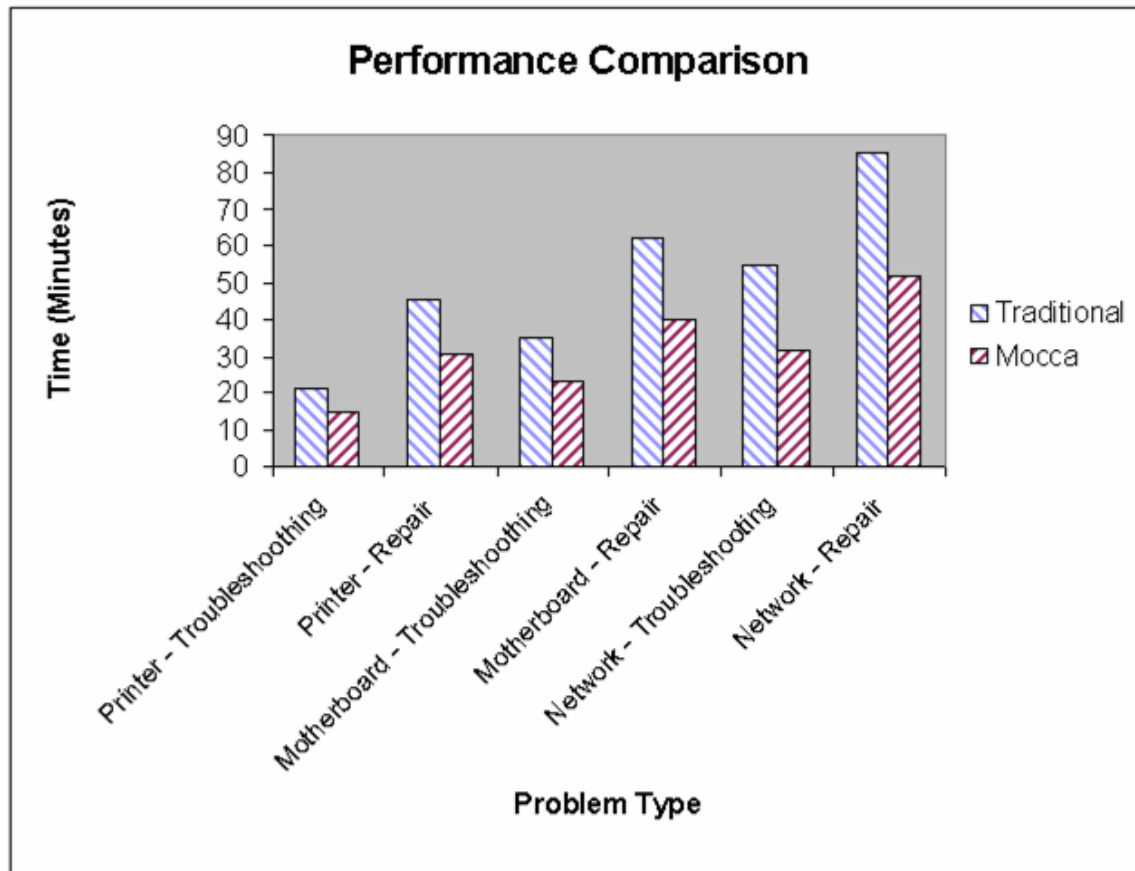
Figure 7. Improvement in problem solving with pre-stored knowledge versus traditional approach

5.2 Master/Apprentice Help Desk

Performance on a bicycle repair task when working alone was compared to working with a helper who could provide guidance through the repair process. There was a video link between the worker and expert, as well as an audio connection. Participants consisted of 60 CMU students (69% male), and two bicycle repair experts. Workers do substantially better at performing these repair tasks with collaborative help. A repeated measures analysis of variance (ANOVA) test [14] was used to examine statistical significance of the results. In a bicycle repair experiment, average time to complete the tasks with a remote expert was half as long as in the solo condition (7.5 versus 16.5 minutes respectively; $p < .001$). This means that the probability that the two average times are the same is less than 0.1%. The quality of the repairs they completed was superior when they had assistance than when they worked alone (79% of the quality points for the collaborative condition versus 51% for the solo condition; $p < .001$). While having access to an expert dramatically improved performance, having better tools for communication with the expert did not improve the number of tasks completed, the average time per completed task, or performance quality. In particular, neither video (comparison of full duplex audio/video with full duplex audio/no video) nor full duplex audio (comparison of full duplex audio/video condition with half duplex audio/video)

helped workers perform more tasks, perform tasks more quickly, or perform them better [6].

5.3 Team Collaboration

Idealink provides a virtual space for groups to manipulate graphical objects related to their work task, sharing observations with each other [15]. Idealink is well-suited for use in wearable computers because it enables users to have adhoc meetings and design sessions, providing them with a canvas on which they can sketch their ideas for a remote user. Asynchronous audio tags enable users to record an audio explanation or annotation of a particular object or procedure. Each session is recorded and archived for later playback, making the knowledge contained within them available for later reference.

An experiment was designed to explore the effectiveness of communicating concepts with Idealink versus a traditional whiteboard. A group problem solving task to design a remote control for a stereo system was performed. Eight groups of four students (from different majors) were participating. Four groups used Idealink and the other four used a standard whiteboard. All sessions were videotaped. The recordings were examined to determine the frequency of requests for clarification as summarized in Table 4.

Collaboration Event	Whiteboard	Idealink
Explicit communication	54	33
Errors and difficulties	3	0
Implicit communication	40	7
Errors and difficulties	2	0

Table 4. Idealink User Evaluation

Two of the observed event types were: 1) Explicit communication had explicit verbal references to a particular object or region of the drawing area (i.e. "Look at the box on the left"); and 2) Implicit communication, using an indirect means of referring to an object or region of the drawing area (i.e. "Look at that one"). Errors occurred when a team member misunderstood a reference made by another team member. Idealink not only reduced the number of communications but also reduced the number of communication errors by providing smaller regions in which collaborators can more easily focus their attention.

5.4 Context Aware Collaboration – Proactive Synthetic Assistant

Two experiments tested the hypothesis that a context-aware cell phone could change caller and driver behaviors. Experiment 1 tested whether remote cell-phone callers would slow or stop their conversation with a driver when signaled. Experiment 2 tested whether a driver's performance while speaking on a cell phone would be improved by slowing or stopping the remote callers' conversation.

Participants (n = 24) were asked to role-play a person seeking to rent an apartment. Each participant made successive cell phone calls to three "landlords," played

by the experimenter. Participants were provided a list of questions to ask the landlord about each apartment (e.g., how many bedrooms the apartment had). At a prespecified point in each call, the landlord would unexpectedly pause for 10 seconds.

Results show that the callers said less than half the number of sentences during the pause when they were sent a signal compared to when the driver remained silent. The spoken message: "The person you have called is busy. Please hold." was the most effective signal.

Experiment 2 used a driving simulator composed from a virtual reality authoring environment and allowed users to navigate a vehicle through a test track. Before beginning the experiment, participants (n = 20) practiced using the driving simulator until they said they felt comfortable. Participants then completed one circuit of the track on the driving simulator under each of three conditions: control (no phone call), call without pause, and call with pause.

Results show that talking on the cell phone caused people to crash more (6.8 crashes) as compared to driving without a call (3.55 crashes). Inducing pauses during the call caused the driver to crash less (3.65 crashes) when using the cell phone.

6 Summary and Future Challenges

In this paper, we have introduced and described a taxonomy of problem solving capabilities for wearable and context aware computers. We have shown how these capabilities impact choices of input/output modalities and user interface models.

An important set of challenges must be addressed to make wearable computing effective with ubiquitous computing environments. How do we develop social and cognitive models of applications? How do we integrate input from multiple sensors and map them into user social and cognitive states? How do we anticipate user needs? How do we interact with the user?

In the future we will focus our efforts on development of a Virtual Coach, which will capture a wearable augmented cognition platform and software application, as well as be able to monitor an individual's cognitive load and route tasks to less loaded individuals. Cognitive performance will be assessed on-line. Providing immediate suggestions to a user for cognitive augmentation and arbitration of resource re-deployment will further enhance performance.

8 Acknowledgements

9 References

[1]   Smailagic, A. and Siewiorek, D. P. (1994). The CMU mobile computers: A new generation of computer systems. In Proceedings of the IEEE COMPCON 94, 467-473.

[2]   Smailagic, A., Siewiorek, D. P. et. al. (1995). Benchmarking an interdisciplinary concurrent design methodology for electronic / mechanical design. Proc. ACM / IEEE Design Automation Conference, 514-519.

[3]   Siewiorek, D.P., Smailagic, A., and Lee, J.C. (1994). An interdisciplinary concurrent design methodology as applied to the Navigator wearable computer system. Journal of Computer and Software Engineering, Ablex Publishing Corporation, 2(3), 259-292.

[4]   Smailagic, A., Siewiorek, D.P., Stivoric, J., Martin, R. (1998). Very rapid prototyping of wearable computers: A case study of custom versus off-the-shelf design methodologies. Journal on Design Automation for Embedded Systems, 3, 217-230.

[5]   Najjar, L., Thompson, J. C., Ockerman, J. J. (1997). A wearable computer for quality assurance in a food processing plant. Proc. The First International Symposium on Wearable Computers, IEEE Computer Society Press, 163-164.

[6]   Fussell, S. R., Kraut, R. E., Siegel J. (2000). Coordination of Communication: Effects of Shared Visual Context on Collaborative Work. Proceedings of the ACM Conference on Computer Supported Cooperative Work.

[7]   Bahl, V. (2000). Internal Document, Microsoft Research, Redmond, Washington.

[8]   Smailagic, A., Siewiorek, D. P., Bass, L., Iannucci, B., Dahbura, A., Eddleston, S., Hanson B. (1999) MoCCA: A mobile communication and computing architecture. Proc. IEEE International Symposium on Wearable Computers, 64-71.

[9]   Smailagic, A. (1998). An evaluation of audio - centric CMU wearable computers. ACM Journal on Special Topics in Mobile Networking, 6, 59-68.

[10]   Bauer, M., Heiber, T., Kortuem, G., Segall, Z. (1998). A collaborative wearable system with remote sensing. Proceedings of the Second International Symposium on Wearable Computers, IEEE Computer Society Press, 10-17.

[11]   http://www.fas.org/man/dod-101/sys/land/land-warrior.htm

[12]   Feiner, S., MacIntyre, B., Höllerer, T. (1997). A touring machine: prototyping 3D mobile augmented reality systems for exploring the urban environment. Proc. The First International Symposium on Wearable Computers, IEEE Computer Society Press, 74-81

[13]   Marinelli, D., Stevens, S.M. (1998) Synthetic Interviews: The Art of Creating a 'Dyad' Between Humans and Machine-based Characters, Proc. The 4th IEEE Workshop on Interactive Voice Technology for Telecommunications Applications, Torino, Italy.

[14]   Girden, E. R. (1992). ANOVA Repeated Measures. Sage Publications, Thousand Oaks, CA.

[15]   Garlan, D. Siewiorek, D. P., Smailagic A., Steenkiste, P. (2002). Project Aura: Toward Distraction-Free Pervasive Computing. IEEE Pervasive Computing, 1, 22-32.

[16]   Billinghurst, M., Weghorst, S., Furness III, T. (1997). Wearable computers for three-dimensional CSCW. Proc. The First International Symposium on Wearable Computers, IEEE Computer Society Press, 39-47.

[17]   Dey, A., Futakawa, M., Salber, D. Abowd, G. (1999). The conference assistant: combining context-awareness with wearable computing. <u>Proceedings of the 3rd International Symposium on Wearable Computers,</u>   IEEE   Computer   Society   Press,   21-28.

[18]   Healey, J., & Picard, R (1998). StartleCam: a cybernetic wearable camera. Proceedings of the Second International Symposium on Wearable Computers, IEEE Computer Society Press, 42-49

[19]   Rekimoto, J. (1996). Transvision: A hand-held augmented reality system for collaborative design. <u>Proceedings of Virtual Systems and Multimedia</u>.

[20]   Starner, T., Weaver, J., Pentland, A. (1997). A wearable computer based American sign language recognizer. <u>Proceedings of The Second IEEE International Symposium on Wearable Computers,</u> IEEE Computer Society Press, 130-137.