

HCI Book Chapter
Wearable Computers
Dan Siewiorek*, Asim Smailagic*, Thad Starner[†]
***Carnegie Mellon University**
[†]Georgia Institute of Technology

Introduction

Computers have become a primary tool for office workers, allowing them to access the information they need to perform their jobs; however, accessing information is more difficult for more mobile users. With current computer interfaces, the user must focus both physically and mentally on the computing device instead of the environs. In a mobile environment, such interfaces may interfere with the user's primary task. However, many mobile tasks could benefit from computer support. Our focus is the design of wearable computers that augment, instead of interfere, with the user's tasks. Carnegie Mellon University's VuMan 3 project provides an example of how the introduction of wearable computing to a task can reap many rewards.

VuMan 3 maintenance inspection wearable computer



Figure <VuMan 3 Savings>. VuMan 3 Savings Factors

Many maintenance activities begin with an inspection in which problems are identified. Job orders and repair instructions are generated from the results of the inspection. The VuMan 3 wearable computer was designed for streamlining Limited Technical Inspections (LTI) of amphibious tractors for the U.S. Marines at Camp Pendleton, California (Smailagic, 1998). The LTI is a 600-element, 50-page checklist that usually takes four to six hours to complete. The inspection includes an item for each part of the vehicle (for example, front left track, rear axle, windshield wipers, etc.). VuMan 3 created an electronic version of this checklist. The system's interface was arranged as a menu hierarchy and a physical dial and selection buttons controlled navigation. The top level consisted of a menu that gave a choice of function. Once the inspection function was chosen, the component being inspected was selected by its location on the vehicle. At each stage, the user could go up one level of the hierarchy.

The inspector selects one of four possible options about the status of the item: Serviceable, Unserviceable, Missing, or On Equipment Repair Order (ERO). Further explanatory comments about the item can be selected (for example, the part is unserviceable due to four missing bolts).

The LTI checklist consists of a number of sections, with approximately one hundred items in each section. The user sequences through each item by using the dial to select "next item," or "next field." A "smart cursor" helps automate some of the navigation by positioning the user at the next most likely action.

As part of the design process, a field study was performed. In typical trouble-shooting tasks one Marine would read the maintenance manual to a second Marine who performs the inspection. With the VuMan 3, only one Marine is needed for the task as he has the electronic maintenance manual with him. Thus, the physical manual does not have to be carried into hard-to-reach places.

The most unanticipated result was a 40% reduction in inspection time. The bottom right image of Figure \ref{fig:vuman3-savings} demonstrates the reason for this result. Here, the marine is on his side looking up at the bottom of the amphibious tractor. In such places it is hard to read or write on the clipboard typically used for inspections. The Marine constantly gets into position, crawls out to read instructions, crawls back into position for the inspection, and then crawls out again to record the results. In addition, the marine tends to do one task at a time when he might have five things he has to inspect in one place. This extra motion has a major impact on the time required to do a task. By making information truly portable, wearable computers can improve the efficiency of this application and many other similar ones.

The second form of time savings with the VuMan 3 occurred when the inspection is finished. The wearable computer requires a couple of minutes to upload its data to the logistics computer. The manual process, however, required a typist to enter the Marine's handwritten text into the computer. Given that the soldier may have written his notes in

cold weather while wearing glove, the writing may require some interpretation. This manual process represents another 30% of the time.

Such redundant data entry is common when users are mobile (Starner, 2004). There are numerous checklist-based applications including plant operations, pre-flight checkout of aircraft, inventory, etc. that may benefit from a form-filling application run on a wearable computer. In the case of the VuMan 3 project, the results were striking. From the time the inspection was started until the data was entered into the logistics computer, 70% of the time was saved by using the wearable. There was a potential savings by reducing maintenance crews from two to one. Finally, there was also a savings in weight over paper manuals.

The Wearable Computing CAMP

Designing wearable computer interfaces requires attention to many different factors due to their closeness to the body and their use while performing other tasks. For the purposes of discussion, we have created the "CAMP" framework which consists of the following factors:

Corporal: Wearables should be designed to interface physically with the user without discomfort or distraction.

Attention: Interfaces should be designed for the user's divided attention between the physical and virtual worlds.

Manipulation: When mobile, users lose some of the dexterity assumed by desktop interfaces. Controls should be quick to find and simple to manipulate.

Perception: A user's ability to perceive displays, both visual and audio, is also reduced while mobile. Displays should be simple, distinct, and quick to navigate.

Power, heat, on-body and off-body networking, privacy, and many other factors also affect on-body computing (Starner 2001). Many of these topics are the subjects of current research, and much work will be required to examine how these factors inter-relate. Due to space, we will concentrate mainly on CAMP principles and practice in the remainder of this chapter.

Corporal: Design Guides for Wearability

The term "wearable" implies the use of the human body as a support environment for the object described. Society has historically evolved its tools and products into more portable, mobile, and wearable form factors. Clocks, radios, and telephones are examples of this trend. Computers are undergoing a similar evolution. Simply shrinking computing tools from the desktop paradigm to a more portable scale does not take advantage of a whole new context of use. While it is possible to miniaturize keyboards, human evolution has not kept pace by shrinking our fingers. There is no Moore's Law for humans. The human anatomy introduces minimal and maximal dimensions that define the shape of wearable objects, and the mobile context also defines dynamic interactions. Conventional

methods of interaction, including keyboard, mouse, joystick, and monitor, have mostly assumed a fixed physical relationship between user and device. With wearable computers, the user's physical context may be constantly changing. Symbol's development of a wearable computer for shipping hubs provides an example of how computing must be adapted for the human body.

As a company, Symbol is well-known for its barcode technology. However, it is also one of the first successful wearable computer companies, having sold over 100,000 units from its WSS 1000 line of wearable computers (see Figure <Symbol>). The WS-1000 consists of a wrist-mounted wearable computer that features a laser barcode scanner encapsulated in a ring worn on the user's finger. This configuration allows the user to scan barcodes while keeping both hands free to manipulate the item being scanned. Because the user no longer has to fumble with a desk-tethered scanner, these devices increase the speed at which the user can manipulate packages and decrease the overall strain on the user's body. Such features are important in shipping hubs, where millions of packages are scanned by hand every year. Symbol spent over US \$5 million and devoted 40,000 hours of testing to develop this new class of device, and one of the major challenges was adapting the computer technology to the needs of the human body (Stein, 1998).



Figure <Symbol>. Symbol's WSS 1000 series wrist-mounted wearable computer with ring scanner.

One of the first observations made was that users may be widely varying shapes and sizes. Specifically, Symbol's scanner had to fit the fingers of both large men and small women. Similarly, the wrist unit had to be mounted on both large and small wrists. Even though the system's wires were designed to be unobtrusive, the system must be designed to break away if entangled and subjected to strain. This policy provided a safeguard for the user.

Initial testing discovered other needs that were obvious in hindsight. For example, the system was strapped to the user's forearm while the user exerted himself moving boxes. Soon, the "soft-good" materials, which were designed for the comfort of the user, became sodden with sweat. After one shift, the user was expected to pass the computer to

the operator on the next shift. Not only was the sweat laden computer mount considered “gross,” it also presented a possible health risk. This problem was solved by separating the computer mount from the computer itself. Each user received his own mount, which he could keep adjusted to his own needs. After each shift, the computer could be removed from the user's mount and placed in the replacement user's mount.

Another unexpected discovery is that the users tended to use the computer as body armor. When a shipping box would begin to fall on the user, the user would block the box with the computer mounted on his forearm as that was the least sensitive part of his body. Symbols designers were surprised to see users adapt their work practices to use the rigid forearm computer to force boxes into position. Accordingly, the computer's case was designed out of high impact materials. However, another surprise came with longer term testing of the computer.

Employees in test company's shipping hubs constantly reached into wooden crates to remove boxes. As they reached into the crates, the computer would grind along the side. After extended use, holes would appear in the computer's casing, eventually damaging the circuitry. Changing the composition of the casing to be resistant to both abrasion and impact finally fixed the problem.

After several design cycles, Symbol presented the finished system to new employees in a shipping hub. After a couple of weeks' work, test results showed that the new employees felt the system was cumbersome, whereas established employees who had participated in the design of the project felt that the wearable computer provided a considerable improvement over the old system of package scanning. After consideration, Symbol's engineers realized that these new employees had no experiential basis for comparing the new system to the past requirements of the job. As employees in shipping hubs are often short-term, a new group of employees were recruited. For two weeks, these employees were taught their job using the old system of package scanning: the employee would reach into a crate, grasp a package, transfer it to a table, grasp a handheld scanner, scan the package, replace the scanner, grasp the package, and transfer it to its appropriate conveyer belt. The employees were then introduced to the forearm-mounted WS-1000. With the wearable computer, the employee would squeeze his index and middle finger together to trigger the ring-mounted scanner to scan the package while reaching for it, grasp the package, and transfer it to the appropriate convey belt in one fluid motion. These employees returned very positive scores for the wearable computer.

This lesson, that perceived value and comfort of a wearable computer is relative, was also investigated by Bodine and Gemperle (2003). In short interviews, users were fitted with a backpack or armband “wearable” and told that the system was either a police monitoring device (similar to those used for house arrest), a medical device for monitoring health, or a device for use during parties. The subjects were then asked to rate the devices on various scales of desirability and comfort. Not surprisingly, the police “wearable” was considered the least desirable. However, the police function elicited more negative *physical* comfort ratings, and the medical function elicited more positive physical

comfort ratings even though they were the same device. In other words, perceived comfort can be affected by the supposed function of the device.

Researchers have also explored wearability in more general terms. Wearability is defined as the interaction between the human body and the wearable object. Dynamic wearability includes the human body in motion. Design for wearability considers the physical shape of objects and their active relationship with the human form. Gemperle *et al.* explored history and cultures including topics such as clothing, costumes, protective wearables, and carried devices (Gemperle, 1998, Siewiorek, 2002). They studied physiology, biomechanics, and the movements of modern dancers and athletes. Drawing upon the experience of CMU's wearables group over two dozen generations of machines representing over a hundred person years of research, they codified the results into guidelines for designing wearable systems. These results are summarized in Table 1.

Attribute	Comments
Placement	Identify where the computer should be placed on the body. Issues include identifying areas of similar size across a population, areas of low movement/flexibility, and areas large in surface area
Humanistic Form Language	The form of the object should work with the dynamic human form to ensure a comfortable fit. Principles include inside surface concave to fit body, outside surface convex to deflect objects, tapering sides to stabilize form on body, and radiusing edges/corners to provide soft form.
Human Movement	Many elements make up a single human movement: mechanics of joints, shifting of flesh, and the flexing and extending of muscles and tendons beneath the skin. Allowing for freedom of movement can be accomplished in one of two ways: by designing around the more active areas of the joints or by creating spaces on the wearable form into which the body can move.
Human Perception of Size	The brain perceives an aura around the body. Forms should stay within the wearer's intimate space, so that perceptually they become a part of the body. The intimate space is between zero and five inches off the body and varies with position on the body.
Size Variations	Wearables must be designed to fit many types of users. Allowing for size variations is achieved in two ways: static anthropometric data, which details point to point distances on different sized bodies and consideration of human muscle and fat growth in three dimensions using solid rigid areas coupled with flexible areas.
Attachment	Comfortable attachment of forms can be created by wrapping the form around the body, rather than using single point fastening systems such as clips or shoulder straps.
Contents	The system must have sufficient volume to house electronics, batteries, etc. that, in turn, constrains the outer form.

Weight	The weight of a wearable should not hinder the body's movement or balance. The bulk of the wearable object weight should be close to the center of gravity of the human body minimizing the weight that spreads to the extremities.
Accessibility	Before purchasing a wearable system, walk and move with the wearable object to test its comfort and accessibility.
Interaction	Passive and active sensory interaction with the wearable should be simple and intuitive.
Thermal	The body needs to breathe and is very sensitive to products that create, focus, or trap heat.
Aesthetics	Culture and context will dictate shapes, materials, textures, and colors that perceptually fit the user and their environment.

Table 1. Design for Wearability attributes.

This team also developed a set of wearable forms to demonstrate how wearable computers might be mounted on the body. Each of the forms was developed by applying design guidelines and follows a simple pattern for ensuring wearability. The pods were designed to house electronic components. All of the forms are between 3/8" and 1" thick, and flexible circuits can fit comfortably into the 1/4" thick flex zones. Beginning with acceptable areas and the humanistic form language, the team considered human movement in each individual area. Each area is unique, and some study of the muscle and bone structure was required along with common movements. Perception of size was studied for each individual area. For testing, minimal amounts of spandex was stretched around the body to attach the forms. The results are shown in Figure [\ref{fig:wearableforms}](#).



Figure <WearableForms>. Forms studied for wearability.

These studies and guidelines provide a starting point for wearable systems designers. However, there is much work to be done in this area. Weight was not considered in these studies, nor were the long-term physiological effects such systems might have on the wearer's body. Similarly, fashion can affect the perception of comfort and desirability of a wearable component. As wearable systems become more common and are used for longer periods of time, it will be important to test these components of wearability.

Attention

Humans have a finite and non-increasing capacity that limits the number of concurrent activities they can perform. Herb Simon observed that human effectiveness is reduced as they try to multiplex more activities. Frequent interruptions require a refocusing of attention. After each refocus of attention a period of time is required to re-establish the context prior to the interruption. In addition human short term memory can hold seven plus or minus two (i.e. five to nine) chunks of information. With this limited capacity, today's systems can overwhelm users with data, leading to information overload. The challenge to human computer interaction design is to use advances in technology to preserve human attention and to avoid information saturation.

In the mobile context, the user's attention is divided between the computing task and the activities in the physical environs. Some interfaces, like some augmented realities (Azuma, 1997) and Dual Purpose Speech (Lyons, 2004), try to integrate the computing task with the user's behavior in the physical world. The VuMan 3 interface did not tightly couple the virtual and real worlds, but the computer interface was designed

specifically for the user's task and allowed the user to switch rapidly between a virtual interface and his hands-on vehicle inspection.

However, many office productivity tasks, such as e-mail or web searching, have little relation to the user's environment. The mobile user must continually assess what attentional resources he can commit to the interface and for how long before switching attention back to his primary task. Oulasvirta *et al* specifically examine such situations (Oulasvirta, 2005) by fitting cameras to mobile phones and observing users attempting web search tasks while following pre-described routes. Subjects performed these tasks in a laboratory, in a subway car, riding a bus, waiting at a subway station, walking on a quiet street, riding an escalator, eating at a cafeteria and conversing, and navigating a busy street. Web pages required an average of 16.2 seconds to load and had considerable variance requiring the user to attend the interface. The subjects shifted their attention from the phone interface more often depending on the task: 35% of page loadings in the laboratory versus 80% of the page loadings while walking a quiet street. The duration of continuous attention on the mobile device also varied depending on the physical environment: 8-16 seconds for the laboratory and cafe versus below 6 seconds for the riding the escalator or navigating a busy street. Similarly, the number of attention switches depending on the demands of the environment.

The authors note that even riding an escalator requires demands on attention (e.g. choosing a correct standing position, monitoring personal space for passers-by, and determining when the end is in order to step off). Accordingly, they are working on a "Resource Competition Framework," based on the Multiple Resource Theory of attention (Wickens, 2000), to relate mobile task demands to the user's cognitive resources. This framework helps predict when the mobile user will need to adopt attentional strategies to cope with the demands of a mobile task. The authors report four such strategies that were observed in their study. The first, *calibrating attention* refers to the process where the mobile user first attends to the environment and determines the amount of attention he needs to devote to the environment versus the interface. *Brief sampling over long intervals* refers to the practice of only attending to the environment in occasional brief bursts to monitor for changes that may require a deviation from plan, such as when reading while walking a empty street. *Task finalization* refers to subjects' preference to finish, when sufficiently close, a task or subtask before switching attention back to the physical environment. *Turntaking capture* occurs when the user is conversing with another person. Attending and responding to another person requires significant concentration, leading to minimal or no attention to the mobile interface.

The third author, who has been using his wearable computer to take notes on his everyday life since 1993, has remarked on similar strategies in his interactions. Describing these attentional strategies and designing interfaces that leverage them will be important in future mobile interfaces. Much research has been performed on aircraft and automobile cockpit design to design interfaces that augment but do not interfere with the pilot's primary task of navigating the vehicle. However, only recently has it begun to be possible to instrument mobile users and examine interface use (and misuse) "in-the-field"

for the mobile computer user. Now, theories of attention can be applied and tested to everyday life situations.

This newfound ability to monitor mobile workers may help us determine how **not** to design interfaces. In contrast to the VuMan 3 success described above, Ockerman's PhD thesis "Task Guidance and Procedure Context: Aiding Workers in Appropriate Procedure Following" warns that mobile interfaces, if not properly designed, may hinder the user's primary task (Ockerman, 2000). Ockerman studied experienced pilots inspecting their small aircraft before flying. When a wearable computer was introduced as an aid to completing the aircraft's safety inspection checklist, the expert pilots touched the aircraft less (a way many pilots develop an intuition as to the aircraft's condition). In addition, the pilots relied too much on the wearable computer system, which was purposely designed to neglect certain safety steps. The pilots trusted the wearable computer checklist to be complete instead of relying on their own mental checklists. Ockerman shows how such interfaces might be improved by providing more context for each step in the procedure. Another approach would be integrating the aircraft itself into the interface (for example, use augmented reality to overlay graphics on the aircraft indicating where the pilot physically inspects the plane).

Most recently, DARPA's Augmented Cognition project (Kollmorgen, Schmorow 2005) aims to create mobile systems that monitor their user's attentional resources and records or delays incoming information in order to present it to the user in a more orderly and digestible time sequence. These systems exploit mobile electroencephalogram (EEG) readings or functional near infrared imaging (fNIR) to monitor the user's brain activations and relate these results to the user's current state (Archinoetics). Such projects, if successful on a larger scale, could reveal much about the mental resources required for truly mobile computing (Archinoetics cite).

The Attention Matrix, shown in Figure <AttentionMatrix>, (Anhalt, Smailagic, Siewiorek, 2001) categorizes activities by the amount of attention they require. The activities are Information, Communication, and Creation. Individual activities are categorized by the amount of distraction they introduce in units of increasing time: Snap, Pause, Tangent, and Extended. The Snap duration is an activity that is usually completed in a few seconds, such as checking your watch for the time. The user should not have to interrupt their primary activity to perform this activity. The Pause action requires the user to stop their current activity, switch to the new but related activity, and then return to their previous task within a few minutes. Pulling over to the side of the road and checking directions is an example of a pause. A Tangent action is a medium length task that is unrelated to the action that the user is engaged in. Receiving an unrelated phone call is an example of a tangent activity. An Extended action is when the user deliberately switches their task, beginning a wholly new long-term activity. For the car driver, stopping at a motel and resting for the night is an extended activity.

As distractions on the left of the matrix take less time from the user's primary activity, our intent is to move activities of the matrix towards the left side (Snap). Our goal is to evaluate how this process extends to a larger sample of applications.

Time →

	Snap	Pause	Tangent	Extended
Information				
active	<ul style="list-style-type: none"> Receiving Notifying Monitoring Serendipity 	<ul style="list-style-type: none"> Message arrival Information accessible Audion Stocks, Sports, Matching similar needs Free food 		<ul style="list-style-type: none"> Audio, Walkman Transferring files from network Reading news
passive	<ul style="list-style-type: none"> Seeking Browsing Finding Verifying 	<ul style="list-style-type: none"> Line length Bus arrival Locate person 	<ul style="list-style-type: none"> Exam calendar Software/hardware help Calendaring Navigation Information on web or built environment Recall previous queries Double checking information 	<ul style="list-style-type: none"> Looking for Class Notes Who else is doing this now? Access personal data Poster, bulletin board information Web Research Reviewing Class Notes
Communication				
artificial	<ul style="list-style-type: none"> Initiating Participating 	<ul style="list-style-type: none"> S.O.S. Emergency Instant messaging 	<ul style="list-style-type: none"> Introductions Queries 	<ul style="list-style-type: none"> Team building Collaborative work Event planning Assassins game Social Planning
informal	<ul style="list-style-type: none"> Broadcasting 		<ul style="list-style-type: none"> Information exchange Scheduling 	<ul style="list-style-type: none"> Posting information to bulletin board Advertising
formal	<ul style="list-style-type: none"> One to One communications with an individual One to Group communications with select group, team or family One to All Possible broadcast communications with unknown people 			
Creation				
Work	<ul style="list-style-type: none"> Recording Synthesizing Generating 	<ul style="list-style-type: none"> Remember this! Add a todo or call list 	<ul style="list-style-type: none"> Forwarding x to y 	<ul style="list-style-type: none"> Class note taking Meeting Filling out survey Registration New ideas Adding information to existing projects
				<ul style="list-style-type: none"> Generating messages Summarizing lecture Mobile tool building

Figure <AttentionMatrix>. Attention Matrix

Manipulation

VuMan 3 Dials Pointing

VuMan 3 added a novel manipulation interface suitable for use when physical attention is occupied. The VuMan3 has a low resolution display and, consequently, a purely textual interface. **Figure <VuMan 3 options screen>** shows a sample screen from the user interface. The user navigates through a geographically organized hierarchy: top, bottom, front, rear; then left, right, and more detail. Eventually, at the node leafs, individual components are identified. There are over 600 of these components. Each component is indicated to be “serviceable” or “unserviceable”. If it is serviceable then no further information is given. If it is unserviceable then one of a small list of reasons is the next screen.

The user can return up the hierarchy by choosing the category name in the upper right corner, or sequence to the next selection in an ordering of the components. Once a component is marked as serviceable or unserviceable, the next selection in the sequence is automatically

displayed for the user. Furthermore, each component has a probability associated with it of being serviceable and the cursor is positioned over the most likely response for that component.

The screen contained navigational information. Sometimes there is more on a logical screen than can fit on a physical screen. Screen navigation icons are on the left hand side of the screen. The user can go to the previous physical screen or next physical screen that are functional parts of the logical screen. The user can always go back to the main menu. In **Figure <VuMan 3 Information>** the options include the vehicle number, number of hours on the vehicle, vehicle serial number, etc., which is used to distinguish this report apart from other reports. The inspection is divided into sections and different people can be inspecting different sections in parallel. The inspector would pick a section, highlight it by rotating the dial, and then select the highlighted item by pressing a button. The inspector would then receive a detailed set of instructions on what to do. In **Figure <VuMan 3 hull forward>** the inspector is instructed to check for damage and bare metal. The "smart cursor" anticipates that the inspector will be filling in the "status field" whose current value is "none". By clicking, a list of options is displayed, the first of which is "serviceable". With the Marine LTI, the item is serviceable in 80% of the cases. By ordering the most probable selection first, the interface emulates a paper checklist where most of the items will be checked as "OK." The smart cursor then assumes the most likely step. There is no need to even move the dial - you merely need to click on the highlighted option. For example, in **Figure <VuMan 3 status>** "serviceable" has been filled in and the box signifying the next activity is "next." If all entries are "serviceable," one would simply tap the button multiple times. If an item is "unserviceable," the dial is turned an "unserviceable" is selected. Next, a list of reasons why that particular device was unserviceable would appear. The dial is rotated and one or more of the options are selected. Since more than one reason may be selected as to why it's unserviceable, 'done' is selected to indicate completion. The selected items would appear in the 'comment' field. When the check list is completed, the data is uploaded to the logistics computer, which would then generate the job work orders.

Several lessons were derived from building the system. As part of the design cycle, a mouse (essentially a disk with buttons) was tested. However, the physical configuration of the device could be ambiguous. Was the left button in the proper position when the mouse's tail was towards the user or away from the user? Were the buttons supposed to be at the top? The dial removed this orientation ambiguous.

Another design lesson was to minimize cables. An earlier system had a cable connecting the battery, a cable for the mouse, and a cable for the display. These wires quickly became knotted. To avoid this problem, the VuMan 3 design used internal batteries, and the dial was built into the housing. There only remaining wire was the one to the display.

A third lesson was that wearable computers have a minimum footprint that is comfortable for your hand. While the keyboards of palmtop computers are getting smaller, evolution has not correspondingly shrunk our fingers. The thickness of the electronics will become thinner. Eventually it will be as thick as a sheet of plastic or incorporated into clothing.

However there will be a minimal footprint for the interface. Furthermore, the interface - no matter where it is located on your body - is operated in the same way. This is a major feature of the dial. It can be worn on your hip or in the small of the back. In airplane manufacturing, where workers navigate small spaces, the hip defines the smallest diameter through which the person can enter. Here a shoulder holster is preferred for the wearable computer.

The Marines' oversized coverall pockets were an advantage for the system. The soldiers could drop the computer into their coveralls and operate it through the cloth of the pocket. In terms of simplicity, as well as orientation independence, the dial integrated with the presentation of information on the screen. Everything on the screen could be considered to be on a circular list. In most cases there are less than a dozen items on a screen that are selectable. This sparse screen is an advantage on a head mounted display where the user may be reading while are moving. The font must be large enough to read while the screen is bouncing. The dial should be an intuitive interface for web browsing. Probably there are less than a half dozen items on a typical page to select, and it is rotated clockwise or counter clockwise. A button is then used to select the highlighted item. VuMan 3 had three types of buttons that all performed the same function. The buttons support left hand and right hand thumb-dominant as well as a central for finger-dominant users.

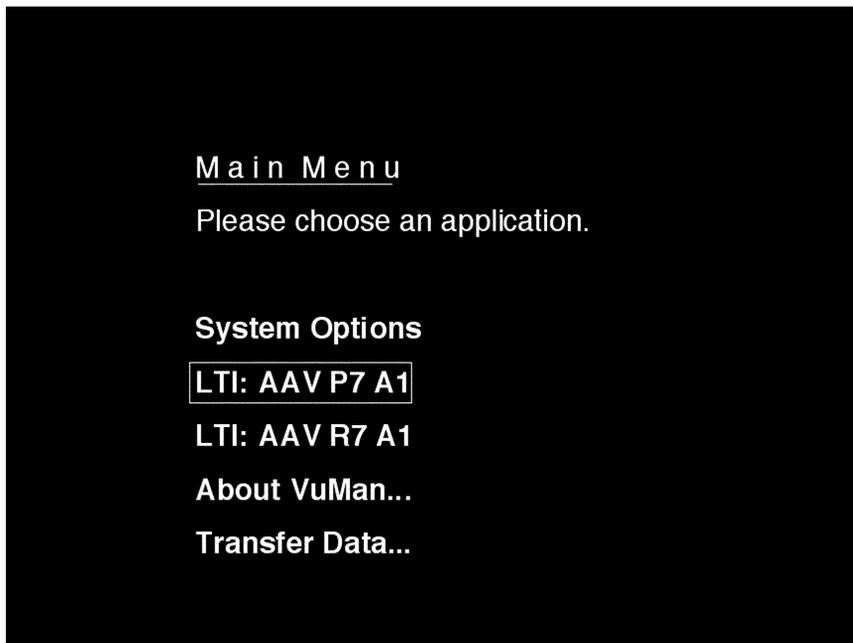


Figure <VuMan 3 options screen>. VuMan 3 Options Screen

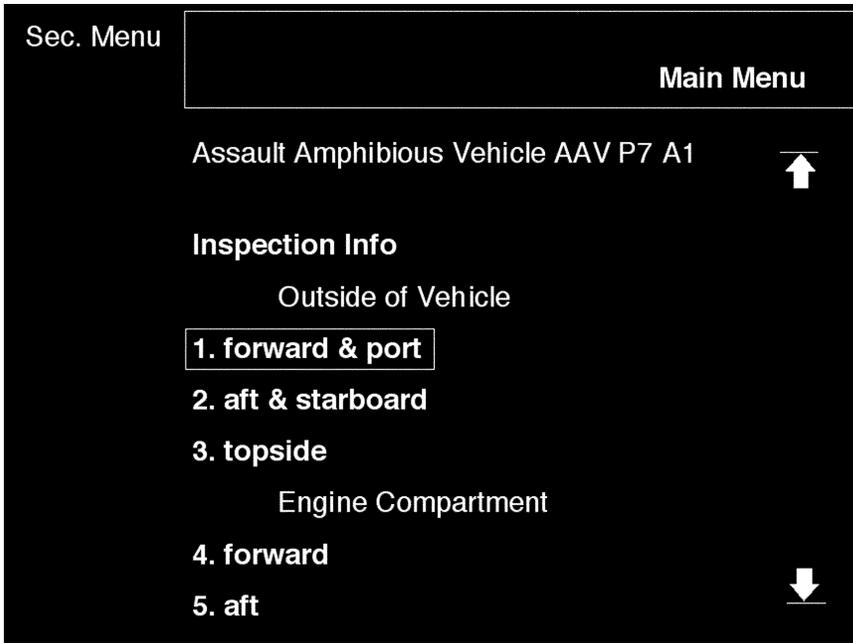


Figure <VuMan 3 Information>. VuMan 3 Information Screen

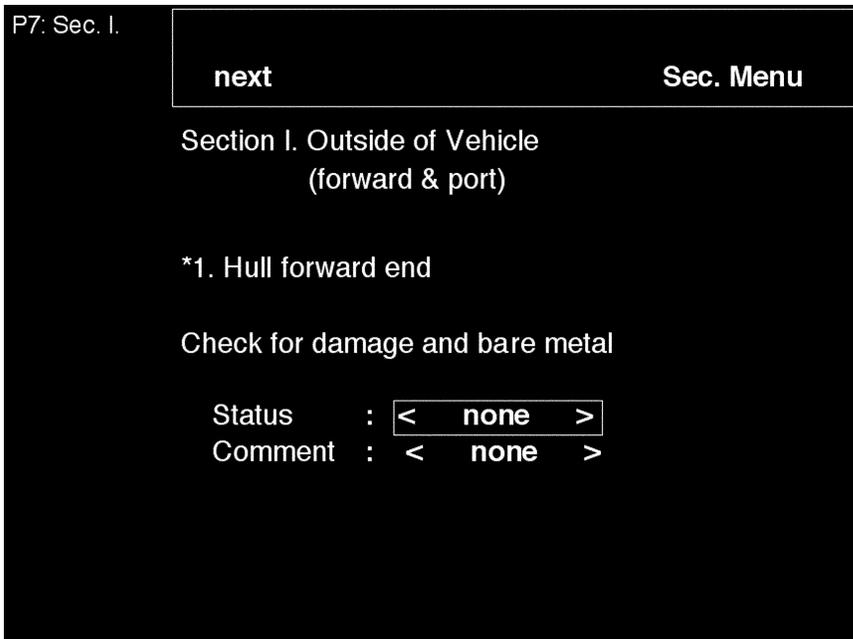


Figure <VuMan 3 hull forward>. VuMan 3 Hull Forward Screen

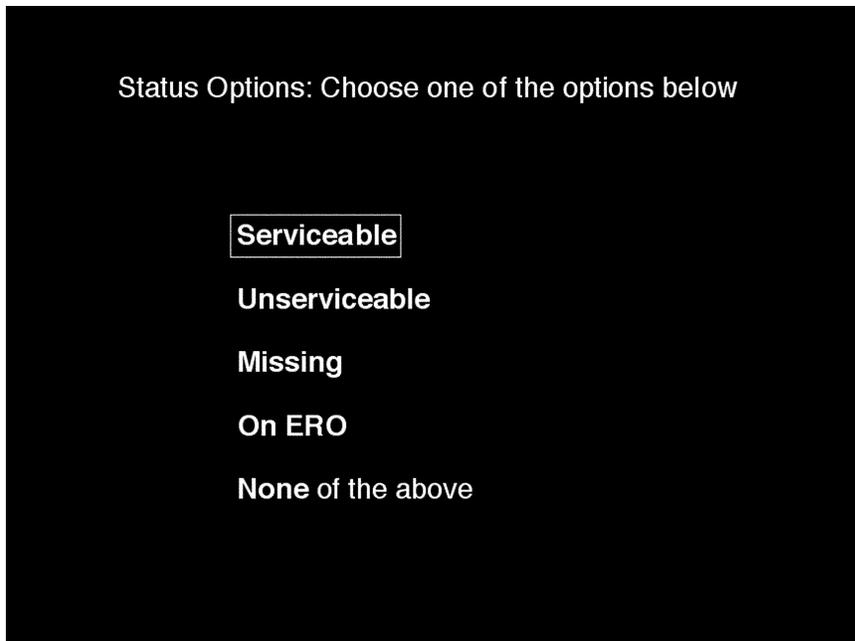


Figure <VuMan 3 status>. VuMan 3 Status Screen

Mobile Keyboards

The VuMan 3 addressed the problem of menu selection in the mobile domain and effectively used a 1D dial to create a pointing device that can be used in many different mobile domains. However, for tasks like wireless messaging, more free-form text entry is needed. While speech technology has made great strides in the last decade, speech recognition is very difficult in the mobile environment and often suffers from high error rates. In addition, speech is often not socially acceptable (in hospitals, meetings, classrooms, etc.). Keyboard interfaces still provide one of the most accurate and reliable methods of text entry.

Since 2000, wireless messaging has been creating billions of dollars of revenue for mobile phone service providers, and over 1 trillion messages are currently being typed per year. Until recently, many of these messages were created using the Multitap or T9 input method on phone keypads. Yet studies have shown that users average a slow 10-20 words per minute (wpm) using these common typing methods (for comparison, a highly skilled secretary on a desktop averages 70-90 wpm). Given the obvious desire for mobile text input, HCI researchers have begun re-examining keyboards. While keyboard entry has been well studied in the past, mobility suggests intriguing possibilities. For example, if an adequate method of typing can be combined with a sufficient display for the mobile market, computing may move "off-the-desktop" permanently.

Traditionally, text entry studies emphasize learnability, speed, and accuracy. However, a mobile user may not be able to devote all of his attention to the text entry process. For example, he may be taking notes on a conversation and wish to maintain eye contact with his conversational partner. Or, he may be in a meeting and may hide his keyboard under the desk to avoid distracting others with the keyboards noise and the motion of his

fingers. The user might also attempt to enter text while walking and need to attend his physical environment instead of looking at the screen. These conditions all describe “blind” typing where the user enters text with only occasional glances at the screen to ensure that the text has been entered correctly.

Lyons *et al.* and Clawson *et al.* have performed longitudinal studies on two keyboards, Handykey's Twiddler (Figure <twiddler>) and the mini-QWERTY “thumb” keyboard (Figure mini-QWERTY), to determine if they might achieve desktop level text entry in the mobile domain. As the “average” desktop entry rate was considered to be 30wpm, including hunt-and-peck typists, this benchmark was chosen as the minimum for speed. Traditionally, very high accuracy is desired for desktop typing. However, as a culture of informal e-mail and SMS messaging has developed, less accurate typing has become common. The community is debating how to reconcile speed and accuracy measures; however, error rates of approximately 5% per character are common in current mobile keyboard studies.



Figure <Twiddler>. Handykey's Twiddler



Figure <mini-QWERTY>. Mini-QWERTY Thumb Keyboard

With the Twiddler, novices averaged 4 wpm during the first twenty-minute session and averaged 47 wpm after twenty-five hours of practice (75 twenty-minute sessions). The fastest user averaged 67 wpm, which is approximately the speed of one of the authors who has been using the Twiddler for twelve years. While twenty-five hours of practice seems extreme, a normal high school typing class involves almost three times that training time to achieve a goal of 40 wpm.

Even so, mobile computer users may already have experience with desktop QWERTY keyboards. Due to their familiarity with the key layout, these users might more readily adopt a mini-QWERTY keyboard for mobile use. Can a mini-QWERTY keyboard achieve desktop rates? The study performed by Clawson *et al*, examined the speed and accuracy of experienced desktop typists on two different mini-QWERTY designs. These subjects averaged 30wpm during the first twenty-minute session and increased to 60wpm by the end of four hundred minutes of practice!

While both of these studies easily achieved desktop typing rates and had error rates comparable to past studies, can these keyboards be used while mobile? While neither study tested keyboard use while the user was walking or riding in a car, both experimented with blind text entry (in that, in at least one condition, typists could not look at the keyboard nor the output of their typing). When there was a statistically significant difference between blind and normal typing conditions, experienced Twiddler

typist slightly improved their speeds and decreased their error rates. However, experienced mini-QWERTY typists were significantly inhibited by the blind condition, with speeds of 46 wpm and approximately three times the error rate even after 100 minutes of practice. These results might be expected in that Twiddler users are trained to type without visual feedback from the keyboard whereas the mini-QWERTY keyboard design assumes that the user can see the keyboard to help disambiguate the horizontal rows of small keys.

The results of these studies demonstrate that there are multiple ways that desktop typing rates can be achieved on a mobile device. The question remains, however, whether the benefits of typing quickly while “blind” or moving will be sufficient to cause users to learn a new text entry method. Other benefits might also affect the adoption of keyboards in the future. For example, a 12 button device like the Twiddler can be the size of a small mobile phone and still perform well, while 40 button mini-QWERTY keyboards may have already shrunk as much as is possible for users' hands. Another factor may be adoption of mobile computing in developing countries. According to Techweb, almost 1 billion mobile phones were shipped in 2005. Many new mobile phone users will not have learned to type on a Roman alphabet keyboard and may be more concerned with quick learning than compatibility with desktop input skills.

Speech Interfaces

Vocollect

Mobile keyboards are not suitable for applications in which hands-free control is necessary, such as warehouse applications. Pittsburgh-based Vocollect focuses on package manipulation—in particular, the warehouse-picking problem. In this scenario, a customer places an order consisting of several different items stored in a supplier's warehouse. The order transmits from the warehouse's computer to an employee's wearable computer. In turn, each item and its location are spoken to the employee through a pair of headphones. The employee can control how this list is announced through feedback via speech recognition and can also report inventory errors as they occur. The employee accumulates the customer's order from the warehouse's shelves and ships it. This audio-only interface also frees the employee to manipulate packages with both hands, whereas a pen-based system would be considerably more awkward. As of December 2000, Vocollect had approximately 15,000 users and revenues between US \$10 and \$25 million.

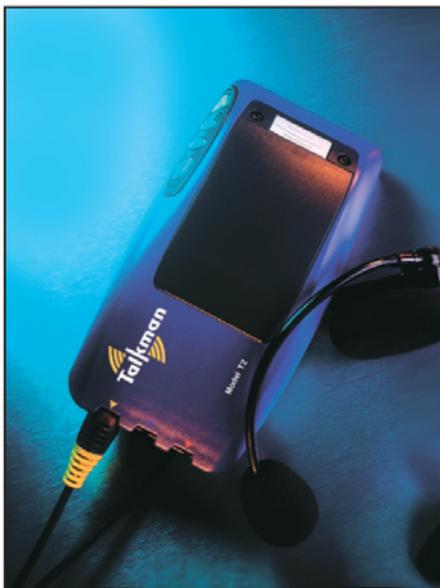


Figure <Vocollect Talkman>. Vocollect's audio-based wearable computer.

Navigator Wearable Computer with Speech Input

Boeing has been pioneering “augmented reality” using a head mounted, see-through display. As the user looks at the aircraft, the next manufacturing step is superimposed on the appropriate portion of the aircraft. One of their first applications is fabrication of wire harnesses. Every aircraft is essentially unique. They may be from different airlines. Even if they are from the same airline, one might be configured for a long haul route and another for a short haul route. The airline may specify different configurations. For example their galleys will be in different places, the wire harnesses would change, etc. Wire harnesses are fabricated months before they are assembled into the aircraft. The assembly worker starts with a peg board measuring about three feet high and six feet long. Mounted on the board is a full sized diagram of the final wire harness. Pegs provide support for the bundles of wire as they form. The worker selects a precut wire, reads its identification number, looks up the wire number on a paper list to find the starting coordinates of the wire, searches for the wire on the diagram, and threads the wire following the route on the diagram. With augmented reality, the worker selects a wire and reads the wire identification from the bar code. A head tracker provides the computer with information on where the worker is looking and superimposes the route for that particular wire on the board. Trial evaluations indicate a savings of 25% of the assembly effort primarily due to elimination of cross referencing the wire with paper lists.

The Navigator 2, circa 1995, is designed for a voice-controlled aircraft inspection application (Siewiorek 1994; Smailagic and Siewiorek 1996). The speech recognition system, with a secondary manually controlled cursor, offers complete control over the application in a hands-free manner, allowing the operator to perform an inspection with minimal interference from the wearable system. Entire, or portions, of aircraft manuals can be brought on-site as needed, using wireless communication. The results of inspection can be downloaded to a maintenance logistic computer.

Consider one portion of Navigator 2's application, 3-dimensional inspections. The application was developed for McClellan Air Force Base in Sacramento, California and the KC-135 aerial refueling tankers. Every five years these aircraft are stripped down to bare metal. The inspectors use magnifying glasses and pocket knives to hunt for corrosion and cracks.

At startup, **Figure <Navigator 2 Finite State>** the application prompts the user for either their choice of activating the speech recognition system or not. The user then proceeds to the Main Menu. From this location, several options are available, including online documentation, assistance, and the inspection task (**Figure < Navigator 2 Main Menu>**). Once the user chooses to begin an inspection, information about the inspection is entered, an aircraft type to examine is selected, and the field of interest is narrowed from major features (Left Wing, Right Tail, etc **Figure <Navigator 2 Region Selection>**) to more specific details (individual panes in the cockpit window glass, **Figure <Sheet metal>**). A coordinate system is superimposed on the inspection region. The horizontal coordinates begin from the nose and the vertical coordinates are “water lines” derived as if the airplane was floating. The inspector records each imperfection in the skin at the corresponding location on the display. The area covered by each defect as well as the type of defect, such as Corroded, Cracked, or Missing, are recorded. To maximize usability, each item or control may be selected simply by speaking its name. **Figure <Navigator 2>** shows the Navigator 2 systems in use.

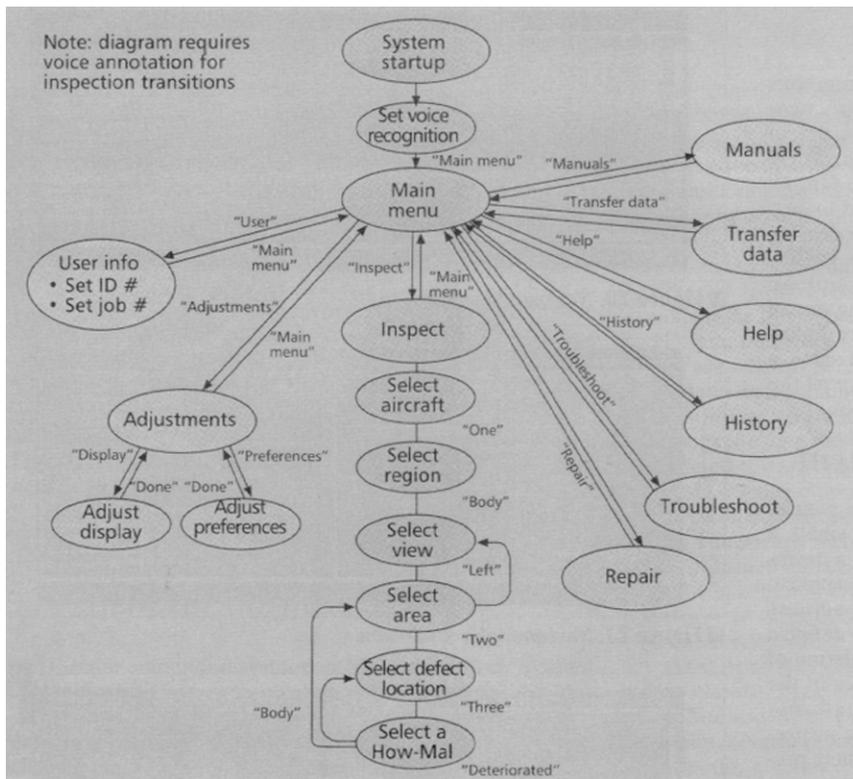


Figure <Navigator 2 Finite State>.

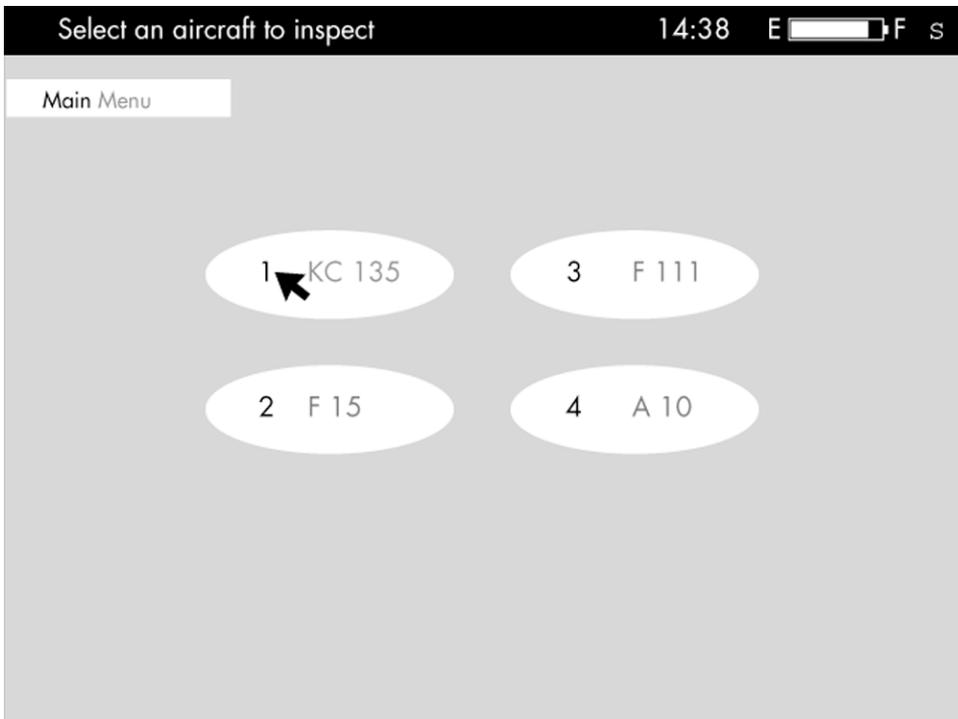


Figure < Navigator 2 Main Menu >

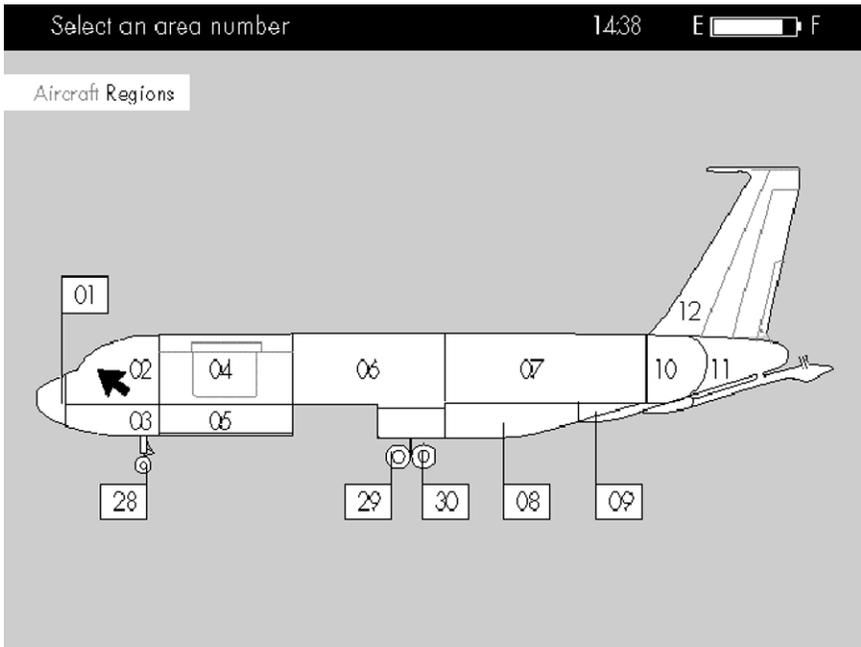


Figure <Navigator 2 Region Selection >

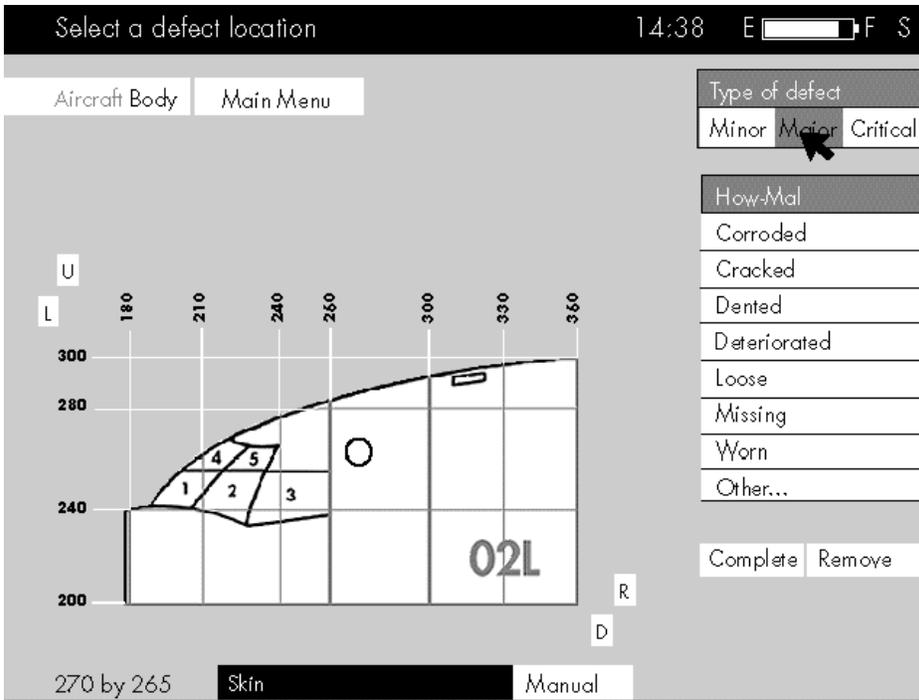


Figure <Sheet metal>.



Figure <Navigator 2>.

The user navigates to the display corresponding to the portion of the skin currently being inspected. This navigation is partially textual based on buttons (choose aircraft type to be inspected) and partially graphical based on side perspectives of the aircraft (choose area of aircraft currently being inspected). The navigation can be performed either through a joystick input device or through the use of speech input. The speech input is exactly the text that would be selected. The positioning of the imperfection is done solely through the joystick since speech is not well suited for the pointing necessary to indicate the position of the imperfection. As the cursor is moved by the joystick, the coordinates and the type of material represented by the cursor is displayed at the bottom of the screen. If a defect is at the current position, a click produces a list of reasons why that material would be defective such as corrosion, scratch, etc. The defect type can be selected by the joystick or by speaking its name and the information would go into the database. The user can navigate to the main selection screen by selecting the "Main menu" option on all of the screens. One level up in the hierarchy can also be achieved through a single selection.

The relationship between the user interface design principles and the Navigator 2 user interface is:

- simplicity of function. The only functions available to the user are to enter skin imperfections for one of four aircraft, to transfer data to another computer, to enter identification information both for the vehicle and for the inspector, and to see a screen that describes the Navigator 2 project.
- no textual input. The identification information required entering numbers. A special dialogue was developed to enable the entering of numeric information using the joystick as an input device. This was cumbersome for the users but only needed to be performed once per inspection.
- controlled navigation. The interface was arranged as a hierarchy. The top level consisted of a menu that gave a choice of function. Once the inspection function and then the vehicle were chosen, the area of the skin inspected was navigated to via selecting an area of the aircraft to expand. Once an imperfection was indicated, the user had to select one of the allowable types of imperfections. At each stage, the user could go up one level of the hierarchy or return to the main menu.

One of the lessons learned with Navigator 2 is the power of forcing the use of a common vocabulary. Since the average age of the aircraft is 35 years, the types of defects encountered is a very stable set. Previously one inspector would call a defect "gouged" while another inspector would call the same defect a "scratch." What's the difference between a gouge and a scratch? How much material does it take? How much time does it take to repair? What skill of labor is needed? The logistics problem is much more difficult without a standardized vocabulary. Thus there is a serendipitous advantage in injecting more technology.

A second lesson is that in some cases the speech recognition front end mistakenly produces the wrong output. Speech recognition systems typically have an error rate of 2% to 10%. The unexpected output may cause the application to produce the wrong result. In one of Navigator 2's early demonstrations the user was attempting to exit the application. The speech recognition system thought a number was spoken. At that point

the application was expecting a second number but there was no match since the user was saying "exit", "quite", "bye", etc. The system appeared to be frozen when in actuality there was a mismatch between what the application software was expecting and what the user thought the application state was. The solution was to give the user more feedback on the state of the application by additional on-screen clues. Also a novel application input test generator was developed that took a description of the interface screens and created a list of all possible legal exits from each screen.

A third lesson learned was the criticality of response time. When speech recognition was done in software on Navigator (circa 1995) it was 12 times real time, which became very frustrating. People are less patient when they are on the move than they are when they are at a desktop. People at a desk are willing to wait three minutes for the operating system to boot up, but when you are on the move, expectations are for instant response like that of portable tools such as a flashlight. For example, some airplanes have a digital computer to control the passengers' overhead lights. It is disconcerting that when the button is pushed it may take two or three seconds before the light turns on. Even a couple of second delay in a hand-held device is disruptive. Users typically continue to push buttons until there is a response. The extra inputs cause a disconnect between the software and the user. The software receives a stream of inputs but the user sees outputs that are related to inputs given a long time before the screen appears. The situation is similar to listening to yourself talk when there is a second or two delay in the sound played back. The user easily becomes very confused.

The field evaluation indicated that the inspection is composed of three phases. The inspectors would spend the same amount of time maneuvering their cherry picker to access a region of the airplane, visually inspecting and feeling the airplane's skin, and recording the defect's type and location. Navigator 2 reduced the paperwork time by half resulting in an overall time savings of about 18%. Training time to familiarize inspectors with the use of Navigator 2 was about five minutes after which they would proceed with actual inspections. A major goal of field evaluations is that users perform productive work. They do not want to redo something that was already done once.

The typical inspection requires about 36 hours discovering approximately 100 defects. Today the inspector takes notes on a clipboard. Upon completion, the inspector fills out forms on a computer. Each defect takes two to three minutes to enter. The data entry is thus an additional 3 to 4 hour task. Navigator 2 transmits the results of the inspection by radio in less than two minutes.

In summary, evaluations of inspectors before and after the introduction of Navigator 2 indicated a 50% reduction in the time to record inspection information (for an overall reduction of 18% in inspection time) and almost two orders of magnitude reduction in time to enter inspection information into the logistics computer (from over three hours to two minutes). In addition, Navigator 2 weighs two pounds compared to the cart the inspectors currently use with 25 pounds of manuals.

Speech Translation

The SR / LT application (Speech Recognition/Language Translation) consists of three phases: speech to text language recognition, text to text language translation, and text to speech synthesis. The application running on TIA-P (Tactical Information Assistant-Prototype, circa 1996) is the Dragon Multilingual Interview System (MIS), jointly developed by Dragon Systems and the Naval Aerospace and Operational Medical Institute (NAOMI). It is a keyword-triggered multilingual playback system, which listens to a spoken phrase in English, proceeds through a speech recognition front-end, plays back the recognized phrase in English, and after some delay (~8-10 secs) synthesizes the phrase in a foreign language (Croatian). The other, local person can answer with Yes, No, and some pointing gestures. The Dragon MIS has about 45,000 active phrases, in the following domains: medical examination, mine fields, road checkpoints, and interrogation. Therefore, a key characteristic of this application is that it deals with a fixed set of phrases, and includes one-way communication. A similar system is used in Iraq as a briefing aid to interrogate former Iraqi intelligence officials and to speak with civilians about information relevant to locating individuals (Chisholm, 2004). This shows the viability of the approach.

TIA-P is a commercially available system, developed by CMU, incorporating a 133 MHz 586 processor, 32MB DRAM, 2 GB IDE Disk, full-duplex sound chip, and spread spectrum radio (2Mbps, 2.4 GHz) in a ruggedized, hand-held, pen-based system designed to support speech translation applications. TIA-P is shown in Figure <TIA-P Photo>.

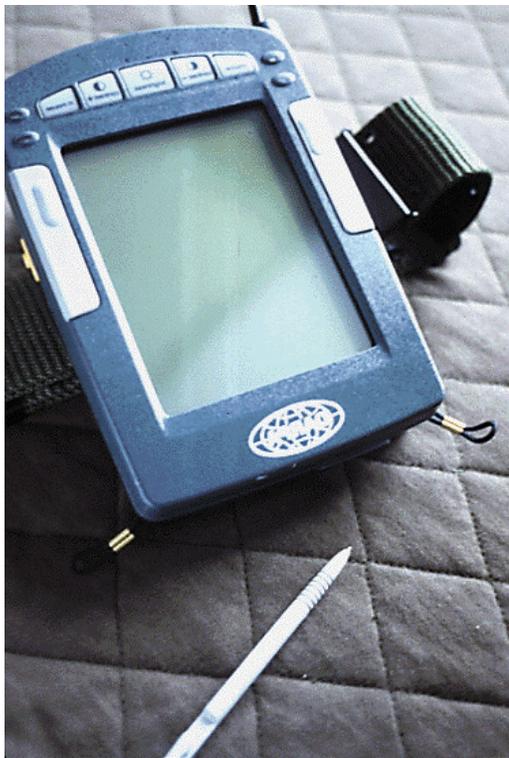


Fig. <TIA-P Photo> TIA-P Wearable Computer

Dragon loads into memory and stays memory resident. The translation uses uncompressed ~20 KB of .WAV files per phrase. There are two channels of output: the first plays in English, and second in Croatian. A stereo signal can be split and one channel directed to an earphone, and the second to a speaker. This is done in hardware attached to the external speaker. An Andrea noise canceling microphone is used with an on-off switch.

Speech translation for one language (Croatian) requires a total of 60 MB disk space. The speech recognition requires an additional 20-30 MB of disk space.

TIA-P has been tested with the Dragon speech translation system in several foreign countries: Bosnia (Figure <TIA-P>), Korea, and Guantanamo Bay, Cuba. TIA-P has also been used in human intelligence data collection and experimentation with the use of electronic maintenance manuals for F-16 maintenance.



Fig. <TIA-P> U.S. Soldier in Balkans Using TIA-P

The following lessons were learned during the TIA-P field tests: wires should be kept to a minimum; handheld display was convenient for checking the translated text; standard external electrical power should be available for use internationally; battery lifetime should be extended; ruggedness is important.

The smart modules (circa 1997) are a family of wearable computers dedicated to the speech processing application (Smailagic, Siewiorek, Reilly, 2001). A smart module provides a service almost instantaneously and is configurable for different applications. The design goals also included: reduce latency, remove context swaps, and minimize weight, volume, and power consumption (Reilly, 1998; Martin, 1999). The functional prototype consists of two functionally specialized modules, performing language translation and speech recognition. The first module incorporates speech to text language recognition and text to speech synthesis. The second module performs text to text language translation. The LT module runs the PANLITE language translation software (Frederking, Brown, 1996), and the SR module runs CMU's Sphinx II continuous,

speaker-independent speech recognition software (Ravishankar, 1996, Li *et al*, 1989) and Phonebox Speech Synthesis software.

Figure 3 depicts the structure of the speech translator, from English to a foreign language, and vice versa. The speech is input into the system through the Speech Recognition subsystem. A user wears a microphone as an input device, and background noise is eliminated using filtering procedures. A language model, generated from a variety of audio recordings and data, provides guidance for the speech recognition system by acting as a knowledge source about the language properties. The Language Translation engine uses an Example-Based Machine Translation (EBMT) system, which takes individual sentence phrases and compares them to a corpus of examples it has in memory to find phrases it knows how to translate. A lexical MT (glossary) translates any unknown word that may be left. The EBMT engine translates individual "chunks" of the sentence using the source language model and then combines them with a model of the target language to ensure correct syntax. When reading from the EBMT corpus, the system makes several random-access reads while searching for the appropriate phrase. Since random reads are done multiple times, instead of loading large, continuous chunks of the corpus into memory, the disk latency times will be far more important than the disk bandwidth. The Speech Generation subsystem performs text to speech conversion at the output stage. To make sure that misrecognized words are corrected, a Clarification Dialog takes place on-screen. It includes the option to speak the word again, or to write it in. As indicated in Figure <Translator Structure>, an alternative input modality could be the text from the Optical Character Recognition subsystem (such as scanned documents in a foreign language), which is fed into the Language Translation subsystem.

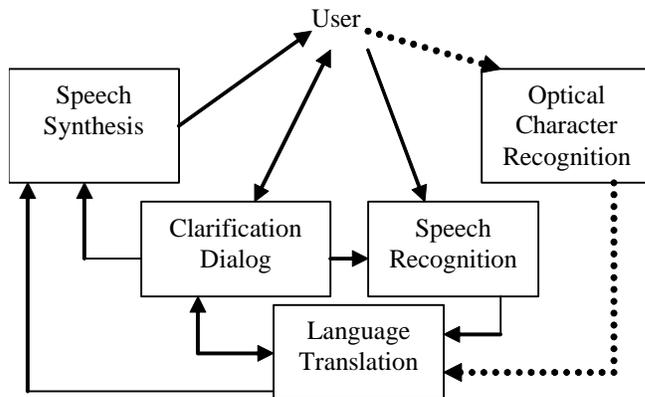


Fig. <Translator Structure>. Speech Translator System Structure

User interface design went through several iterations based on feedback during field tests. The emphasis was on getting completely correct two - way speech translation, and having an easy to use, straightforward interface for the clarification dialogue.

The speech recognition code was profiled and tuned. Profiling was performed to identify “hot spots” for hardware and software acceleration and to reduce the required computational and storage resources. A six times speedup was achieved over the original

desktop PC system implementation of language translation, and five times smaller memory requirements (Christakos, 1998). Reducing OS swapping and code optimization made a major impact. Input to the module is audio and output is ASCII text. The speech recognition module is augmented with speech synthesis. Figure <Smart Module> illustrates a combination of the language translation module (LT), and speech recognizer (SR) module, forming a complete stand-alone audio-based interactive dialogue system for speech translation.

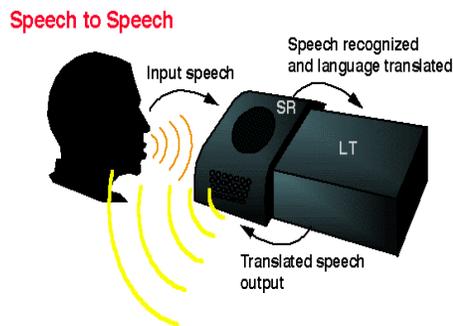


Fig. <Smart Module>. Speech Recognizer (SR) and Language Translator (LT) Smart Module

Target languages included Serbo-Croatian, Korean, Creole French, and Arabic. Average language translation performance was one second per sentence.

The key factors that determine how many processes can be run on a module are memory, storage space, and available CPU cycles. To minimize latency, the entirety of an application's working dataset should be able to stay memory resident.

Figure <Speech translator> depicts the functional prototype of the Speech Translator Smart Module, with one module performing language translation, and another one speech recognition and synthesis.

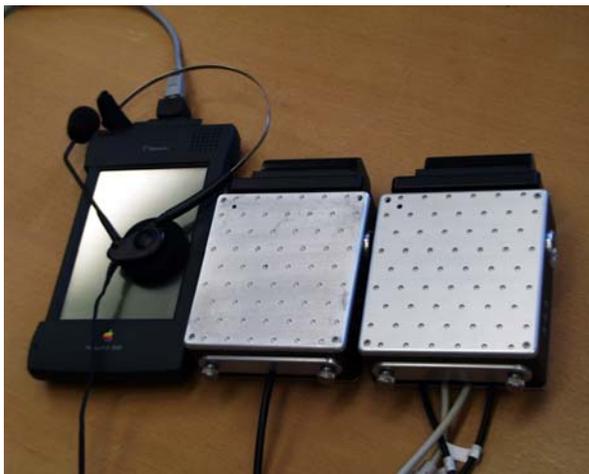


Fig. <Speech translator>. Speech Translator SM Functional Prototype

Wearable Tactile Displays

Tactile interfaces have been used successfully in wearable computing in many applications, ranging from early work on sensory prostheses and navigation aids (Collins et al., 1977; Bach-y-rita & Kercelz 2003) to recent developments in learning manual skills (Huang et al., 2010) and rehabilitation (Markow et al., 2010; Dimitrijevic, 1996). Most tactile displays are based on mechanical actuators or electrical stimulation of the skin. Examples of mechanical actuators include vibrators constructed from masses mounted off-center on a spinning motor shaft, solenoids, piezoelectric actuators, and pin arrays. Generally, off-center mass vibrators are used for less precise tasks, such as presenting an alert, due to their longer start and stop times (in the range of 100ms). Electrical stimulation systems use surface electrodes to create a vibration-like sensation without the use of moving parts. However, depending on the water content of the skin, location, voltage, current, electrical waveform, and electrode size, the perception may range from a tingle, itch, or buzz to a sharp or burning pain (Kaczmarek et al., 1991). While electrical stimulation has many potential benefits, including design simplicity, display resolution, negligible latency, and adjustability of sensation, the changing moisture level of human skin and inconsistent contact during movement makes these systems difficult to use in practice (Lee, 2010).

With both electrical and mechanical methods of stimulation, a tactile display interface designer needs to consider where on the body the display will be placed due to the highly varying sensitivity of different areas (Guyton, 1991). High resolution and fidelity displays can be used on the tongue, lips, and fingers (Bach-y-rita & Kercelz 2003), but the wrists (Lee, 2010), legs, and back can be surprisingly insensitive. In addition, a tactile display may be masked while a user is in motion due to the self-stimulation caused as clothing moves over the body.

One of the most common uses of a mobile tactile system is to provide better feedback for a graphical user interface (GUI). HCI researchers have shown that tactile feedback can increase users' accuracy when pressing virtual buttons rendered on a touch screen (Hoggan et al, 2009), as is common on many modern mobile phones. Alerts are another common use of mobile tactile displays. A simple example is the "silent," vibration mode on most mobile phones, where the phone vibrates to indicate an incoming call. (Lee & Starner, 2010) have shown that a wrist-based, three vibrator system could be used to communicate richer alert information to the user, such as caller ID. Several projects have explored how best to place actuators on the forearm and determined which features of tactile patterns convey information most efficiently (Lee & Starner, 2010; Brown et al., 2006; Chen et al., 2008; Borst et al, 2009). The assumption in many of these projects is that tactile alerts may be less distracting than visual or auditory alerts during critical tasks such as driving. This assumption is based on evidence from experimental psychology that users can better divide their attention across modalities as opposed to in the same modality (Wickens & Hollands, 1999). Indeed, Lee found that her tactile system is less distracting when the user is performing a visually intensive task than the current practice of retrieving and visually checking the screen of a mobile phone (Lee & Starner, 2010).

Similarly, soldiers have found tactile alert systems useful to communicate commands covertly and with little distraction while in the field (Gilson et al., 2007). Wearable tactile displays have been used for many years as sensory prosthetics or to augment the user's natural sensors. For example, much work has been performed creating tactile displays that help people who are blind navigate an environment (Collins et al., 1977). Similarly, directional tactile systems have been used by soldiers for improving situational awareness while clearing buildings (Lindeman et al., 2005). Bach-y-rita's work in the area of sensory prosthetics is of particular interest, as his displays often involve electrical stimulation of the tongue or forehead and have been used for sensory substitution for sight, vestibular balance, and tactile sensation from other parts of the body (Bach-y-rita & Kercelz 2003).

“Passive Haptic Learning” is a recent use of mobile tactile displays. In a series of experiments described by (Huang et al., 2010), subjects learn simple piano melodies while attending other tasks. Participants were equipped with the Mobile Music Touch system; a mobile phone-based music player and a fingerless glove fitted with small vibrators inserted above the thumb and each finger (see Figure XX). The melody to be learned is played repeatedly through earphones and, as each note is played, the finger that would use to play the respective key on the piano (if the user was at a piano and not mobile) is stimulated. Even though participants were required to focus their attention on a distractor task, such as a reading comprehension exam, they still learned the note sequence. A control group who heard the audio repeatedly but did not receive tactile stimulation performed significantly worse. These experiments suggest that passive training for manual tasks might be possible with wearable tactile displays.



FIGURE <Mobile Music Touch Glove>: The Mobile Music Touch glove allows users to learn note sequences while performing other tasks.

Building on the above work, (Markow et al., 2010) report preliminary results where practice with the Mobile Music Touch system may assist in hand dexterity and sensation

rehabilitation in participants with quadriplegia due to partial spinal cord injury. In more mature work, (Dimitriovic et al., 2006) describe a series of experiments using a "Mesh Glove" that uses electrostimulation, sometimes applied below conscious sensation, on the hand to help stroke patients recover arm mobility without active participation. While the Mesh Glove is not designed to be mobile necessarily, the two systems suggest that wearable tactile displays might be worn during the user's everyday life to aid in rehabilitation.

Performance Evaluation

Figure <Performance> illustrates the response time for speech recognition applications running on TIA-P and SR Smart Module. As SR is using a lightweight operating system (Linux) versus Windows 95 on TIA-P and the speech recognition code is more customized, it has a shorter response time. An efficient mapping of the speech recognition application onto the SR Smart Module architecture provided a response time very close to real-time. To ensure system responsiveness, it was important to provide feedback to the person in near real-time.

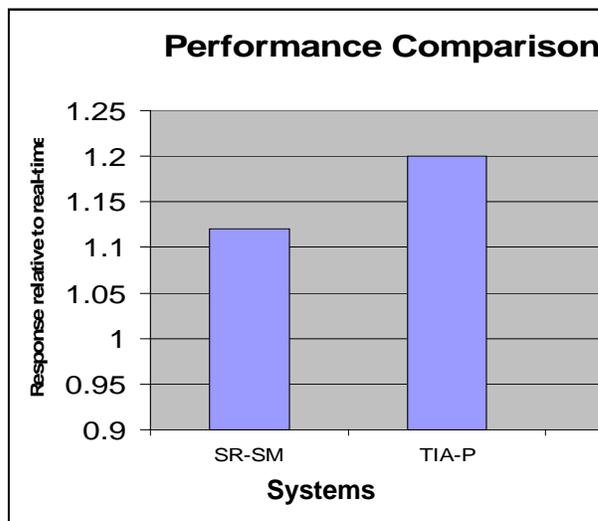


Fig. <Performance>. Response Times (lower is better)

The lessons learned from tests and demonstrations include: manual intervention process to correct misrecognized words incurs some delay; swapping can diminish the performance of the language translation module; the size of display can be as small as a deck of cards.

The required system resources for speech translator software are several times smaller than for the laptop / workstation version, as shown in Table <Resources>.

	Laptop / Workstation	Functional Module SR / LT	Optimized Module SR/LT
Memory Size	195 MB	53 MB	41 MB
Disk Space	1GB	350 MB	200 MB

Table <Resources>. Comparison of Required System Resources

Dual Purpose Speech

In industry, most speech recognition on mobile computers concentrates on the tasks of form filling or simple interface commands and navigation. One reason is that speech interfaces are often socially interruptive when other people are nearby. Speech translation, as with the TIA system above, is a different class of interface. The computer is an essential enabler of the conversation. Lyons *et al* (Lyons, 2004) introduce a different type of conversation enabler in their Dual Purpose Speech work.

Dual Purpose Speech is easiest to discuss using a scenario. Tracy, a wearable user equipped with a head-up display and a Twidder keyboard, is in conversation with a recently introduced colleague. Pressing a button on the keyboard, the wearable user enables speech recognition and says “Bob, what is your phone number so that I have it for later?”

The wearable recognizes that its user wants to record a phone number and starts the user's contact list application. It attempts to recognize the name spoken and enters that into the application. However, it also saves the speech so that the user can correct the text later if there is an error.

Bob responds “Area code 404”

“404,” repeats Tracy.

“555-1212,” completes Bob.

“555-1212,” continues Tracy who presses another button on her keyboard indicating the interaction is over, “Ok, I have it!”

On Tracy's head-up display a new contact has been made for “Bob (404) 555-1212.” When Tracy finishes her conversation, she clicks an “accept” button on the application because she has recognized the information correctly. Tracy could also edit the information or play back the audio recorded during the interaction with Bob. Note that Tracy verbally repeated the information that Bob provided—a good conversational practice. Tracy both confirmed that she understood the information and provided Bob with an opportunity to correct her if necessary. However, this practice is also good from a privacy standpoint. Tracy wears a noise-canceling microphone which is thresholded to record only her own voice and not that of her conversational partners. In this way, Tracy respects the privacy of her colleagues.

Lyons *et al* have designed Dual Purpose Speech applications for scheduling appointments, providing reminders for the user, and communicating important information to close colleagues. However, the key point of this research from the perspective of this section is that these applications allow the user to manipulate information on their wearables as part of the process of communicating (thus, the “dual

purpose” name). The users may actively format their speech so as the system can better understand them, and they may have to correct the system afterwards. However, the interface is manipulated and the information is entered as part of a social process.

This style of interface provides a contrast to the traditional desktop computer where the user's attention is assumed to be dedicated to the interface. Other wearable computing related fields also attempt to create interfaces that are driven by the user's interactions with the environment. For example, Feiner's early augmented reality systems attempted to display appropriate repair instructions based on the user's actions during the repair process (Feiner, 1993). Such awareness of the user's context and goals may allow wearable computers to be utilized where a user's lack of attentional or physical resources would normally preclude traditional desktop applications.

Perception

Just as dexterity is impaired when a user is on-the-go, the user's ability to perceive a wearable's interface is lessened. The vibration and visual interference from a moving background interferes with visual tasks. Background noise and the noise from the body itself affect hearing. The moving of clothes over the body and the coupling of mechanical shock through the body can lessen the user's ability to perceive tactile displays. Sears *et al* describe these detriments to mobile interaction caused by environmental and situational factors as “Situationally-Induced Impairments and Disabilities” (Sears, 2003). These researchers and others are developing procedures to test human performance in mobile computing tasks in context (in this case, walking a path) (Barnard, 2005; Barnard and Press). Such research is sorely needed as not enough is known about how to adequately simulate mobile computing scenarios in testing. For example, in Barnard *et al*'s work on performing reading comprehension tasks on PDAs while walking, lighting levels affected workload measures more when walking a path than when walking on a treadmill. The community needs to develop understanding about the interactions between mobility, attention, and perception in common mobile computing scenarios in order to adequately develop testing environments for mobile interfaces.

In the past, such work focused on cockpits, both for aviation and automobiles (Wickens, 2000; Melzer, 1997; Velger, 1998). However, the U.S. military's Land Warrior project has highlighted the need for such research for dismounted users who are on-the-go (Blackwood, 1997). Some researchers have begun exploring mobile output devices for very specific tasks. For example, Krum (Krum, 2004) describes experiments with a head-up display (HUD) which focus on determining how to render overhead views of an area to encourage learning of the layout of the surrounding environment while the user is navigating to a goal on foot. As mobile augmented reality is becoming practical from a technical standpoint, researchers have begun to address perceptual issues. While not a mobile experiment, Laramee and Ware have investigated head-mounted displays to determine the relative effects of rivalry and visual interference between binocular and monocular displays with varying levels of transparency. As the market determines which

mobile contexts are most important for users, experiments such as these will help determine how to design interfaces to least interfere with the user's primary tasks while providing the most value in terms of augmentation.

Research Directions

The evolution of computing has shown it takes several years to develop a user interface style which often emerge quite a while after the technology threshold has been passed. The thresholds represent the time when microprocessors have the capability of supporting the indicated form of interface. Figure <UI Performance Thresholds> depicts the increase in microprocessor performance (measured in millions of instructions per second, or MIPS) as a function of time. In the early 1960's, Gordon Moore of Intel made the observation/prediction that the capacity of semiconductor chips was doubling every year. Similar trends have been noted for microprocessor speed, magnetic disk storage capacity, and network bandwidth. The points depicted in Figure <UI Performance Thresholds> are the performance thresholds necessary for each of the user interface types. Thus a textual interface requires 1 MIPS, a graphical user interface (GUI) 10 MIPS, a handwriting interface 30 MIPS, a speech recognition interface 100 MIPS, natural language understanding 1000 MIPS, and vision understanding, 10,000 MIPS.

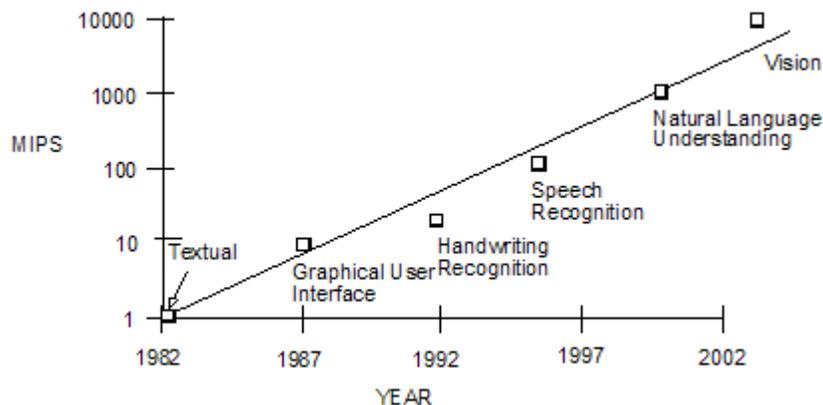


Figure <User Interface Performance Thresholds>

Because ease of use is so closely associated with human reaction, it is much more difficult to quantify. There are at least three basic functions related to ease of use: input, output, and information representation.

Figure <Use Modalities> summarizes several points for each of these basic functions. Note that unlike the continuous variables for capacity and performance, the ease of use metrics are discrete.

Just as the performance of microprocessors has increased over time as shown in Figure <UI Performance Thresholds>, the characteristics of the user interface shown in Figure <Use Modalities> are also moving out with time. For example, the keyboard with an alpha/numeric display using textual information is representative of timesharing systems of the early 1970's. The keyboard and mouse, graphical output, and iconic desktop are representative of personal computers of the early 1980's. The addition of handwriting recognition input, speech synthesis output, and multimedia information began emerging in the early 1990's. It takes approximately one decade to broadly disseminate new input, output, and informational representations. In the 2000s decade speech recognition, position sensing and eye tracking are becoming common inputs. Heads-up projection displays should allow superposition of information onto the user's environment.

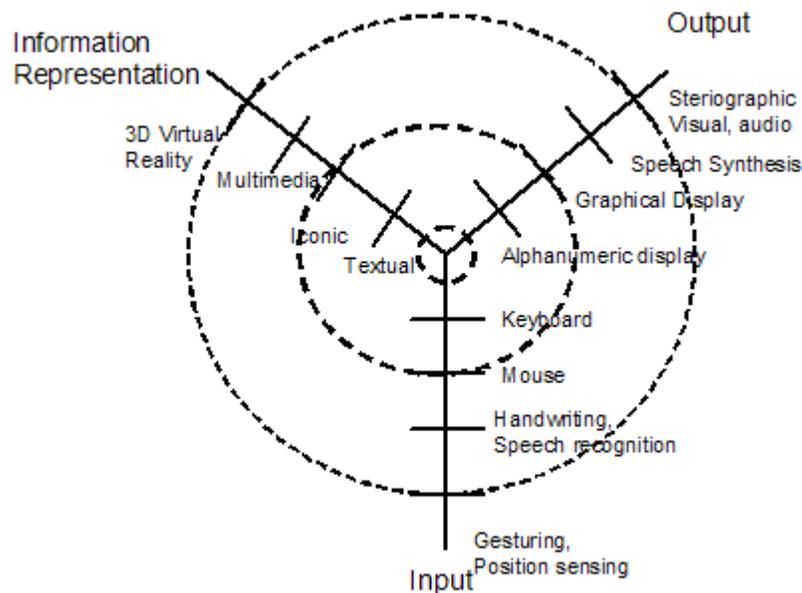


Figure <Use Modalities> Kiviat Graph for Wearable Computer Use Modalities

Context Awareness

The next step in the evolution of wearable computers is context awareness. Context-aware computing is aware of a user's state and surroundings and the mobile computer modifies its behavior based on this information. A user's context can be quite rich, consisting of attributes such as physical location, physiological state (such as body temperature, heart rate, and skin resistance), emotional state (such as angry, distraught, or calm), personal history, daily behavioral patterns, etc. If a human assistant were given such context, he or she would make decisions in a proactive fashion, anticipating user needs. In making these decisions, the assistant would typically not disturb the user at

inopportune moments except in an emergency. The goal is to enable mobile computers to play an analogous role, exploiting context information to significantly reduce demands on human attention. Context-aware intelligent agents can deliver relevant information when a user needs that information. These data make possible many exciting new applications, such as augmented reality, context aware collaboration, wearable assisted living, augmented manufacturing, and maintenance.

Example System: Virtual Coach

The Seating Coach (or Power Wheelchair Virtual Coach) is an intelligent system that can guide power wheelchair users in achieving clinician established goals for body positioning. It was developed at Carnegie Mellon, with the University of Pittsburgh Center for Assistive Technology as the project client. The Seating Coach provides capabilities such as interacting with the user in a manner appropriate to their capability, inferring user capabilities from the data, indicating user compliance, and creating reminders to do past due activities. The Seating Coach sensing and computational infrastructure determines if a user employs auxiliary seating functions according to the physical therapist's prescriptions and coaches him/her to use them in a proper and timely fashion. The Seating Coach can help power wheelchair users to reduce the risk of chronic sores by comparing auxiliary seating function use against a prescription of positions and their durations as established by a physical therapist. It can aid clinicians in tracking results from training and reinforces proper technique to reduce the incidence of injuries caused by improper power wheelchair use.

A power wheel chair allows the user to recline, tilt, elevate the seat and change leg-rest elevation of the chair. Tilt involves the same change in angle of backrest, seat and leg-rest. Recline changes the backrest angle only and leg-rest elevation changes only the leg-rest angle. The seat elevation only changes the elevation from the ground.

The Seating Coach records sensor data on user position and usage patterns. An array of pressure sensors is distributed over the back rest and seat cushion providing the pressure information to the virtual coach, as shown in Figure <Virtual Coach>. Three tilt sensors determine the tilt angle of the back rest, seat recline, and leg rest elevation, as illustrated in Figure <Tilt Functions>. Tilt, recline, and leg rest elevation are monitored for any improper sequences in using seat functions, such as recline without tilt, leg rest elevation without recline, and recline or tilt angles that are too large, as well as any inappropriate use of seat functions during driving. Infrared sensors are used to detect obstacles behind the chair and determine the height of the seat. Pressure sensors are monitored for weight distribution inferring body positions.

The data analysis software extracts underlying user patterns. A clinician-friendly interface allows therapists to prescribe physical activities, rules for proper use of the wheelchair, as well as parameters for user compliance goals. We created a prescription format which is easy to use and expressive enough to cover a range of subjects and conditions. The prescription encompasses all the power seat functions and sets limits for the user. The prescription is specified using the following information: activity type, parameter, duration of the activity, time gap after which to repeat the activity, and alert after specified number of rule violations (Table <Sample prescription>).

After entering a usage prescription, the clinician can periodically monitor the wheelchair user's compliance to those recommendations. Reminders are generated to prompt the user to comply while alerts indicate non compliance and are sent to the user, as shown in Figure <Virtual Coach Display>.



Figure <Virtual Coach>: Virtual Coach

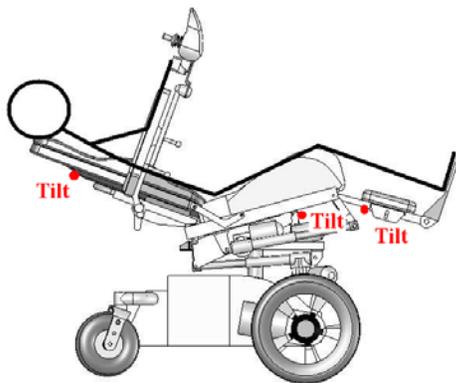


Figure <Tilt Functions>: Tilt function and placement of sensors

Activity	Parameter			Duration			Gap			Alert after
	Min	Ideal	Max	Min	Ideal	Max	Min	Ideal	Max	
Tilt	25°	30°	35°	25 sec	30 sec	35 sec	20 min	30 min	2 hrs	10
Recline	10°	15°	20°	4 mins	5 mins	6 mins	4 hrs	5 hrs	6 hrs	15
Feet Elevation	25°	30°	35°	50 sec	1 min	1min 10 sec	1hr 30 mins	2 hrs	2 hrs 30 mins	20
Pressure	0	60mm	200mm	0 sec	0 sec	30 mins	0 sec	0 sec	0 sec	5

Table <Sample Prescription>: Sample prescription, filled by the clinician

In addition, a Wizard of Oz experiment was conducted where users made selections from a variety of feedback modalities and preferences to create a user interface (Liu et al 2010). Nine power seat functions (PSF) users and six clinicians were recruited for this study. The subjects reviewed modalities with various properties using a computer demonstration program with supplemental devices. Their preferences and suggestions

were collected using a questionnaire and interviews. An animation of PSF usage tasks was preferred because it conveyed essential information. The subjects rank ordered the interaction options as a function of situation. For example, 40% of the subjects selected vibration for the reminding theme in the noisy restaurant scenario, and 46.7% selected speech for the reminding theme in the home scenario. As another example, ranking of vibration location on the seat had armrest ranked highest (60%), and headrest as lowest (6.7%). These studies will inform user interaction designs for virtual coaches.



Fig. <Virtual Coach Display>. Example System: Virtual Coach

Example: eWatch, A Proactive Assistant

Our research on context-aware computing employs unsupervised machine learning techniques to combine real time data from multiple sensors into a model of behavior that is individualized to the user. The eWatch is a wearable sensing, notification, and computing platform built into a wrist watch form factor making it highly available, instantly viewable, ideally located for sensors, and unobtrusive to users (Smailagic, Siewiorek 2005). Bluetooth communication provides a wireless link to a cellular phone or stationary computer. eWatch senses light, motion, audio, and temperature and provides visual, audio, and tactile notification. The system provides ample processing capabilities with multiple day battery life enabling realistic user studies. Figure <eWatch> shows a few representative eWatch screenshots: sensor waveforms, calendar, and messages. Figure <eWatch Hardware> illustrates sensors and main hardware components on eWatch. We developed a wearable computing platform with power-aware hardware and software architectures, and showed how online nearest neighbor classification can identify and recognize a set of frequently visited locations.

Knowing about the user's location is an important aspect of a context-aware system. Using eWatch we developed a system that identifies previously visited locations. Our method uses information from the audio and light sensor to learn and distinguish different environments. We recorded and analyzed the audio environment and the light conditions at several different locations. Experiments showed that locations have unique background noises such as car traffic, talking, noise of computers, air conditioning and television. The



Fig. <eWatch> eWatch - A Pro Active Assistant

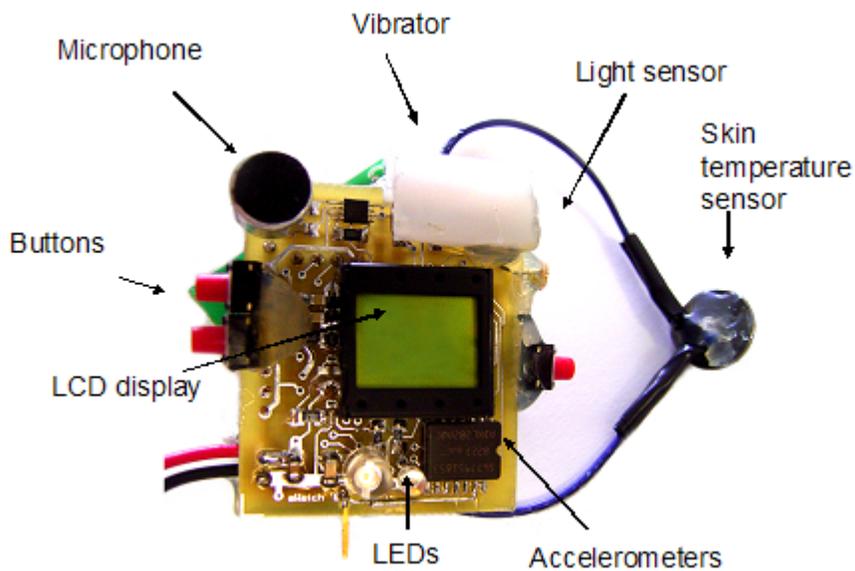


Fig. <eWatch Hardware>. The eWatch Hardware

light sensor sampling at a high frequency can also provide additional information beyond the brightness of the location. We observed that the frequency characteristics of light conditions tend to remain constant in most locations. For our study, audio data was

recorded with the built-in microphone at a sample rate of 8 kHz and the light sensor at a frequency of 2048Hz. At every location five consecutive recordings of audio and light were taken, separated by 10 second pauses. For every recording we sampled the microphone for four seconds (32000 samples) and the light sensor for 0.5 seconds (1024 samples). The recorded data was then compressed and stored into flash memory. Locations frequently visited by the user were recorded; the rooms of the user's apartment (living room, kitchen, bedroom, bathroom), their office, the lab, different street locations on the way to university, the interior of a bus, and several restaurants and supermarket. Each location was visited multiple times on different days. In total, we collected 600 recordings at 18 different locations.

We estimated the power spectral density of the recorded sensor data using Welch's method. A 128-point FFT was calculated for a sliding window over the complete recording and averaged over frequency domain coefficients for all windows. The result is a smoothed estimation of the power spectral density. To reduce the number of feature components, the Principal Component Analysis was used. The dimensionality of the feature vector was reduced to its first five principal components. To visualize the feature space, Figure <Audio and Light > shows the first three components of the feature vectors after a Linear Discriminant Analysis (LDA) transformation.

The nearest neighbor method with a 5-fold cross validation was used for classification. Three different feature sets were evaluated: features from the light sensor only, microphone only and both sensors combined. As expected, the combination of both sensors gave the best results in identifying the location. The classification with the light sensor alone gave an overall result of 84.9% correctly classified samples. The classifier confused the *lab* and *office* location and also the *bus* with the *street*. This occurred because both location pairs can have similar light conditions. Using only the audio sensor the overall recognition accuracy was 87.4%. The *office* and *apartment* location were confused in this case. Both sensors combined gave the best result of 91.4%. Locations that could not be distinguished well with only one sensor were classified more accurately with both sensors combined.

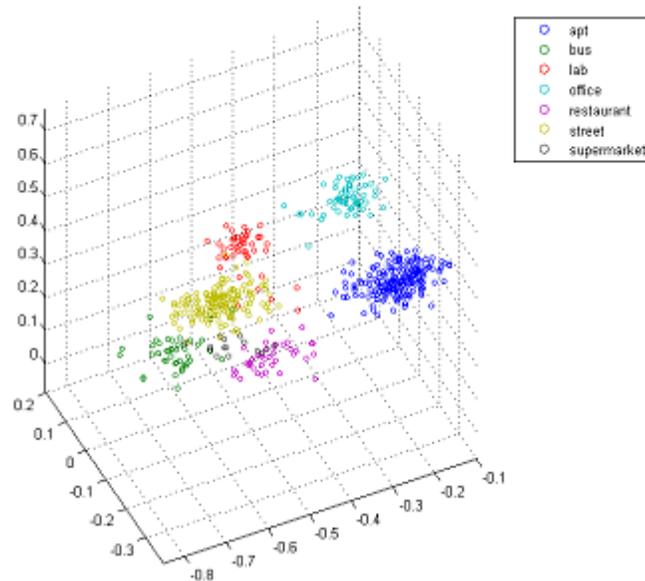


Fig. <Audio and Light> Learning Context: Audio and Light Sensor Clustering

Wearable Cognitive Augmentation

An important goal of our research is to determine the cognitive state of a user – especially the user’s cognitive load – from external observations. Knowing the user’s cognitive state would enable development of proactive cognitive assistants that anticipate user needs much like a human assistant does.

What makes this attempt possible is an unprecedented advance in measuring and understanding brain activity during complex cognition using functional magnetic resonance imaging (fMRI), Figure <fMRI>. The brain activity measured with fMRI is only one step removed from the neural activity itself. fMRI provides a measure of the oxygenated hemoglobin in the capillary beds in which the neural activity is occurring. Routinely used protocols in neurocognitive research on advanced MRI scanners sample the entire cortex approximately once per second. It is feasible to pursue a research plan is to recognize some of the brain/cognitive states that should be amenable to improvement, and then develop an intelligent tutoring system that uses the fMRI-measured brain activation to guide the tutoring, to infer current mental states and to rapidly guide the learner to desired mental states.

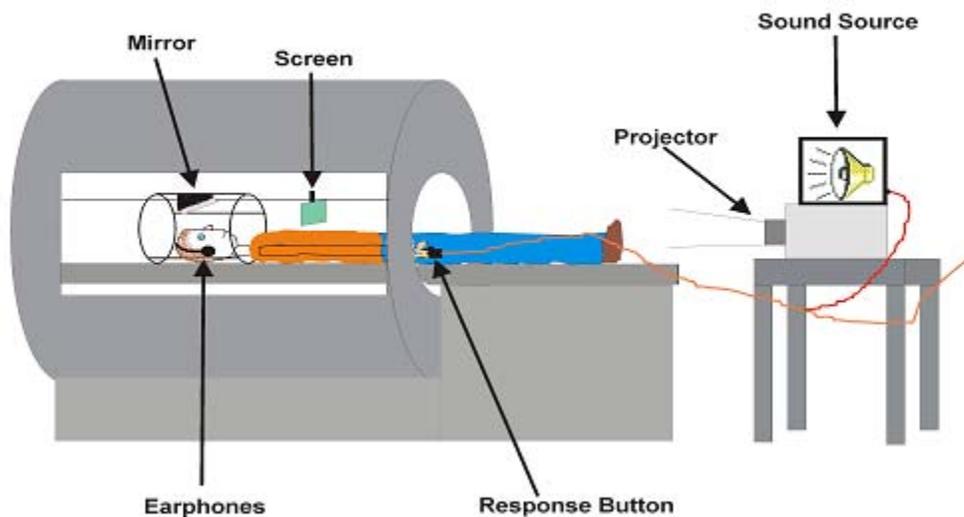


Fig. <fMRI> fMRI Experiment Configuration

There is also a maximum on the total activation across cortical areas. Such a system-wide capacity constraint might be expected to operate when subjects co-perform two tasks that draw on non-overlapping brain areas. The requirement of non-overlap assures that any constraint on performance is not due just to competition for the same neural mechanisms.

In a study that found evidence for such a constraint, the two tasks were auditory sentence comprehension and mental rotation (Just et al., 2001). If there were no system-wide capacity constraint, one would expect that because the two tasks draw on different neural substrates (language and spatial-related areas respectively), the activation in the dual task would simply be the union of the activations in each of the two single tasks. However, the activation in the dual task was far less than the union of the two single tasks. The activation associated with each individual task decreased by 30-50% in the dual task condition, as shown in the representative brain slices of individual subjects, Figure <Human Brain>, and in the graph showing the group data results, Figure <Capacity>. The decrease in the dual task condition applied to 17 of the 18 subjects. Thus there appears to be a detectable upperbound on the total amount of activation that can be sustained in a set of cortical areas. (This study was widely applied to the question of what happens in the brain during driving and using a cell phone simultaneously).

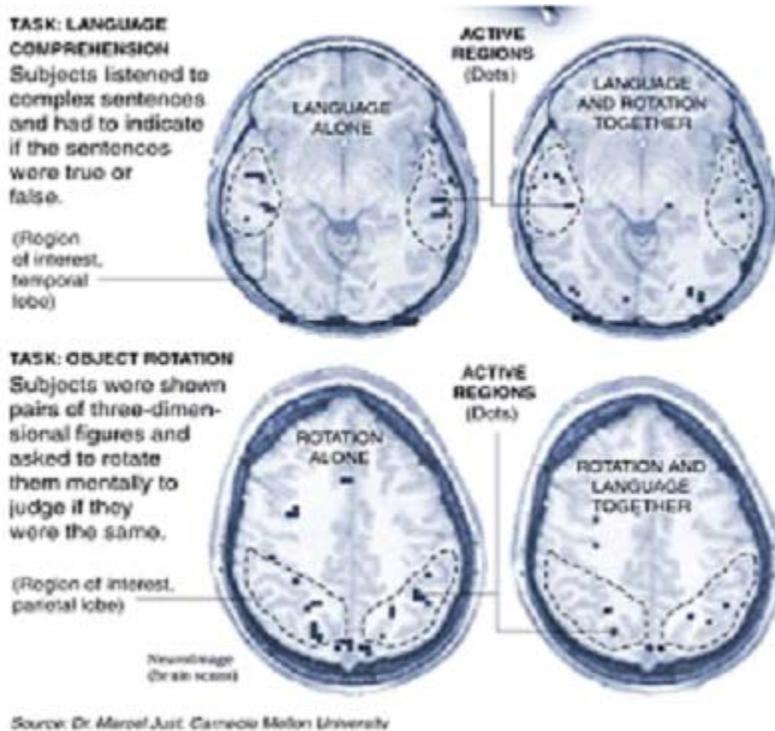


Fig. <Human Brain> Study of Cortex

Capacity constraint in association areas:

Activation volume is less in dual task compared to single tasks, even for tasks without cortical overlap (auditory comprehension and mental rotation)

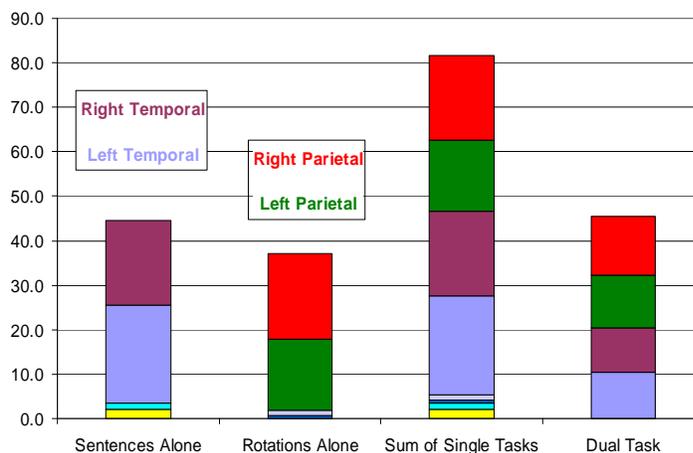


Fig. <Capacity> Activation Volume in Dual Task Comparing to Single Task

Conclusion and Future Challenges

Wearable computers are an attractive way to deliver a ubiquitous computing system's interface to a user, especially in non-office-building environments. The biggest challenges in this area deal with fitting the computer to the human in terms of interface, cognitive model, contextual awareness, and adaptation to tasks being performed. These challenges include:

- 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 applications will be required. Furthermore, there may be a set of metaphors each tailored to a specific application or a specific information type.
- 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 low-weight, low-energy 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.
- Quick interface evaluation methodology. Current approaches to evaluate a human computer interface requires elaborate procedures and with scores of subjects. Such an evaluation may take months and is not appropriate for use during interface design. These evaluation techniques should especially focus on decreasing human errors and frustration.
- Matched capability with applications. The current thought is that technology should provide the highest performance capability. However, this capability is often unnecessary to complete an application and enhancements such as full-color graphics require substantial resources and may actually decrease ease 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.
- Context Aware Applications. 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? These, plus many other questions, have to be addressed before context aware computing becomes possible. Some initial results have been reported in (Krause, Smailagic, Siewiorek 2006).

References

- Anhalt, J., Smailagic, A., Siewiorek, D., *et al* (2001). Towards context aware computing. *IEEE Intelligent Systems*, 6(3), 38-46.
- Azuma, R. (1997). A survey of augmented reality. *Presence*, 6(4), 355-386.
- Bach-y-Rita, P. & Kerzelz, S. (2003). Sensory substitution and the human-machine interface. *Trends in Cognitive Science*. 7(12), 541-546.
- Barnard, L., Yi, J.S., Jacko, J. A., Sears, A. (2005). An Empirical Comparison of Use-in Motion Evaluation Scenarios for Mobile Computing Devices. *International Journal of Human Computer Studies*, 62, 487-520.
- Barnard, L., Yi, J., Jacko, J. A., & Sears, A. (In press). Capturing the Effects of Context on Human Performance in Mobile Computing Systems. To appear: *Personal and Ubiquitous Computing*.
- Blackwood, W. (1997). *Tactical display for soldiers*. National Academy of Sciences, Washington, D.C.
- Blaskó, G., Feiner, S., Coriand, F. (2005). Exploring interaction with a simulated wrist-worn projection display. *Proc. Ninth IEEE Intl. Symp. on Wearable Computers*. (pp. 2-9). IEEE Computer Society Press, Los Alamitos, CA.
- Bodine, K., Gemperle, F. (2003). Effects of functionality on perceived comfort of wearables. *Proc. Seventh IEEE Intl. Symp. on Wearable Computers*. IEEE Computer Society Press, Los Alamitos, CA.
- Borst, C. & Baiyya, V., (2009). A 2d haptic glyph method for tactile arrays: design and evaluation. *Proceedings of the World Haptics Conference*, 599–604.
- Brown, L., Brewster, S., and Purchase, H. (2006). Multidimensional tactons for non-visual information presentation in mobile devices. *Proceedings of the 8th Conference on Human-Computer interaction with Mobile Devices and Services*, Helsinki, Finland.
- Chamberlain, A., Kalawsky, K. (2004). A comparative investigation into two pointing systems for use with wearable computers while mobile. *Proc. Eighth IEEE International Symposium on Wearable Computers* (pp. 110-117). IEEE Computer Society Press, Los Alamitos, CA.
- Chen, H.-Y., Santos, J., Graves, M., Kim, K., and Tan, H. (2008) Tactor localization at the wrist. *Proceedings of EuroHaptics*, Madrid, Spain, 209–218,. Collins, C., Scadden, L.

& Alden, A. (1977). Mobile studies with a tactile imaging device. Proceedings of the Fourth Conference on Systems & Devices For The Disabled, Seattle, WA.

Chisholm, M. (2004). Technology that speaks in tongues. *Military Information Technology*, 8(2). Retrieved from <http://www.military-information-technology.com/article.cfm?DocID=424>

Christakos, C.K. (1998). *Optimizing a language translation application for mobile use*. Master's thesis, Carnegie Mellon University, Department of Electrical and Computer Engineering, Pittsburgh, PA, USA.

Clawson, J., Lyons, K., Starner, T., and Clarkson, E. (2005) The impacts of limited visual feedback on mobile text entry using the mini-QWERTY and Twiddler keyboards. *Proc. IEEE Intl. Symp. on Wearable Computers*.

Dimitrijevic, M., Soroker, N. & Pollo, F. Mesh glove electrical stimulation. *Science & Medicine*, Vol. 3, 1996.

Feiner, S., MacIntyre B., and Seligmann, D. (1993). Knowledge-based augmented reality. *Communications of the ACM*, 36(7), 52-62.

Frederking, R. E., Brown, R. (1996). The Pangloss-lite machine translation system: expanding MT horizons. *Proceedings of the Second Conference of the Association for Machine Translation in the Americas* (pp. 268-272).

Gemperle, F., Kasabach, C., Stivoric, J, Bauer, B., and Martin, R. (1998). Design for wearability. *Second International Symposium on Wearable Computers*, (pp. 116-122). IEEE Computer Society Press, Los Alamitos, CA.

Gilson, R., Redden, E., and Elliot, L. (eds.). (2007). Remote tactile displays for future soldiers. Army Research Laboratory Technical Report ASL-SR-0152.

Guyton, A. (1991), *Basic Neuroscience and Anatomy*. W.B. Saunders Co., Philadelphia, PA.

Hoggan, E., Brewster, S. & Johnston, J. (2008) Investigating the effectiveness of tactile feedback for mobile touchscreens. Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, Florence, Italy.

Hong, J.I., Landay, J.A. (2004). an architecture for privacy-sensitive ubiquitous computing. *Proc. Second International Conference on Mobile Systems, Applications, and Services*, (pp. 177-189). IEEE Computer Society Press, Los Alamitos, CA.

Huang, K., Kohlsdorf, D., Starner, T., Ahlrichs, C., Ruediger, L., Do, E., & Weinberg, G. (2010). Mobile music touch: mobile tactile stimulation for passive learning. Proceedings

of the SIGCHI Conference on Human Factors in Computing Systems, Atlanta, GA, 791-800.

Iachello, G., Smith, I., Consolvo, S., Abowd, G et al (2005). Control, deception, and communication: evaluating the deployment of a location-enhanced messaging service. *Proc. Seventh International Conference on Ubiquitous Computing*, (pp. 213-231). IEEE Computer Society Press, Los Alamitos, CA.

Kaczmarek, K., Webster, J., Bach-Y-Rita, P. & Tompkins, W. (1991). Electrotactile and vibrotactile displays for sensory substitution systems. *IEEE Trans. On Biomedical Engineering*, Vol. 38, 1-16.

Kollmorgen, G. S., Schmorow, D., Kruse, A., Patrey, J. (2005). The cognitive cockpit-state of the art human-system integration. *Proc. 2005 Interservice/Interindustry Training Simulation and Education Conference*, DARPA, Arlington, VA.

Krause, A., Smailagic, A., and Siewiorek, D.P. (2006). Context-aware mobile computing: learning context-dependent personal preferences from a wearable sensor array. *IEEE Transactions on Mobile Computing*, 5(2), 113-127.

Laramee, R. and Ware, C. (2002). Rivalry and interference with a head mounted display. *ACM Transactions on Computer-Human Interface*, 9(3), 238-251.

Lee, S (2010). Buzzwear: supporting multitasking with wearable tactile displays on the wrist., Unpublished doctoral dissertation. Georgia Institute of Technology, School of Interactive Computing, Atlanta, GA.

Lee, S. & Starner, T. (2010). BuzzWear: alert perception in wearable tactile displays on the wrist. *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, Atlanta, GA, 433-442.

Li, K.F., Hon, H.W., Hwang, M.J., and Reddy, R. (1989). The Sphinx speech recognition system. *Proc. IEEE ICASSP*.

Liu, H.-Y., Cooper, R., Cooper, R., Smailagic, A., Siewiorek, D., Ding, D., Chuang, F.C. (2010), Seating Virtual Coach: A Smart Reminder for Power Seat Function Usage, *Technology and Disability*, IOS Press, 21(4), 41-48.

Lindeman, R., Sibert, J., Mendez-Mendez, E., Patil, S. & Phifer, D. (2005). Effectiveness of Directional Vibrotactile Cuing on a Building-Clearing Task. *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, Portland, Oregon, 271-280.

Lyons, K., Skeels, C., Starner T., Snoeck, B., Wong, B., and Ashbrook, D. (2004). Augmenting conversations using dual-purpose speech. *Proc. User Interface and Software Technology*. (pp. 237-246).

- Lyons, K., Skeels, C., Starner, T., Snoeck, C., Wong, B., Ashbrook, D. (2004). Augmenting conversations using dual-purpose speech. *Proc. User Interface and Software Technology*. (pp. 237-246).
- Lyons, K., Starner, T., and Plaisted, D. (2004) Expert typing using the Twiddler one handed chord keyboard. *Proc. IEEE Intl. Symp. on Wearable Computers*. (pp. 94-101).
- Lyons, K. and Starner, T. (2001). Mobile capture for wearable computer usability testing. *Proc. Intl. Symp. on Wearable Computers*.
- Markow, T., Ramakrishnan, K., Huang, K., Starner, T., Eicholts, S., Garrett, H., Profita, A., Scarlata, C., Schooler, A., Tarun., Backus, D., (2010) Mobile Music Touch: Vibration Stimulus as a Possible Hand Rehabilitation Method. Proceedings of Pervasive Computing Technologies for Healthcare, Munich, Germany.
- Martin, T. (1999). *Balancing batteries, power, and performance: system issues in CPU speed-setting for mobile computing*. PhD Thesis, Carnegie Mellon University, Department of Electrical and Computer Engineering, Pittsburgh, PA, USA.
- Melzer, J. and Moffitt, K. (1997). *Head mounted displays: designing for the user*. McGraw-Hill, New York.
- Ockerman, J. (2000). Task guidance and procedure context: aiding workers in appropriate procedure following. Technical report, Georgia Institute of Technology, Atlanta, GA.
- Oulasvirta, A., Tamminen, S., Roto, V., Kuorelahit, J. (2005). Interaction in 4-second bursts: the fragmented nature of attentional resources in mobile HCI. *Proc. of SIGCHI Conference on Human Factors in Computing Systems*. (pp. 919-928).
- Ravishankar, M. (1996). *Efficient algorithms for speech recognition*. PhD thesis, Carnegie Mellon University, School of Computer Science, Pittsburgh, PA, USA.
- Reilly, D. (1998). *Power consumption and performance of a wearable computing system*. Masters thesis, Carnegie Mellon University, Electrical and Computer Engineering Department, Pittsburgh, PA, USA.
- Sears, A., Lin, M., Jacko, J. and Xiao, Y. (2003). When Computers Fade: Pervasive Computing and Situationally-Induced Impairments and Disabilities. *Proceedings of HCII 2003*. (pp. 1298-1302).
- Siewiorek, D.P. (2002). Issues and challenges in ubiquitous computing: new frontiers of application design. *Communications of the ACM*, 45(12), 79-82.

- 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*, 2(2), 259-292.
- Siewiorek, D.P, Smailagic, A., Bass, L., Siegel, J., Martin, R., Bennington, B. (1998). Adtranz: a mobile computing system for maintenance and collaboration. *Proceedings 2nd IEEE International Conference on Wearable Computers*.
- Smailagic, A. (1997). "ISAAC: a voice activated speech response system for wearable computers. *Proceedings of the IEEE International Conference on Wearable Computers*.
- Smailagic A., Siewiorek D.P., Martin R., Stivoric, J. (1998). Very rapid prototyping of wearable computers: a case study of VuMan 3 custom versus off-the-shelf design methodologies. *Journal of Design Automation for Embedded Systems*, 3(2-3), 219-232.
- Smailagic, A. and Siewiorek, D. P. (1994). The CMU mobile computers: a new generation of computer systems. *Proceedings of the IEEE COMPCON 94*. (pp. 467-473). IEEE Computer Society Press. Los Alamitos, CA.
- Smailagic, A. and Siewiorek, D. (1996). Modalities of interaction with CMU wearable computers. *IEEE Personal Communications*, 3(1), 14-25.
- Smailagic, A., Siewiorek, D. P., & Reilly, D. (2001). CMU wearable computers for real-time speech translation, *IEEE Personal Communications*, 8(2), 6-12.
- Smailagic, A., and Siewiorek, D. (1993). A case study in embedded system design: the VuMan 2 wearable computer. *IEEE Design and Test of Computers*, 10(3), 56-67.
- Starner, T., Snoeck, C., Wong, B., and McGuire, R. (2004) Use of mobile appointment scheduling devices. *Proc. ACM Conference Human Factors in Computing Systems*. (pp. 1501-1504).
- Starner, T. (2001). The challenges of wearable computing: part 1+2, *IEEE Micro*, 21(4), 44-67.
- Stein, R., Ferrero, S., Hetfield M., Quinn, A., Krichever, M. (1998). Development of a commercially successful wearable data collection system, *Proc. IEEE Intl. Symp. on Wearable Computers*. (pp. 18-24).
- Velger, M. (1998). *Helmet-mounted displays and sights*. Artech House, Inc., Norwood, MA
- Wickens, C. D., & Hollands, J. (1999). *Engineering psychology and human performance* (3rd edition). Prentice Hall.

