

Contrasting paradigms for the development of wearable computers

by C. Baber
D. J. Haniff
S. I. Woolley

In this paper, current applications of wearable computers are reviewed and categorized according to dimensions of "time" and "reference." The time dimension is based on whether the system uses information that is stored, information that is current, or information that can help in predicting future events. The reference dimension is concerned with the type of application, event, task, environment, person, or artifact. Each of these categories can be described in terms of its temporal features (stored, current, or predicted). It is proposed that these dimensions distinguish wearable computers from their desk-bound counterparts, and this raises the question of appropriate paradigms for wearable computers. A user-centered methodology is then presented and illustrated by paramedic and fire-fighter applications.

There has been rapid development in the field of wearable computers in the past five years. Publication of papers on wearable computers at ACM's special interest group on human-computer interaction (CHI) series of conferences in the IEEE (Institute of Electrical and Electronics Engineers) journals suggest that the concept has achieved academic respectability. Commercial products have appeared in recent years, e.g., Xybernaut Corporation and Teltronics, Inc., both produce Pentium** processors that can be worn by users. With the recent announcement of the IBM wearable computer (in addition to development work by major telecommunications organizations), it would appear that the concept of the wearable computer has matured.

Wearable computing

The term "wearable computing" implies a significant computational load. However, many processing tasks

require communication processes or environmental responses that are not computationally expensive and would therefore not require the power, and accompanying overhead, of a conventional microprocessor architecture. Ideally, the processing demands of wearable devices would be well-specified such that appropriate processor and peripheral architectures could be properly identified to match the activity demands of the device. Where noncomputational tasks, i.e., control and response functions, are required, arrays of master and slave microcontrollers may be appropriate. From this perspective, it would be possible to broadly categorize wearable devices based on their processing requirements, i.e., wearable control devices might require wearable signal-processing or communication devices, which might require wearable computational devices. This scheme implies a continuum of processing capabilities, ranging from task- and processor-specific to general-purpose computers. In order to explore this further, the following subsection presents a brief discussion of microprocessors, microcontrollers, and related architectures.

Microprocessors and microcontrollers. Microprocessors, microcontrollers, and digital signal processors (DSP) have been traditionally distinguished in terms of the level of processing complexity they can handle, e.g., from general-purpose to control-specific to signal-specific processors.

©Copyright 1999 by International Business Machines Corporation. Copying in printed form for private use is permitted without payment of royalty provided that (1) each reproduction is done without alteration and (2) the *Journal* reference and IBM copyright notice are included on the first page. The title and abstract, but no other portions, of this paper may be copied or distributed royalty free without further permission by computer-based and other information-service systems. Permission to *republish* any other portion of this paper must be obtained from the Editor.

Microprocessors, originally designed by Intel Corporation as general-purpose solutions for the electronic calculator market, quickly became the workhorse of the new desktop computer market. Microcontrollers were rugged, low-cost versions of microprocessors designed for embedded-systems-control applications such as automotive engine management. Although general-purpose microprocessors can still be found in many embedded systems, particularly where significant processing is required, the trend is toward microcontrollers, i.e., dedicated, high-performance, control-specific processors.

Microcontrollers provide a range of features required by control applications and their environments. A typical set of features (in comparison with microprocessors) might include: low power consumption, resilience to noise and vibration, small footprints, fast and versatile handling of input and output, efficient interrupt capabilities, fail-safe features such as watchdog timers for automatic recovery in the event of system lock-up and brown-out protection for recovery from power supply anomalies, and on-board features for serial and parallel communications.

Recently the distinction between microprocessors, microcontrollers, and DSPs has become blurred.¹ For example, Intel Corporation has added multimedia extensions (MMX) to its Pentium processor to support DSP and other multimedia operations. Hitachi Semiconductor, Inc. is producing a hybrid DSP version of its SH-2 microcontroller, and Microchip Technology's high-end PIC** microcontrollers have on-board multipliers.

There is now a great variety of microprocessor, microcontroller, and DSP families from which to choose, some of which began as application-specific solutions and have found other applications, and others that began as general-purpose microprocessors from which application-specific variants have emerged. Ideally processor architectures would be finely tuned to their applications, addressing the processing and I/O demands of the target activities.

It is possible that the breadth of the new processor continuum will be sufficient to supply the needs of wearable computing in the near future, although it is likely that any microprocessor solutions will require important modifications,² in particular pertaining to power management and heat dissipation.³

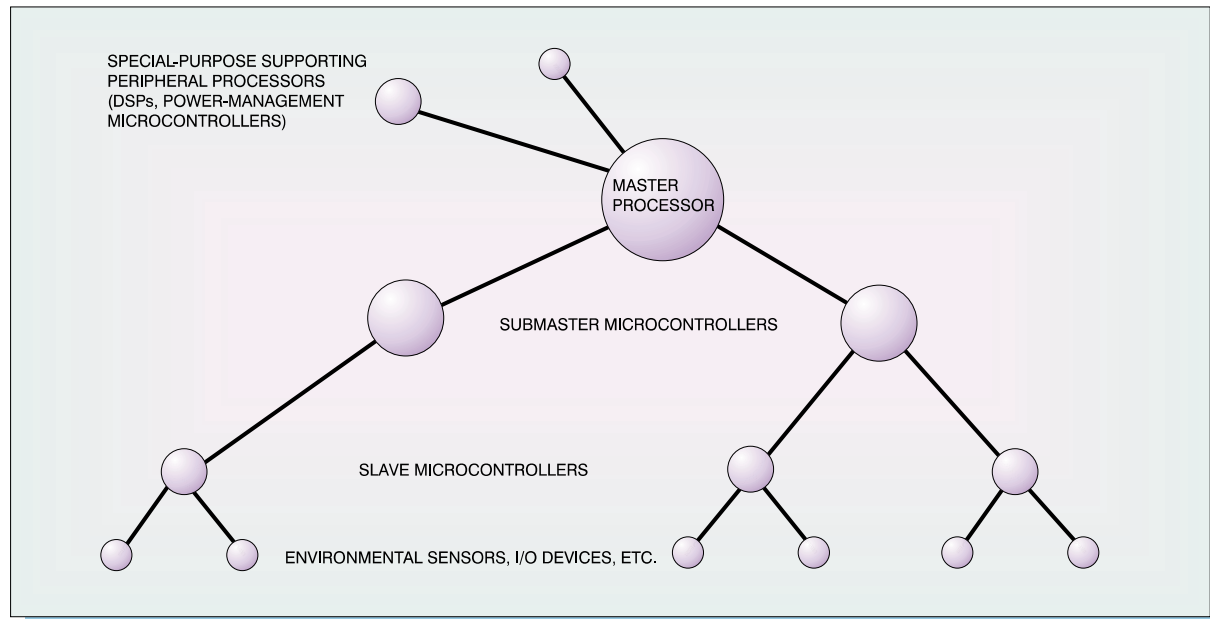
Furthermore, the inherently mobile nature of wearable systems demands efficient use of available power (battery packs) and external communications. The physical attributes of most of today's more powerful microprocessors makes them inappropriate solutions for wearable applications. The boards are large, heavy, fragile, and hot; they lack the robustness, control features, and low-power characteristics preferred in embedded systems. The failure of battery power supplies—despite improved packaging and methods of charging—to keep pace with Moore's Law⁴ means that power supply limitations are a significant impediment to the usability of wearable and hand-held computing devices. It is therefore essential that power economy is considered in the design of systems and their processing units.

Wearable computers as embedded systems. Ideally the wearable application activities will dictate the architecture and performance of the processor or processors. Distributing processing tasks among master, submaster, and slave microprocessors and microcontrollers and DSPs may significantly reduce processing and power demands on master processing systems. Configurable computing offers new and exciting alternatives to the flexibility vs performance trade-off.⁵ Hardware utilizing field-programmable gate arrays (FPGAs) can now provide highly tuned configurable logic blocks that can be modified within milliseconds (and perhaps microseconds in the future).

Wearable computing may be able to usefully implement distributed master and slave processor designs similar to those used in other embedded control systems (see Figure 1). For example, most luxury cars are now controlled by networked microprocessors and microcontrollers. The microcontrollers receive signals from sensors around the vehicle and report changes in expected value, i.e., usually once a specified threshold has been exceeded, the microcontroller sends a signal to another device. Similar distributions of low-cost processors could usefully add to the fail-safe capability of wearable systems with a serial or parallel body LAN (local area network) networking protocol, testing system integrity and making the best use of available resources.

Wearable computing can be seen as a *new embedded system* application. Ideally, processors will operate reliably in relatively hostile environments with minimum power consumption. Many of the new applications of wearable computing are safety-critical, for example, the applications reported in the final

Figure 1 Schematic of wearable master-slave system



sections of this paper. It may be that safety-critical applications will be the proving ground of wearable devices. If systems are to be successful, robust, light-weight, low-power solutions are required.

Our aim in writing this paper is to present a position on wearable technology. Clearly the conclusions of the paper will be grounded in contemporary technology. However, it is proposed that this paper might stimulate exploration and research beyond the current limits of software engineering and systems development, and continue the debate as to what is meant by the term wearable computer.

Classifying applications

In this section, applications that have been reported in the public domain will be reviewed. It is proposed that information used in these applications can be classified according to two dimensions: time and reference. Obviously wearable computers could be classified along other dimensions (see Underkoffler⁶). We considered including a dimension of location in this scheme. While location raises important issues (such as where the device will be used, how the device will communicate with other devices or be used to communicate with other people), it was felt that

this would not be applicable to the current applications that are reported in the wearable computers literature. Location is an important concern for mobile technologies, and there will be closer links with wearable and mobile computers in the immediate future. However, communication is not necessarily a central issue with wearable computers at the moment (with the obvious exception of the systems developed by Mann⁷). Consequently, we have opted for two dimensions with which to consider and classify contemporary wearable computers. The classification scheme could be considered analogous to a design space, and as with any design space, it might be the gaps in the scheme that are of most interest, i.e., those areas that are logically deducible from the scheme but have not yet been developed.

Time. Defining applications in terms of time moves us some distance from conventional applications. It does not make much sense to apply temporal factors to, for instance, a word processing package. Typically, wearable applications are intended to provide useful information *now* and will either interrupt or alert the user to the presence of potentially useful information. An analogy with desktop applications could be an alert to incoming e-mail. Another defining feature of wearable computers is that the in-

interaction between user and computer takes place while the user is doing other activities. This means that time becomes an issue not only as far as the type of information is concerned, but also in terms of how the user divides time and attention between computer and world. The temporal aspect of information can be defined by the terms past, present, and future. However, we propose the following terms:

- **Stored** (data collected during previous operation are recalled)
- **Current** (information from the user's world is ready for use)
- **Predicted** (the computer aids the user in making predictions about likely future events, usually over short time scales)

Reference. The second dimension we propose is reference. This describes the topic to which information refers. At present we have five terms under this dimension: event, task, environment, person, and artifact. In this section, we also incorporate the temporal dimensions defined above in order to review wearable applications.

Event. Events are activities or situations that occur in a defined time period. Consequently, many event topics will be covered by personal digital assistant (PDA) products, with their ability to combine functions such as diaries, appointment reminders, alarm clocks, etc. Current event information refers to something that is happening now (a good example of this type of information would be the pager, which indicates to the wearer that someone is trying to make contact at this moment). Stored events are activities that have already occurred and could be held in a diary record. Access could be similar to a simple PDA interface, or there could be sophisticated "association management" to allow the computer to advise the wearer on the salience of an event. Predicted events are activities that require resource allocation in the future. In this instance, it is not easy to see how an on-body device offers potential improvements over contemporary technology. However, Rhodes and Starner's Remembrance Agent⁸ is a good illustration of a wearable event manager that incorporates stored, current, and predicted information. In addition, the Nomadic Radio described by Sawhney and Schmandt⁹ provides asynchronous event-based auditory information that is current, as opposed to synchronous information provided through normal auditory communications, for example a telephone call. The Forget-Me-Not system¹⁰ presents stored information about the context of an

event (e.g., where it occurred, the people involved, and the task carried out) in order to augment memory.

Task. Tasks are specific activities that a person is performing. In contemporary wearable applications, task information typically refers to instructions, diagrams, or procedures. However, it could also refer to some monitoring of user performance, e.g., we are working on an aid, to assist in the training of surgical skills, that combines instrumented forceps (through strain gauges) with a display to indicate whether excessive force is being used in a newly acquired surgical procedure. In this instance, an aspect of the task is being monitored, evaluated, and reported back to the wearer. Stored task information could be, for example, a record of actions that the person has performed, e.g., to aid faultfinding by allowing the user to backtrack to earlier tasks.

Current task information would allow the user to access information that was applicable to a specific task, e.g., a particular step in a procedure,¹¹ or fault isolation trees for maintenance work,¹² or circuit diagrams and drawings for repair work.¹³ Predicted task information could present the user with information concerning the proposed outcome of a sequence of tasks; e.g., Webster et al.¹⁴ use a display of a three-dimensional model to show students what a particular model should look like after construction.

Environment. This type of information allows the wearer to make sense of the immediate environment. In this context, environment can be defined by its spatial characteristics, for example, locations of buildings, routes between points, etc., and by its physical characteristics, for example, temperature, humidity, noise level, etc. Stored environmental information could take the form of a visit "history" to show the wearer places the wearer has visited,¹⁵ which could be useful for tourists visiting a large site of antiquity. Current environmental information would allow the wearer to overlay relevant information onto the real-world scene, e.g., by using a transparent head-mounted display to project information onto building structures.¹⁶ Alternatively, current environmental information could take the form of instructions for particular route-finding procedures presented over headphones to visually impaired wearers.¹⁷ Physical environmental information is illustrated by the fire fighter example in this paper. Predicted environmental information would present the wearer with alternative routes through an environment, e.g., fire fighters might need the location

of utilities within a building so that they can plan appropriate responses to a fire.

Person. The role of wearable computers to handle information relating to persons has received a great deal of attention from different researchers. In this paper, we divide persons into self and others. Typically stored information relating to self is considered from a physiological perspective. For example, work by Picard¹⁸ shows how changes in physiological state can be stored to produce a model of the wearer's emotional state. For stored information relating to others, the basic premise is that an intelligent computer could process, say, facial or vocal features of a person, and then relay that information to another person. On a prosaic level, this might help avoid the embarrassment of forgetting someone's name, but on a more useful level, police officers might find it useful to have some means of aiding identification of either known criminals or a suspect being questioned on the street. Mann¹⁹ suggests that information about persons could be held with their pictures to provide the wearer with additional details to be used in conversation. For example, this might allow the user to remember when he or she last met the person and to recall information specific to either that meeting or to that person. Starner et al.²⁰ describe the use of their remembrance agent as a conversation aid in a discussion with a colleague; the agent can gather notes at a conference and present appropriate associations that can be used in the conversation.

Current self information, in this context, could be very useful to fire fighters who may need to know their own rate of oxygen consumption or heart rate in order to determine how long to spend in a fire. Current other information could be used to track or locate someone, either through "tagging" the person, or through heat-sensitive cameras used to find persons buried under rubble. These examples would allow the person topic to be linked with the environment topic. Current other information relates to the use of wearable computers in computer-supported cooperative work. Billinghurst et al.²¹ demonstrate that seeing the person one is working with significantly enhances performance (as compared to either seeing nothing or seeing a "virtual equivalent"). Predicted self information could, for instance, cover exercise monitors. The idea for predicted other information could be used for warning of possible dangerous activity in the person being spoken to, say, for use by police officers or psychiatric nurses. Alternatively, predicted person information could re-

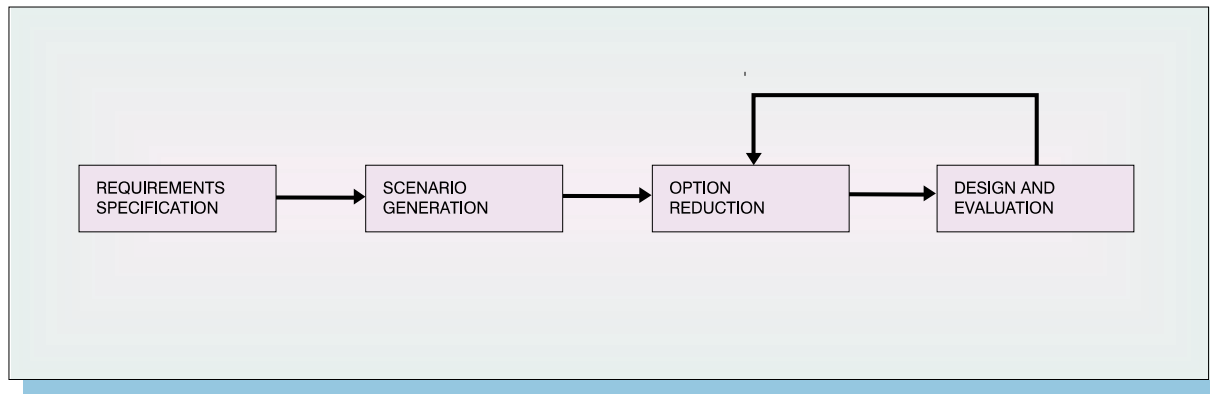
fer to the physiological condition of an accident victim; e.g., a paramedic could enter or record the patient's vital signs and be informed of likely changes in state.

Artifact. An artifact would be an object or thing in the world that the computer is able to recognize. The computer might be capable of performing object recognition to aid the wearer in information retrieval tasks, such as looking for a specific book in the library, or it might be capable of presenting additional information to help search for books with specific content. These tasks would require use of stored information (both in terms of the rules or features to be used for object recognition, and in terms of "knowledge" of the content of specific books). One might also wish to find specific information about an object; for example, Feiner et al.²² report the use of wire-frame graphics to highlight the location of specific features in a printer. This allows the wearer to use stored information to find components. The use of a computer to help find something would also relate to current information; for instance, Foner²³ describes the use of "sonification" to search for and identify objects. The Ubiquitous Talker²⁴ can provide additional contextual information about objects through speech as well as visually. Jebara et al.²⁵ have developed the Dynamic Personal Enhanced Reality System (DyPERS), which presents audio and video clips associated with an object; the clips were recorded autonomously in a previous encounter with that object. Another way in which stored information could be used is through records of a person's performance on a defined task. This would allow the computer to support or monitor improvement over time, and ultimately provide support for learning or training. Alternatively, the computer might identify objects that a person is using during the performance of a task, and evaluate their performance. This would link the artifact topic to the task topic. One example of predicted information for artifacts would be an aid for machine inspectors, who could interrogate machine tools without having to either open or approach the machine. This would provide a means of checking tool life as the wearer walked through the factory.

Activity-based design methodology

Norman²⁶ has recently questioned whether the contemporary approach to the design of computer applications can be sustained for future technologies. He suggests that a primary reason why the desktop metaphor remains in vogue is that it allows design-

Figure 2 Stages of design process



ers and manufacturers to strive for the production of multipurpose products, i.e., products and applications that can be used for any job in any office. This seems to make good business sense, with most people finding most of the functions useful. Nevertheless, it also leads to claims that the majority of the functions offered will not be used by the majority of users.²⁷

Norman's proposal is that future computers will offer restricted function sets, and that people will select the function set most appropriate to their defined requirements. He calls this "activity-based computing," in that computers will be designed to support specific activities. This would mean that the wearer would have less equipment to operate and carry, and it could also mean that interaction with the computer could be performed via familiar objects and products. Relating this argument to the discussion in our first section, we propose that microprocessors are intended to be used for multipurpose applications (although, of course, they can be used for single-function applications), while microcontrollers tend to be used for single-function applications. While developments are blurring the distinction between these products, the distinction is the basis for the argument in this section.

We believe that activity-based computing extends the basic assumptions of user-centered design and requirements engineering, because it allows us to consider the architecture that might be appropriate for a specific wearable product. In this section we report two design projects (paramedic and fire-fighter applications) currently underway at the University of Birmingham. The paramedic application (P³Co)

uses a PC-based wearable computer with a see-through head-mounted display and the fire-fighter application (HotHelmet) uses microcontrollers connected to peripherals mounted in the fire fighter's helmet. Both applications initially used a user-centered design process. The approach taken in this research is to use scenario-based design methods²⁸ as a means of defining suitable applications of wearable technology. There are several reasons why we have opted for scenario-based design approaches in the initial stages of this work:

1. The approaches allow us to concentrate on user requirements rather than on technical issues, which means that our initial designs address the needs of potential users rather than the capability of available technology.
2. The intention is to produce scenarios that are as clear and succinct as possible while retaining sufficient detail to promote new design concepts. The technique that we are using is evolving into a scenario-based requirements engineering methodology and can be described by Figure 2.

Stage 1: Requirements specification. There are two sets of requirements generated in this work. The first set consists of general requirements that can be applied to all wearable products, and the second set consists of requirements that are generated for each specific application. The general requirements are taken from human factors literature and reports of wearable computers; consequently, they are being continually updated. The following three groups of bulleted items represent the set of general requirements that we are currently using:

- Information displayed will be succinct and inspectable.
- The user will not be overwhelmed by information.
- Information will be easy to interpret and assimilate by the user.

These requirements indicate the general requirement for the display to be informative without being cluttered and suggest that there will be limits to the amount of information that can be employed. The limits will be defined by the technology, e.g., in terms of available display space, screen resolution, etc., and by the limits of human information processing, e.g., in terms of available attention.

- The user will be able to retrieve relevant information quickly.
- The equipment will support continuous interaction.
- The dialogue will be easy to remember and follow.

Using the computer will require additional cognitive effort on the part of the wearer; for instance, the user must ensure that interacting with the computer will not interfere with other work. The equipment will be available when needed, i.e., there will be no requirement to engage in actions to log on, etc.

- There will be minimal interference between task and computer use.
- The computer can be used while the wearer is in motion.
- The computer can be used while one or both hands are busy.

These requirements emphasize the need to use the computer while doing other work. This means that if the other work involves a great deal of manual activity, then requiring manual interaction with the computer would intrude on this other work. A design will fail to meet these requirements if users must switch between task and computer.

Specific requirements are generated from a scenario for using the application.

Stage 2: Scenario generation. We first define a particular scenario to illustrate the context of use of the application. The scenario also allows generation of specific requirements (each requirement is given its own identification code). A context-of-use scenario is a short description of work activity, usually in terms

of current performance. The aim is to capture as succinctly as possible the activity, consequences, and characteristics of work in a format that can be readily accepted and discussed by potential end users. Previous work has suggested that context-of-use scenarios based on users' own words can lead to better appreciation of technological issues than descriptions couched in technical terms. An example of a context-of-use scenario for a speech-driven call-center system is: "Late one evening a customer telephones the main office to discuss possible design modifications to a product. The design team has gone home, but the help desk is staffed and it is possible to have the call routed to an engineer on the help desk. The caller decides that the matter would be best handled by a specific member of the design team and leaves a brief message."

The scenario is used to illustrate the importance of the general requirement and also to indicate how the task fits in with other activity during the type of incident in question. The initial phase of design involves interviews and focus-group discussions with end users. These are videotaped and analyzed in order to define specific requirements and possible design alternatives. A scenario is developed on the basis of information from end users, often in terms of typical work activity, and will be subject to revision as further discussions are held. The activity is defined in terms of activity type, e.g., data entry, situation awareness, access to specialist knowledge. Following this, specific requirements are outlined.

The specific requirements will arise from both discussion with potential end users and our own interpretation of the requirements. Typically, there will be a set of specific requirements, all of which can affect the acceptability and usability of the technology. Consequently, we ask end users to rank the specific requirements in order to get an idea of the importance of each requirement.

In addition to specific requirements, the scenario also allows us to produce a first-pass generation of possible design combinations. We examine the source of information that end users (in our first example, paramedics) currently use and define design combinations in terms of the generic tasks.

Assuming a user-centered perspective means that we consider design in terms of specific combinations of modality. A modality defines the way information is received by the person, e.g., through visual or auditory means, and by the user's action, whether spo-

Table 1 Initial acceptance criteria

Factors	Method	Metrics	Worst	Target	Best
<i>Performance measures</i>					
Task	Critical path	Time: current	–15 percent	–5 percent	Same
	User trials	Time	–15 percent	–5 percent	Same
Training	1st vs 3rd trial	Percent improvement	No change	3rd > 1st	—
<i>Usability</i>					
User evaluation	SUS	Scale (0–100)	50	60	70
	SUMI	Scale (0–100)	50	60	70
	Heuristics	Scale (0–10)	<6	6	>6

SUS = System Usability Scale (see Brooke²⁹).

SUMI = Software Usability Measurement Inventory (see Kirakowski³⁰).

For heuristics scale, see Nielsen.³¹

ken or manual. The activities that a user performs, both with the computer and with objects or persons in the world, are considered in terms of possible modalities. For our paramedic example, treatment of a person would be manual, inspection of a person for injury would be both visual and manual, receiving information from the person would be auditory. On the other hand, recording information could be performed using pen and paper (manual), or through speaking to a computer or into a dictaphone (speech). The use of some modalities will not be immediately obvious for a specific activity, and might not even be possible.

Discussion with end users allows us to constrain the initial design ideas to possible modality combinations. The next phase is to check back with the persons helping with the scenario to ensure that the descriptions are correct and complete. At present we are using rankings to prioritize the particular requirements. This forms part of an ongoing process, with additional information presented at later meetings. Thus, the document remains a living text that is open to modification and that we encourage people to modify.

Stage 3: Option reduction. The scenario allows us to generate broad design concepts, in terms of combinations of modalities. This next stage allows us to consider the pros and cons of possible designs that are illustrative of a particular modality combination. In order to reduce the design space to a workable limit, we first apply the specific requirements to eliminate some designs.

Each design option is considered and an illustrative design concept proposed. This usually takes the form of either a sketch or a verbal description of what the

system might look like. The concept is then rated against a specific requirement (0—does not fit, 1—might fit, 2—definite fit). In order to determine whether a design concept can be carried into the next stage of design, we apply a simple selection criterion, based on the binomial theorem (see Appendix).

Stage 4: Design and evaluation. The final phase of this process is to build working prototypes and have them evaluated by end users. For each prototype, we define a set of evaluation criteria. For example, Table 1 presents the initial evaluation criteria applied to the P³Co product (discussed later). If the design fails to meet one of the evaluation criteria at this stage, then it requires modification. Setting levels of performance in the design evaluation will later help us to evaluate the product.

We now describe two examples of systems designed using this methodology: the first system in support of paramedics, the second in support of fire fighters.

Example: P³Co (Paramedic patient reporting and protocol support computer). In the design shown in the first sidebar, we have specified the general activity “PPS1.0: Paramedic treating a casualty will record baseline observations and casualty details and will receive guidance on protocol.” Next, a “context of use” description is developed (a similar approach has been presented by Coble et al.³²). Then we rank the requirements on a scale from 1 to 5, from most to least important, and list the modalities through which the tasks can be performed.

Given the modalities, we can produce an initial set of options for designs. Treatment of the casualty can only be performed manually, whereas recording ob-

- 1. *General requirement:* PPS1.0: Paramedic treating a casualty will record baseline observations and casualty details and will receive guidance on protocol.
- 2. *Application type:* Access specialist knowledge; data entry
- 3. *Context of use:* A resident telephones the emergency number 999 and requests an ambulance, for a relative (casualty) who has lost consciousness. The call is communicated to the dispatch center. Details regarding the injury to the casualty and any casualty details, together with location of the incident, are recorded. This information is communicated to the closest available ambulance, which responds within a specified time period. On arrival at the scene, the ambulance crew members assess the situation, collect any additional information, and perform the ATLS (advanced trauma life support) procedure. This procedure requires visual and manual checks of airway and breathing and checks of circulation, e.g., pulse, heart rate, and blood pressure. Basic observations will also require reporting the Glasgow coma scale and EMV (eyes, motor, verbal) scores for unconscious patients. If the casualty has suffered cardiac arrest, the immediate concern is with recovery of normal sinus rhythm, and a paramedic will use ECG (electrocardiogram) to diagnose rhythm and apply appropriate treatment protocol by making readings and administering defibrillation and drugs. The paramedic makes a report and the casualty is taken to the hospital and handed over to hospital personnel.

4. <i>Specific requirements:</i>	<i>Ranking:</i>
PPS1.1: The paramedic will maintain full attention on treating casualty.	3
PPS1.2: Patient details and baseline observations are recorded.	4
PPS1.3: Baseline observations are recorded during treatment.	2
PPS1.4: Protocol guidance appropriate to treatment is provided.	5
PPS1.5: Casualty handling keeps hands busy.	1
PPS1.6: Observations are transmitted directly to hospital.	1

5. <i>Modality matrix:</i>		
Activity	Information Source	Modality
Treat casualty	Casualty	Manual
Speak to casualty	Casualty	Speech
Use equipment	Equipment	
Make readings	Equipment, casualty	
Record observations	Paper (computer)	
Check protocol	Paper (computer)	

Table 2 Matching options to specific requirements (0, the criterion is not met; 1, the criterion is partly met; 2, the criterion is fully met)

Requirement	Design Option											
	1	2	3	4	5	6	7	8	9	10	11	12
Casualty handling requires both hands	0	0	0	0	2	2	2	2	2	2	2	2
Observations sent directly to hospital	1	1	1	1	2	2	2	2	1	1	1	1
Baseline observations made during treatment	0	0	0	0	2	2	2	2	1	1	1	1
Paramedic to maintain attention on casualty	0	0	0	0	2	2	2	2	2	2	2	2
Patient details and baseline observations recorded	1	1	1	1	2	2	2	2	1	1	1	1
Protocol guidance appropriate to treatment	2	1	2	1	2	1	2	1	2	1	2	1
Total rating	4	3	4	3	12	11	12	11	9	8	9	8

servations could be performed manually (by writing down readings), automatically (by having the casualty monitored), or via speech. Examining the possible combination of options produces a finite design space of twelve design options. (Three activities have only one modality, one has three modalities, and two have two modalities; $1 \times 1 \times 1 \times 3 \times 2 \times 2 = 12$.)

Each design option is illustrated by a verbal description or a sketch. Space constraints limit the number of concepts that can be presented, but two design concepts are described here:

Option 1 describes the current system. Paramedics' treatment of the patient, use of equipment, and recording of observations are all manual tasks. This means that there is a possibility that one of these activities might be slighted at the expense of the others, i.e., recording of observations is typically made after treatment (which means that significant values are captured by the paramedic and held in working memory). Checking of readings and protocol are both visual tasks, and performing one of these tasks could interfere with performance, i.e., checking of protocol requires interruption of patient treatment.

Option 5 describes a system that employs speech to record information and visual display of protocol information.

The option reduction stage involves evaluation of options against specific requirements. Table 2 shows the ratings applied to each design for the specific requirements. Each design option was considered by the team (in this instance, one paramedic, one industrial designer, and two of the authors). If the team felt that the option did not meet a specific requirement, it was allocated a score of 0, while a score of 2 indicated a general feeling that the option met the requirement. Inclusion is defined by a significant majority (see Appendix) with a score of at least 10. From this, a candidate set of accepted options is defined that includes design options 5, 6, 7, and 8.

The process was followed for each of the general domain requirements, resulting in two candidate design options: Option 5—visual display and speech recognition, and Option 7—auditory display and speech recognition. It was further concluded that Option 7 had a lower overall score, because of the problems associated with auditory information and working memory limits. Thus, it was decided that Option 5 was the best candidate for this application, and a demonstration model has been developed.

The initial requirements for the paramedic system make it necessary to use a platform capable of running speech recognition, with enough storage for data concerning patients and some protocols and procedures, and potential for communications support.

Given these technical requirements, we developed a working prototype using commercially available technology: Seattle Sight Systems' monocular, monochrome display and InterVision System's wearable 486-based PC (see Figure 3). Future developments could take advantage of recent developments in display technology, e.g., MicroOptical Corporation's eyeglass-based displays, or wearable Pentium-based processors.

In order to determine what information to display to the user, we follow an object-oriented programming approach. There are many object-oriented programming techniques;³³ we start with a scenario description. In brief, we take the "context of use" scenario and extract "objects" and "actions" from this description. Specifying the importance of these objects and activities then allows them to be grouped into broad categories, e.g., the object "casualty" has associated with it a set of objects (blood pressure, pulse, respiration rate, temperature, blood sugar, peak oxygen flow, oxygen saturation) and activities (procedure, decision making, data collection, reporting). These objects and activities lead to an initial screen design (see Figure 2). The screens are developed in Visual Basic** 5.0, running on a Pentium II processor with an AURIX speech recognizer (from DERA, the Defense and Evaluation Research Agency of the UK Ministry of Defense) to allow hands-free interaction.

The baseline observation screen (Figure 4) requires the paramedic to record basic physiological information. The speech recognition system can access the data entry fields sequentially (initiated by the word "next") or directly by the field name (for example "pulse"). The medical form (Figure 5) is a dynamic display that presents procedural information and allows the user to enter data. For example, the interface will display the drugs that could be administered and will allow the user to make the decision on treatment for the patient. An on-line procedural aid is also accessible from the medical screen.

During user evaluation, the real users, in our case paramedics, follow scenarios in a test procedure. The user performance is monitored and compared with the existing system. Reports of the user trials can be found in Baber et al.³⁴ The usability and acceptability are assessed for the proposed system. Prior to evaluation, we have specified a set of criteria that can be used to evaluate the success of the application. These are shown in Table 3.

Figure 3 Initial prototype of P³Co



Figure 4 Baseline observations screen

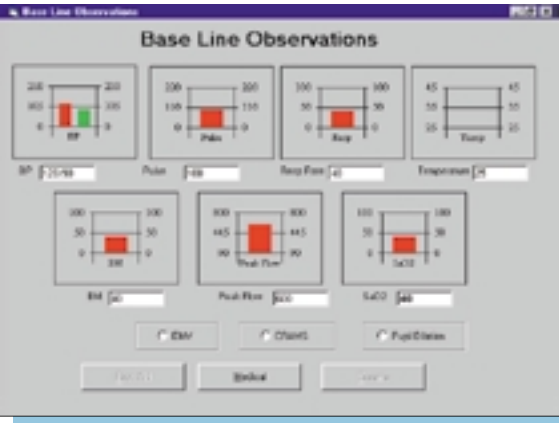


Figure 5 Medical treatment screen



Table 3 Current performance against initial acceptance criteria, as shown in Table 1

Factors	Method	Metrics	Worst	Target	Best	Current
<i>Performance measures</i>						
Task	Critical path	Time: previous	–15 percent	–5 percent	Same	–2.9 percent
	User trials	Time: previous	–15 percent	–5 percent	Same	Same
Training	1st vs 3rd trial	Percent improvement	No change	3rd > 1st	—	*
<i>Usability</i>						
User evaluation	SUS	Scale (0–100)	50	60	70	62.5
	SUMI	Scale (0–100)	50	60	70	*
	Heuristics	Scale (0–10)	<6	6	>6	7

*Data not yet fully analyzed

Example: HotHelmet. We will simply discuss the final design concept of this product, shown in the second sidebar. From initial discussions with fire fighters, one of the proposed products was a device that would inform the fire fighter of the ambient temperature in a burning building; with current protective clothing, fire fighters can move unknowingly into areas of high temperature.

The number of design options here is eight ($2 \times 1 \times 2 \times 1 \times 2$). However, closer inspection of the modality matrix suggests that at least two of the activities require concurrent manual and visual activity. Thus, the final number is only two ($1 \times 1 \times 1 \times 1 \times 2$). The type of activity indicates that the device would support situation awareness but would not require any data entry (nor would it require any interaction with the fire fighter). Consequently, the device would need to sample from the surrounding environment and then present an indication of temperature. Further discussion led us to conclude that the absolute temperature was not important, i.e., it was more important to signal that specific thresholds had been exceeded rather than provide a continuous indication of temperature. The resulting design has a temperature sensor (currently an inexpensive thermistor) connected to a microcontroller (currently a PIC16C84 model from Microchip Technology), with a single, three-color LED (light-emitting diode) display for output (the display's color changes relative to the excitation voltage). The presence of the microcontroller, as a means of comparing thermistor output with reference levels, makes this a very limited-function computer. The advantage of this setup is that it all fits inside the helmet, and the LED is attached inside the fire fighter's visor. There are several manufacturers of breathing apparatus, e.g., Drägerwerk, AG and Interspiro, Ltd. who have been experimenting with using microcontrollers as part of the air supply monitoring process. Other manufacturers, e.g.,

Fire Service Telemetry, are investigating the use of microcontrollers to provide timing information (e.g., countdown to evacuation) and biometric data (e.g., heart rate, oxygen consumption, etc.) and integrating this kit into the communications systems (e.g., to send information concerning an individual).

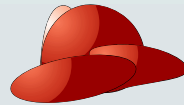
Discussion. The methodology described here not only considers the user, by taking a scenario-based design approach, and the flow of information, by using object-oriented design, it also considers the modality of the system, which is vital to wearable computer design.

For microprocessor-oriented applications, such as the paramedic system, the objects, attributes, and operations can easily map onto a standard desktop machine, whether it is mobile or desk-bound, by using application development tools, such as Visual Basic. However, microcontroller-oriented applications, such as the fire-fighter system, require greater interaction with the external environment. The object model therefore has to be refined to include sources of input and output to ascertain whether the model is technically feasible or desirable. For example, a fire fighter may have the attributes described in Table 4.

Considering the likely user of the information, e.g., the individual fire fighter or access control officer outside the building, determines whether information needs to be presented to the wearer directly or via mobile communications. Furthermore, asking what the wearer will do with the information tells us whether digits or graphics are needed or whether a single indicator will suffice. In this example, an indication that a value is at or above threshold could be presented using a LED display. A countdown (coupling rate of oxygen consumption to the amount of

1. *General requirement:* FHS1: Present salient information in a clear, easily perceptible format to the wearer.

2. *Application type:* Situation awareness

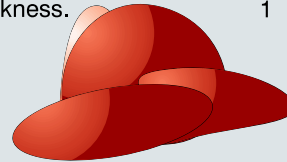


3. *Context of use:* A fire fighter, wearing breathing apparatus, is working with a “buddy” in a smoke-filled building. They are searching for possible casualties, following standard search procedures. As they move through the building, the fire fighter lays a line to aid their return. Communications between the fire fighters are largely by hand signals, but they have radio communications with the access control officer outside the building.

4. *Specific requirements:*

Ranking:

FHS1.1: The fire fighter will not need to concentrate on using the computer.	1
FHS1.2: The information will be easy to assimilate quickly.	4
FHS1.3: The fire fighter will not need to enter data or otherwise interact with the computer.	2
FHS1.4: The system will be usable in extremes of heat, noise, and darkness.	1



5. *Modality matrix:*

Activity	Information Source	Modality
Move through building	Immediate visual environment, together with feeling floor, walls, etc.	Manual/visual
Lay trail	Line laid	Manual
Search for victims	Search pattern, together with looking and feeling in area	Manual/visual
Check air supply	Raise gauge to visual field	Visual
Communicate with others	Speak on radio, sign to buddy	Speech/manual

air in a breathing apparatus) could be presented using, say, six LED characters in a row, each of which flicks off at a defined level of air in the tank. For the fire-fighter application, it is necessary to have multiple inputs to a computer with the ability to process each data stream. The final choice of which type of

processor to use depends on the application domain and information requirements. However, the point of the discussion in this paper is that wearable computers need not simply be mini-PCs, but could involve all manner of new microprocessor- or microcontroller-based products.

Table 4 Self attributes for a fire fighter

Attributes	Input to Processor	Display Parameters	Output to Location
Heart rate	Chest electrode	Threshold	Self
Body temperature	Aural thermometer	Digit/Graph	Other
Oxygen consumption	Spirometer	Countdown	Self
Blood oxygen level	Oximeter	Threshold/Digit	Other

Table 5 Example

x	0	1	2	3	4	5	6	7	8	9	10
n											
5	0.031	0.188	0.5	0.812	0.969	1					
6	0.018	0.109	0.344	0.656	0.891	0.984	1				
7	0.008	0.062	0.227	0.5	0.773	0.938	0.992	1			
8	0.004	0.035	0.145	0.363	0.637	0.855	0.965	0.996	1		
9	0.002	0.02	0.09	0.254	0.5	0.746	0.91	0.98	0.998	1	
10	0.001	0.011	0.055	0.172	0.377	0.623	0.828	0.945	0.989	0.999	1

Acknowledgment

This work is supported in part by the Engineering and Physical Sciences Research Center (EPSRC) grant GR/L48508 ("Human Factors of Wearable Computers").

Appendix: Application of binomial theorem

In order to define a majority of items in a way that is statistically significant, we employ a simple application of the binomial theorem proposed by Baber.³⁵ In this application, a significant majority is defined from a table (see Table 5). If we want to ensure that the majority reaches statistical significance, we can set a value of α (potential statistical error, or p) to be 5 percent. This means that we will select items with a table entry of less than 0.05. Assume the number of items (n) is ten. Reading from Row 10 of Table 5, we see two values that meet the criteria: 0.001 and 0.011. These correspond to x values of 0 and 1. Thus, for $n = 10$, a significant majority ($p < 0.05$) would be ten ($10 - 0$) or nine ($10 - 1$) items.

**Trademark or registered trademark of Microchip Technology, Inc., Intel Corporation, or Microsoft Corporation.

Cited references and note

1. J. Eyre and J. Bier, "DSP Processors Hit the Mainstream," *IEEE Computer* **31**, No. 8, 51–59 (August 1998).
2. S. Mok, R. Reinschmidt, and L. Smith, "Designing and Packaging of a Pentium-Processor-Based MCM-D Module for Wearable Personal Computers, Notebooks and Embedded Control Applications," *IEEE International Conference on Multichip Modules*, IEEE Press, New York (1997), pp. 208–213.

3. C. H. Amom, E. R. Egan, A. Smailagic, and D. P. Siewiorek, "Thermal Management and Concurrent System Design of a Wearable Multicomputer," *IEEE Transactions on Technology* **20**, 128–137 (1997).
4. In 1965 when preparing a talk, Gordon Moore noticed that up to that time microchip capacity seemed to double each year. In the last few years the definition has changed (with Gordon Moore's approval) to reflect that the doubling has slowed somewhat—to about every 18 months.
5. J. Villasenor and W. H. Mangione-Smith, "Configurable Computing," *Scientific American*, 66–71 (June 1997).
6. J. Underkoffler, "Antiseditary Beigeless Computing," *Personal Technologies* **1**, No. 1, 28–40 (1997).
7. S. Mann, "Smart Clothing: The Wearable Computer and WearCam," *Personal Technologies* **1**, No. 1, 21–26 (1997).
8. B. Rhodes and T. Starner, "Remembrance Agent: A Continuously Running Automated Information Retrieval System," *Proceedings of the First International Conference on the Practical Applications of Intelligent Agents and Multi-Agent Technology*, London (April 1996), pp. 487–495.
9. N. Sawhney and C. Schmandt, "Speaking and Listening on the Run: Design for Wearable Audio Computing," *Proceedings of the Second International Symposium on Wearable Computing (ISWC '98)*, Pittsburgh, PA (October 19–20, 1998), pp. 108–115.
10. M. Lamming and M. Flynn, "Forget-Me-Not: Intimate Computing in Support of Human Memory," *Proceedings of FRIEND21: International Symposium on Next Generation Human Interfaces*, Tokyo, Japan (February 2–4, 1994), pp. 125–128.
11. C. Baber, D. Haniff, L. Cooper, J. Knight, and B. A. Mellor, "Preliminary Investigations into the Human Factors of Wearable Computers," *People and Computers XIII*, H. Johnson, L. Nigay, and C. R. Roast, Editors, Springer-Verlag, Inc., Berlin (1998), pp. 313–326.
12. C. Esposito, "Wearable Computers: Field Test Observations and System Design Guidelines," *Personal Technologies* **1**, No. 2, 81–87 (1997).
13. L. Bass, C. Kasabach, R. Martin, D. Siewiorek, A. Smailagic, and J. Stivoric, "The Design of a Wearable Computer," *Pro-*

- ceedings of CHI '97, ACM Press, New York (1997), pp. 139–146.
14. A. Webster, S. Feiner, B. MacIntyre, W. Mussie, and T. Krueger, "Augmented Reality in Architectural Construction, Inspection and Renovation," *Proceedings of the ASCE Third Congress in Civil Engineering*, Anaheim, CA (June 17–19, 1996), pp. 913–919.
 15. G. Abowd, A. K. Dey, R. Orr, and J. Brotherton, "Context Awareness in Wearable and Ubiquitous Computing," *Digest of Papers for the First International Symposium on Wearable Computers*, IEEE Computer Society Press, New York (1997), pp. 179–180.
 16. S. Feiner, B. MacIntyre, M. Haupt, and E. Solomon, "Windows on the World: 2D Windows for 3D Augmented Reality," *Proceedings of User Interface Software Technology (UIST '93)*, Atlanta, GA (November 3–9, 1993), pp. 145–155.
 17. J. M. Loomis, R. G. Golledge, and R. L. Klatzky, "Navigation System for the Blind: Auditory Display Modes and Guidance," *Presence* 7, No. 2, 204–218 (1998).
 18. R. Picard, *Affective Computing*, MIT Press, Cambridge, MA (1997).
 19. S. Mann, "Wearable Computing: A First Step Toward Personal Imaging," *Computer* 30, No. 2, 25–32 (1997).
 20. T. Starner, S. Mann, B. Rhodes, J. Levine, J. Healey, D. Kirsch, R. Picard, and A. Pentland, "Augmented Reality Through Wearable Computing," *Presence* 6, No. 4, 386–398 (1997).
 21. M. Billinghurst, S. Weghorst, and T. Furness III, "Shared Space: An Augmented Reality Approach for Computer Supported Collaborative Work," *Virtual Reality* 3, No. 1, 25–46 (1998).
 22. S. Feiner, B. MacIntyre, and D. Seligmann, "Knowledge-Based Augmented Reality," *Communications of the ACM* 36, No. 7, 53–62 (July 1993).
 23. L. Foner, "Artificial Synesthesia via Sonification: A Wearable Augmented Sensory System," *Digest of Papers for the First International Symposium on Wearable Computers*, IEEE Computer Society Press, New York (1997), pp. 156–157.
 24. K. Nagao and J. Rekimoto, "Ubiquitous Talker: Spoken Language Interaction with Real World Objects," *Proceedings of the 14th International Joint Conference on Artificial Intelligence (IJCAI-95)*, Volume 2 (1995), pp. 1284–1290.
 25. T. Jebara, B. Schiele, N. Oliver, and A. Pentland, "DyPERS: Dynamic Personal Enhanced Reality System," *Proceedings of the 1998 Image Understanding Workshop*, Monterey, CA (November 20–23, 1998), pp. 1043–1048.
 26. D. A. Norman, *The Invisible Computer*, MIT Press, Cambridge, MA (1998).
 27. T. K. Landauer, *The Trouble with Computers*, MIT Press, Cambridge, MA (1995).
 28. J. A. Carroll, *Scenario-Based Design: Envisioning Work and Technology in System Development*, John Wiley & Sons, Inc., New York (1995).
 29. J. Brooke, "SUS: A 'Quick and Dirty' Usability Scale," *Usability Evaluation in Industry*, P. W. Jordan, B. Thomas, B. A. Weerdmeester, and I. L. McClelland, Editors, Taylor & Francis, Inc., London (1996), pp. 189–194.
 30. J. Kirakowski, "The Software Usability Measurement Inventory: Background and Usage," *Usability Evaluation in Industry*, P. W. Jordan, B. Thomas, B. A. Weerdmeester, and I. L. McClelland, Editors, Taylor & Francis, Inc., London (1996), pp. 169–177.
 31. J. Nielsen, *Usability Engineering*, Academic Press, Inc., New York (1993).
 32. J. M. Coble, J. Karat, and M. G. Kahn, "Maintaining Focus on User Requirements Throughout the Development of Clinical Work Station Software," *Proceedings of CHI '97*, ACM Press, New York (1997), pp. 170–177.
 33. G. Booch, *Object-Oriented Analysis and Design with Applications*, The Benjamin/Cummings Publishing Co., Redwood City, CA (1994).
 34. C. Baber, T. S. Arvantis, D. Haniff, and R. Buckley, "A Wearable Computer for Paramedics: Initial Design Studies," accepted for *Interact '99, 7th IFIP Conference on Human-Computer Interaction*, Edinburgh (August 30–September 3, 1999).
 35. C. Baber, "Repertory Grid Theory and Its Application to Product Evaluation," *Usability Evaluation in Industry*, P. W. Jordan, B. Thomas, B. A. Weerdmeester, and I. L. McClelland, Editors, Taylor & Francis, Inc., London (1996), pp. 157–166.

Accepted for publication April 26, 1999.

Chris Baber *University of Birmingham, School of Electronic and Electrical Engineering, Edgbaston, Birmingham, B15 2TT, United Kingdom (electronic mail: BABERC@novell2.bham.ac.uk).* Dr. Baber holds a Ph.D. degree in interactive speech technology. In 1990 he was appointed lecturer in ergonomics in the School of Manufacturing and Mechanical Engineering at the University of Birmingham, and he has recently joined the School of Electronic and Electrical Engineering. His principal interests involve human interaction with technology, particularly through the medium of speech recognition. He has authored two books and published more than 25 refereed journal papers.

David J. Haniff *University of Birmingham, School of Manufacturing and Mechanical Engineering, Edgbaston, Birmingham, B15 2TT, United Kingdom (electronic mail: D.J.HANIFF@bham.ac.uk).* Mr. Haniff holds a B.Sc.(Honors) degree in computer science and an M.Sc. degree in cognitive science. He is currently employed as a research associate at the University of Birmingham, working on the development of applications for wearable computers. He is also registered for a Ph.D. degree in augmented reality.

Sandra I. Woolley *University of Birmingham, School of Electronic and Electrical Engineering, Edgbaston, Birmingham, B15 2TT, United Kingdom (electronic mail: S.I. Woolley@bham.ac.uk).* Dr. Woolley holds a Ph.D. degree in electrical engineering and lectures at the School of Electronic and Electrical Engineering at the University of Birmingham. Her teaching and research interests include microprocessing architectures and applications, data compression, and digital imaging. She has previously worked for Lucas Aerospace, British Gas Plc., and the United States Department of Commerce.

Reprint Order No. G321-5705.