Situation Aware Computing with Wearable Computers

Bernt Schiele, Thad Starner, Brad Rhodes, Brian Clarkson, and Alex Pentland
{bernt,testarne,rhodes,clarkson,sandy}@media.mit.edu
Media Laboratory, Massachusetts Institute of Technology
Cambridge, MA 02139

June 11, 1999

1 Motivation for contextual aware computing

For most computer systems, even virtual reality systems, sensing techniques are a means of getting input directly from the user. However, wearable sensors and computers offer a unique opportunity to re-direct sensing technology towards recovering more general user context. Wearable computers have the potential to "see" as the user sees, "hear" as the user hears, and experience the life of the user in a "first-person" sense. This increase in contextual and user information may lead to more intelligent and fluid interfaces that use the physical world as part of the interface.

Wearable computers are excellent platforms for contextually aware applications, but these applications are also necessary to use wearables to their fullest. Wearables are more than just highly portable computers, they perform useful work even while the wearer isn't directly interacting with the system. In such environments the user needs to concentrate on his environment, not on the computer interface, so the wearable needs to use information from the wearer's context to be the least distracting. For example, imagine an interface which is aware of the user's location: while being in the subway, the system might alert him with a spoken summary of an e-mail. However, during a conversation the wearable computer may present the name of a potential caller unobtrusively in the user's head-up display, or simply forward the call to voicemail.

The importance of context in communication and interface can not be overstated. Physical environment, time of day, mental state, and the model each conversant has of the other participants can be critical in conveying necessary information and mood. An anecdote from Nicholas Negroponte's book "Being Digital" [Negroponte, 1995] illustrates this point:

> Before dinner, we walked around Mr. Shikanai's famous outdoor art collection, which during the daytime doubles as the Hakone Open Air Museum. At dinner with Mr. and Mrs. Shikanai, we were joined by Mr. Shikanai's private male secretary who, quite significantly, spoke perfect English, as the Shikanais spoke none at all. The conversation was started by Wiesner, who expressed great interest in the work by Alexander Calder and told about both MIT's and his own personal experience with that great artist. The secretary listened to the story and then translated it from beginning to end, with Mr. Shikanai listening attentively. At the end, Mr. Shikanai reflected, paused, and then looked up at us and emitted a shogun-size "Ohhhh."

> The male secretary then translated: "Mr. Shikanai says that he too is very impressed with the work of Calder and Mr. Shikanai's most recent acquisitions were under the circumstances of ..." Wait a minute. Where did all that come from?
>
> This continued for most of the meal. Wiesner would say something, it would be translated in full, and the reply would be more or less an "Ohhhh," which was then translated into a lengthy explanation. I said to myself that night, if I really want to build a personal computer, it has to be as good as Mr. Shikanai's secretary. It has to be able to expand and contract signals as a function of knowing me and my environment so intimately that I literally can be redundant on most occasions.

There are many subtleties to this story. For example, the "agent" (i.e. the secretary) *sensed* the physical location of the party and the particular object of interest, namely, the work by Calder. In addition, the agent could attend, parse, understand, and translate the English spoken by Wiesner, *augmenting* Mr. Shikanai's abilities. The agent also *predicted* what Mr. Shikanai's replies might be based on a *model* of his tastes and personal history. After Mr. Shikanai consented/specified the response "Ohhhh," the agent took an appropriate action, filling in details based on a model of Wiesner and Negroponte's interests and what they already knew. One can imagine that Mr. Shikanai's secretary uses his model of his employer to perform other functions as well. For example, he can remind Mr. Shikanai of information from past meetings or correspondences. The agent can prevent "information overload" by attending to complicated details and prioritizing information based on its relevancy. In addition, he has the knowledge and social grace to know when and how Mr. Shikanai should be interrupted for other real-time concerns such as a phone call or upcoming meeting. These kinds of interactions suggest the types of interfaces a contextually-aware computer might assume.

While the computer interface described in "Being Digital" is more of a long term goal than what can be addressed by current technology, many situationally aware applications are doable. This chapter summarizes several of the current wearable computing and augmented reality research projects at the MIT Media Laboratory that explore the dimensions of user and physical modeling. In particular, the Remembrance Agent and Augmented Reality Remembrance Agent describe applications made possible by contextual awareness. The DyPERS, Wearable Computer American Sign Language Recognizer, and DUCK! projects also have associated applications, but the emphasis for these projects is to push the sensor and context recognition technology to new limits. Finally, the Environmentally-A-Wearable project demonstrates new pattern recognition technology that can be used in the next generation of contextually aware applications. For more complete information on a particular project and related work, the reader is encouraged to refer to the original papers on these projects.

2 Remembrance Agent

The Remembrance Agent is a program that continuously "watches over the shoulder" of the wearer of a wearable computer and displays one-line summaries of notes-files, old email, papers, and other text information that might be relevant to the user's current context [Rhodes, 1997]. These summaries are listed in the bottom few lines of a head-up display, so the wearer can read the information with a quick glance. To retrieve the whole text described in a summary line, the wearer hits a quick chord on a chording keyboard.

The original RA was entirely text-based. On the input side, the user would enter or read notes, papers, email, or other text either on a wearable or a desktop computer. The RA would continuously watch a segment of the text being entered or read, and would find and suggest the "most relevant" documents from a set of pre-indexed text. Relevance was determined using text-retrieval techniques similar to those used in web search engines. While the wearable and desktop versions worked the same, the wearable version tended to allow more interactive, real-time usage. For example, someone taking notes on a conversation with the wearable RA is often able to connect the current conversation to previously taken notes, which might prompt more insightful questions.

The current version of the wearable RA still uses the text-based input, but adds many other general fields by which a current context can be described. For example, a user's context might be described by a combination of the current time of day and day of the week (provided by the wearable's system clock), location (provided by an infrared beacon in the room), who is being spoken to (provided by an active badge), and the subject of the conversation (as indicated by the notes being taken). The suggestions provided by the RA are based by a combination of all these elements.

To insure that the RA is useful in a wide variety of domains, the design makes as few assumptions as possible about the application domain. The information suggested can be any form of text or any information tagged with text, time, location, or person information. Similarly, few assumptions about the user's context are made. Often the RA could make more finely honed suggestions if a more specific domain were assumed. For example, if the RA were used by a Federal Express delivery person, many deductions could be made from routing and package information, and much more specific and potentially useful information could be suggested. However, such a deductive engine would be difficult to apply outside of the delivery domain. For the sake of making a general system, this research is attempting to push the envelope by producing as useful suggestions as possible while still making as few assumptions as possible about the application domain.

Overlay vs. Augmented Reality: The RA outputs suggestions on a head-up display (HUD), which in normal use provides some but not all features expected from an augmented reality interface. Most importantly, the HUD allows the RA to get the wearer's attention when presenting an important suggestion. This is an important distinction from a palm-top interface, where the display is only visible when the user thinks to look at it. The HUD also provides an overlay effect, where the wearer can both read the suggestion and view the real world at the same time. However, in normal use the RA does not register its annotations with specific objects or locations in the real world as one might expect from a full augmented reality system. In most cases such a "real-world fixed" display wouldn't even make sense, since suggestions often are conceptually relevant to the current situation without being relevant to a specific object or location.

2.1 Augmented Reality Remembrance Agent

One of the most distinctive advantages of wearable computing is the coupling of the virtual environment with the physical world. Thus, determining the presence and location of physical objects relative to the user is an important problem. Once an object is uniquely labeled, the user's wearable computer can note its presence or assign virtual properties to the object. Hypertext links, annotations, or Java-defined behaviors can be assigned to the object based on its physical location [Starner et al., 1997b], [Nagao and Rekimoto, 1995]. This form of

ubiquitous computing [Weiser, 1991] concentrates infrastructure mainly on the wearer as opposed to the environment, reducing costs, maintenance, and avoiding some privacy issues. Mann [Mann, 1997] argues in favor of mobile, personal audio-visual augmentation in his wearable platform.

Figure 1: Multiple graphical overlays aligned through visual tag tracking. Such techniques as shown in the following 3 images can provide a dynamic, physically-realized extension to the World Wide Web.

Objects can be identified in a number of different ways. With Radio Frequency Identification (RFID), a transmitter tag with a unique ID is attached to the object to be tracked [Hull et al., 1997]. This unique ID is sensed by special readers which can have ranges from a few inches to several miles depending on the type and size of the tag. Unfortunately, this method requires a significant amount of physical infrastructure and maintenance for placing and reading the tags.

Computer vision provides several advantages over RFID. The most obvious is to obviate the need for expensive tags for the objects to be tracked. Another advantage of computer vision is that it can adapt to different scales and ranges. For example, the same hardware and/or software may recognize a thimble or a building depending on the distance of the camera to the object. Computer vision is also directed: If the computer identifies an object, the object is known to be in the field of view of the camera. By aligning the field of view of the eye with the field of view of the camera, the computer may observe the objects that are focus of the user's attention.

Figure 2: When a tag is first located, a red arrow is used to indicate a hyperlink. If the user shows interest by staring at the object, the appropriate text labels are displayed. If the user approaches the object, 3D graphics or movie sequences are shown.

We have used computer vision identification to create a physically-based hypertext demonstration platform [Starner et al., 1997b] as shown in Figure 1. The system was later extended by Jeff Levine as part of his Master's thesis [Levine, 1997]. Even though the system requires

the processing power of an SGI, it maintains the feel of a wearable computer by sending video to and from the SGI and head-mount wirelessly. Visual "tags" uniquely identify each active object. These tags consist of two red squares bounding a pattern of green squares representing a binary number unique to that room. A similar identification system has been demonstrated by [Nagao and Rekimoto, 1995] for a tethered, hand-held system. These visual patterns are robust in the presence of similar background colors and can be distinguished from each other in the same visual field. Once an object is identified, text, graphics, or a texture mapped movie can be rendered on top of the user's visual field using a head-up display as shown in Figure 1. Since the visual tags have a known height and width, the visual tracking code can recover orientation and distance, providing 2.5D information to the graphics process. Thus, graphics objects can be rotated and zoomed to match their counterparts in the physical world. This system is used to give mini-tours of the laboratory space as shown in Figure 2. Active LED tags are shown in this sequence, though the passive tags work as well. Whenever the camera detects a tag, it renders a small red arrow on top of that object indicating a hyperlink. If the user is interested in that link and turns to see it, the object is labeled with text. Finally, if the user approaches the object, 3D graphics or a texture mapped movie are rendered on the object to demonstrate its function. Using this strategy, the user is not overwhelmed upon walking into a room but can explore interesting objects at leisure.

This physically-based hypertext system has proven very stable and intuitive to use by visitors to the laboratory. However, can this system be generalized to work without explicit tagging of objects? To answer this question, the variant of the system is being used to visually identify buildings to create an augmented reality tourist agent for downtown Boston.

Using GPS and an inertial head tracking system, strong priors can be established on what buildings may be visible using a hand-built associative model of the city. This lessens the burden of the vision system from trying to distinguish between all potential objects the tourist may see over the day to the handful that might be currently visible. In addition, the inertial tracker can be used as a means of direct control for the user. For additional information, the tourist simply stares at the building of choice. The system recognizes this lack of head motion as a fixation point and attempts identification using computer vision, conditioned on location and head orientation. Currently, the multidimensional histogram techniques developed by Schiele [Schiele, 1997] are being used to experiment with visual identification. As vision software becomes stable, separate training and test "tours" of an area of Boston will be videotaped including fixation events as described above. The stream of synchronized GPS and head tracking information will be recorded using a wearable computer. Three error measures can then be calculated. The first evaluates identification of fixation events. Next, assuming perfect recognition of fixation events, the error rate of the building recognition system will be calculated. Finally, the total combined accuracy of the system will be determined, using the same definition of accuracy as presented in the previous section. Depending on the results of these tests, experiments with additional sensors such as a sonic range to target system may be performed.

2.2 Dynamic Personal Enhanced Reality Agent

A recent extension of the above introduced augmented reality remembrance agent does not use tag but a generic object recognizer in order to identify objects in the real world. The system, called "Dynamic Personal Enhanced Reality System" (DyPERS, [Schiele et al., 1999]), retrieves 'media memories' based on associations with real objects the user encounters. These

are evoked as audio and video clips relevant for the user and overlayed on top of real objects the user encounters. The system uses an audio-visual association system with a wireless connection to a desktop computer. The user's visual and auditory scene is stored in real-time by the system (upon request) and is then associated (by user input) with a snap shot of a visual object. The object acts as a key such that when the real-time vision system detects its presence in the scene again, DyPERS plays back the appropriate audio-visual sequence.

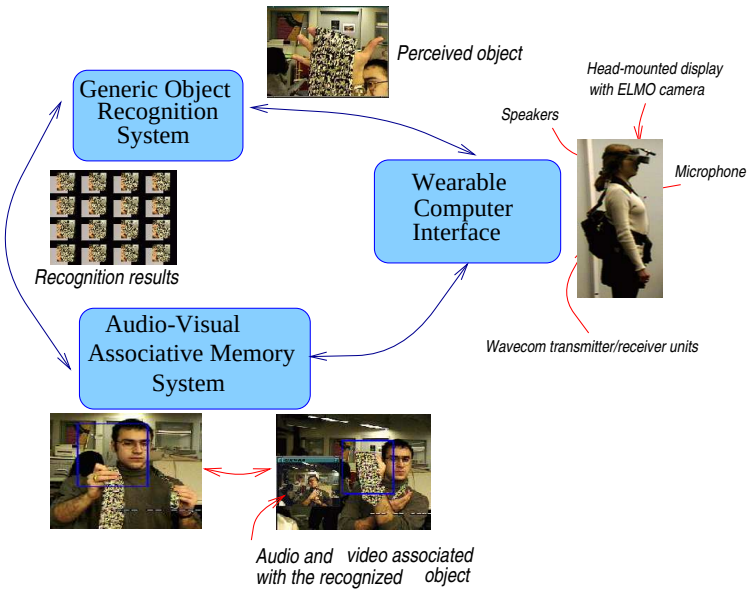

Figure 3: DyPERS's architecture

The system's building blocks are depicted in Figure 3. The audio-visual associative memory operates on a record-and-associate paradigm. Audio-visual clips are recorded by the push of a button and then associated to an object of interest. Subsequently, the audio-visual associative memory module receives object labels along with confidence levels from the object recognition system. If the confidence is high enough, it retrieves from memory the audio-visual information associated with the object the user is currently looking at and overlays this information on the user's field of view.

Whenever the user is not recording or associating, the system is continuously running in a background mode trying to find objects in the field of view which have been associated to an A/V sequence. DyPERS thus acts as a parallel perceptual remembrance agent that is constantly trying to recognize and explain – by remembering associations – what the user is paying attention to. Figure 4 depicts an example of the overlay process. Here, in the top of the figure, an 'expert' is demonstrating how to change the bag on a vacuum cleaner. The user records the process and then associates the explanation with the image of the vacuum's body. Thus, whenever the user looks at the vacuum (as in the bottom of the figure) he or she automatically sees an animation (overlaid on the left of his field of view) explaining how to change the dust bag. The recording, association and retrieval processes are all performed online in a seamless manner.

An important part of the system is the generic object recognizer, based on a probabilistic recognition system. Objects are represented by multidimensional histograms of vector

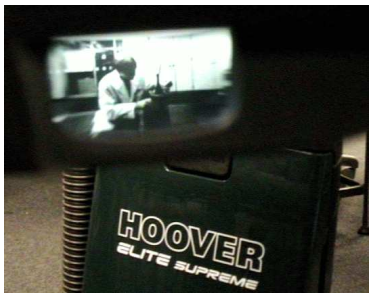

Figure 4: Sample Output Through heads-up-display (HUD)

Figure 5: A DyPERS user listening to a guide during the gallery tour

responses from local neighborhood operators. Simple matching of such histograms (using χ^2-statistics or intersection [Schiele, 1997]) can be used to determine the most probable object, independent of its position, scale and image-plane rotation. Furthermore the approach is considerably robust to view point changes. This technique has been extended to probabilistic object recognition [Schiele and Crowley, 1996], in order to determine the probability of each object in an image only based on a small image region. Experiments showed that only a small portion of the image (between 15% and 30%) is needed in order to recognize 100 objects correctly in the presence of viewpoint changes and scale changes. The recognition system runs at approximately 10Hz on a Silicon Graphics O2 machine using the OpenGL extension library for real-time image convolution.

Obviously, the discrimination of 100 objects is not enough to be of practical use in an unconstrained real world scenario. However, by using information about the physical environment, including the location of the user, the time of day and other available information, the number of possible objects can be significantly reduced. Furthermore, information about the user's current interests further reduces the number of interesting objects.

The current system has been used in a museum tour scenario: A small gallery was created using 20 poster-sized images of various famous works ranging from the early 16th century to contemporary art. Three classes of users in different interaction modes were asked to walk through the gallery while a guide was reading a script that described the paintings individually. The guide presented biographical, stylistic and other information for each of the paintings while the subjects either used DyPERS, took notes or simply listened to the

explanations. After the completion of the guide's presentation, the subjects were required to take a 20-question multiple-choice test containing one query per painting presented. The users of the DyPERS system obtained slightly better results than the other test persons, indicating the possible usefulness of such a remembrance system.

Other applications of DyPERS using the record-and-associate paradigm are the following:

- Daily scheduling and to-do list can be stored and associated with the user's watch or other personal trigger object.
- A conversation can be recorded and associated with the individual's business card.
- A teacher records names of objects in a foreign language and associates them with the visual appearance of the object. A student could then use the system to learn the foreign language.
- A story teller could read a picture book and associate each picture with its text passage. A child could then enjoy hearing the story by triggering the audio clips with different pages in the picture book.
- The system could be used for online instructions for an assembly task. An expert associates the image of the fully packaged item with animated instructions on how to open the box and lay out the components. Subsequently, when the vision system detects the components placed out as instructed, it triggers the subsequent assembly step.

Many of the listed scenarios are beyond the scope of this chapter. However, the list should convey to the reader the practical usefulness of a system such as DyPERS.

3 User-observing wearable cameras

In the previous section, head-mounted camera systems face forward, trying to observe the same region as the user's eyes to identify objects in the user's environment. However, by changing the angle of the camera to point down, the user himself can be tracked. This novel viewpoint allows the user's hands, feet, torso, and even lips to be observed without the gloves or body suits associated with virtual reality gear. The hat-mount of Figure 7 provides a surprisingly stable mounting point for the camera with a built-in reference feature: the nose. Since the nose remains stable in the same area of the image and has a known color and size, it can serve as a calibration object for observing the rest of the body. Thus, the different lighting conditions of the mobile environment can be addressed.

In the following sections, a user-observing wearable camera observes and detects the user's hand gestures in order to recognize American sign language. In section 4 a user-observing camera in conjunction with a forward pointing camera identifies the user's location as well as the user's task.

We are not aware of any other wearable computer system under development that visually observe the user's body. However, room and desktop-based camera interaction systems are more common. Of recent interest is the desk-based sign language recognizer [Vogler and Metaxas, 1998] which uses three cameras to recover 3D movement of the arms similar what is described in the following section.

3.1 A Wearable Computer American Sign Language Recognizer

Starner proposed a system [Starner, 1995] for recognizing American Sign Language using colored gloves and a camera placed in the corner of the room. Mounting the camera onto

the signer provides a unique view as shown in Figure 6. The eventual goal is to design a self-contained system into an ordinary-looking cap to (loosely) translate sign to English. The computer is installed in the back of the cap, the camera and speaker are hidden in the brim of the cap, and the head band is constructed of thin rechargeable batteries. Through the use of this wearable computer, a signer can converse with a non-signer simply by donning the cap. While it is possible to create such a cap, the current system uses a tethered SGI for analysis.

Figure 6: The baseball cap mounted camera and its perspective.

The current system tracks the signer's hands by searching the camera image for blobs matching an *a priori* model of the subject's natural skin color. Second moment analysis [Horn, 1986] is performed on the blobs which results in an eight element feature vector for each hand: position, change in position, angle, eccentricity, mass, and magnitude of the first principal component of the blob. Tracking runs at 10 frames per second. Training and recognition occur using HMM's (hidden Markov models) [Young, 1993]. The system is evaluated using conventions established by the speech recognition community. In this case, a database of 500 five word sentences created from a 40 word lexicon [Humphries et al., 1990] is randomly divided into independent training and test sets. Accuracy is determined by the equation

$$acc = \frac{N - S - D - I}{N}$$

where N is the total number of words, S is the number of words unrecognized or "substituted," D is the number of words deleted, and I is the number of words inserted. Using this measure, the system has been very successful with 98% accuracy with a grammar and 92% accuracy with no grammar. Details on this system and its evaluation can be found in a forthcoming journal publication [Starner et al., 1998].

While this wearable system is being designed to be directly controlled by the user, the environment helps prototype wearable computing equipment and demonstrate a set of tools directed at recovering user context. Specifically, complex sets of time varying signals (i.e., gestures) can be recognized from a self-observing body-mounted camera through the use of color blob analysis and HMM's. However, the user is constrained to looking straight ahead, and the system is tested and trained in the same space. What is necessary to generalize this system to identifying less constrained gestures in a mobile setting?

A current experiment, associated with the DUCK! environment below, uses the wearer's nose (as seen at the bottom of Figure 6) as a calibration object for adjusting the skin model during tracking. The nose provides a good model for the color of the user's skin and appears in a fixed place in the camera frame no matter how the user moves his head. Thus, as the user walks, the gesture tracker can continuously recalibrate to account for changes in lighting. Of course, such a tracker is subject to the caveat that it will not work in dark environments

unless a light is provided in the cap. The evaluation of this new tracking system is simple: use a video recorder to store the images from the cap camera during a normal day, run the tracker on the tapes, and count the number of "dropped" frames (not counting frames without sufficient illumination).

4 The Patrol Task

The "Patrol task" is an attempt to test techniques from the laboratory in less constrained environments. Patrol is a game played by MIT students every weekend in a campus building. The participants are divided into teams denoted by colored head bands. Each participant starts with a rubber suction dart gun and a small number of darts. The goal is to hunt the other teams. If shot with a dart, the participant removes his head band, waits for fighting to finish, and proceeds to the second floor before replacing his head band and returning.

Originally, Patrol provided an entertaining way to test the robustness of wearable computing techniques and apparatus for other projects, such as hand tracking for the sign language recognizer described above. However, it quickly became apparent that the gestures and actions in Patrol provided a relatively well defined language and goal structure in a very harsh "real-life" sensing environment. As such, Patrol became a context-sensing project within itself. The next sections discuss current work on determining player location and task using only on-body sensing apparatus.

Sensing for the Patrol task is performed by two hat-mounted wide-angle cameras (Figure 7). The larger of the two cameras points downwards to watch the hands and body. The smaller points forward to observe what the user sees. Figure 7 shows sample images from the hat. While it is possible to provide enough on-body computation to run feature detection in real-time, we currently record to video tape during the game for experimental purposes.

4.1 Location

As mentioned earlier, user location often provides valuable clues to the user's context. By gathering data over many days, the user's motions throughout the day might be modeled. This model may then be used to predict when the user will be in a certain location and for how long [Orwant, 1993].

Today, most outdoor positioning is performed in relation to the Global Positioning System (GPS). Differential systems can obtain accuracies on the order of centimeters. Current indoor systems such as active badges [Want and Hopper, 1992], [Lamming and Flynn, 1994] and beacon architectures [Long et al., 1996] [Schilit, 1995], [Starner et al., 1997a] require increased infrastructure for higher accuracy, implying increased installation and maintenance. Here, we attempt to determine location based solely on the images provided by the Patrol hat cameras, which are fixed-cost on-body equipment.

The Patrol environment consists of 14 rooms that are defined by their strategic importance to the players. The rooms' boundaries were not chosen to simplify the vision task but are based on the long standing conventions of game play. The playing areas include hallways, stairwells, classrooms, and mirror image copies of these classrooms whose similarities and "institutional" decor make the recognition task difficult.

Hidden Markov models (HMM's) were chosen to represent the environment due to their potential language structure and excellent discrimination ability for varying time domain processes. For example, rooms may have distinct regions or lighting that can be modeled by the states in an HMM. In addition, the previous known location of the user helps to limit his

Figure 7: Left: The two camera Patrol hat. Right: the downward- and forward-looking Patrol views.

current possible location. By observing the video stream over several minutes and knowing the physical layout of the building and the mean time spent in each area, many possible paths may be hypothesized and the most probable chosen based on the observed data. HMM's fully exploit these attributes. For a review of HMM's see [Rabiner, 1989].

As a first attempt, the means of the red, green, blue, and luminance pixel values of three image patches are used to construct a feature vector in real-time. One patch is taken from the image of the forward looking camera. This patch varies significantly due to the head motion of the player. The next patch represents the coloration of the floors and is derived from the downward looking camera in the area just to the front of the player and out of range of average hand and foot motion. Finally, since the nose is always in the same place relative to the downward looking camera, a patch is sampled from the nose, providing information about the lighting variations as the player moves through a room.

Approximately 45 minutes of annotated Patrol video were analyzed for this experiment (six frames per second). 24.5 minutes of video, comprising 87 area transitions, are used for training the HMMs. As part of the training, a statistical (bigram) grammar is generated. This "grammar" is used in testing to weight those rooms which are considered next based on the current hypothesized room. An independent 19.3 minutes of video, comprising 55 area transitions, are used for testing. Note that the computer must segment the video at the area transitions as well as label the areas properly.

Table 1 demonstrates the accuracies of the different methods tested. For informative purposes, accuracy rates are reported both for testing on the training data and the independent test set. Accuracy is calculated by $Acc = \frac{N-D-S-I}{N}$, where N is the total number of areas in the test set, D (deletions) is the number of area changes not detected, S (substitutions) is the number of areas falsely labeled, and I (insertions) is the number of area transitions falsely detected. Note that, since all errors are counted against the accuracy rate, it is possible to get large negative accuracies by having many insertions.

The simplest method for determining the current room is to determine the smallest Euclidean distance between a test feature vector with the means of the feature vectors comprising the different room examples in the training set. Given this nearest neighbor method as a comparison, it is easy to see how the time duration and contextual properties of the HMM's improve recognition. Testing on the independent test set shows that the best model is a 3-state HMM, which achieves 82% accuracy. In some cases accuracy on the test data is better than the training data, probably due to changing video quality from falling battery voltage.

Another important attribute is how well the system determines when the player has entered a new area. Figure 8 compares the 3-state HMM and nearest neighbor methods to the hand-labeled video. Different rooms are designated by two letter identifiers. As can be

Table 1: Patrol area recognition accuracy

method	*training set*	*test set*
2-state HMM	51.72%	21.82%
3-state HMM	68.97%	81.82%
4-state HMM	65.52%	76.36%
5-state HMM	79.31%	40.00%
Nearest Neighbor	-400%	-485.18%

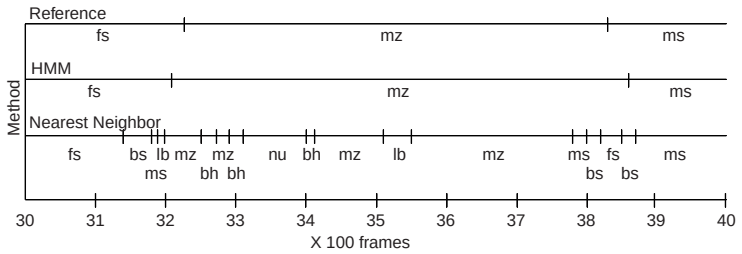

Figure 8: Typical detection of Patrol area transitions.

seen, the 3-state HMM system tends to be within a few seconds of the correct transition boundaries while the nearest neighbor system oscillates between many hypotheses.

As mentioned earlier, one of the strengths of the HMM system is that it can collect evidence over time to hypothesize the player's path through several areas. How much difference does this incorporation of context make on recognition? To determine this, the test set was segmented by hand, and each area was presented in isolation to the 3-state HMM system. At face value this should be a much easier task since the system does not have to segment the areas as well as recognize them. However, the system only achieved 49% accuracy on the test data and 78% accuracy on the training data. This result provides striking evidence of the importance of using context in this task and hints at the importance of context in other user activities.

4.2 User Tasks

By identifying the user's current task, the computer can assist actively in that task by displaying timely information or automatically reserving resources that may be needed [Feiner et al., 1993], [Starner et al., 1997b]. However, a wearable computer might also take a more passive role, simply determining the importance of potential interruptions (phone, e-mail, paging, etc.) and presenting the interruption in the most socially graceful manner possible.

Here we describe an experiment to recognize the user tasks aiming, reloading and "other" tasks. In order to recognize such user tasks we use a generic object recognition system recently proposed by Schiele and Crowley (see [Schiele and Crowley, 1996] for details). In the context of the Patrol data this system can be used for recognition of image patches that correspond to particular motions of a hand, the gun, a portion of an arm, or any part of the background. By feeding the calculated probabilities as feature vectors to a set of hidden Markov models (HMM's), it is possible to recognize different user tasks such as aiming and

reloading.

In order to use the recognition system we define a library of images grouped into images corresponding to the same action. Each image is split into 4x4 sub-images used as image patch database. In the experiment below we define three different image groups, one of each action so that the system calculates 3 groups$\times$16 = 48 probabilities at 10Hz. These probabilities are then used as feature vector for a set of HMM's trained to recognize different tasks of the user.

For two actions (aiming and reloading) we train a separate HMM containing 5 states on an annotated 2 minutes video segment containing 13 aiming actions and 6 reloading actions. Everything which is neither aiming nor reloading is modeled by a third class, the "other" class (10 sequences in total). The actions have been separated into a training set of 7 aiming actions, 4 reloading actions and 3 other sequences for training of the HMM's. Interestingly, the actions are of very different length (between 2.25sec and 0.3sec). The remaining actions have been used as test set. Table 2 shows the confusion matrix of the three action classes.

Table 2: Confusion matrix between aiming, reloading, and other tasks.

	aiming	reloading	"other"
aiming	6	0	0
reloading	0	1	1
"other"	0	1	6

Aiming is relatively distinctive with respect to reloading and "other", since the arm is stretched out during aiming, which is probably the reason for the perfect recognition of the aiming sequences. However, reloading and "other" are difficult to distinguish, since the reloading action happens only in a very small region of the image (close to the body) and is sometimes barely visible.

These preliminary results are certainly encouraging, but have been obtained for perfectly segmented data and a very small set of actions. However, an intrinsic property of HMM's is that they generalize to unsegmented data well. Furthermore the increase of the task vocabulary will enable the use of language and context models which will help the recognition of single tasks.

4.3 Use of Patrol Context

While preliminary, the systems described above suggest interesting interfaces. By using head-up displays, the players could keep track of each other's locations. A strategist can deploy the team as appropriate for maintaining territory. If aim and reload gestures are recognized for a particular player, the computer can automatically alert nearby team members for aid.

Contextual information can be used more subtly as well. For example, if the computer recognizes that its wearer is in the middle of a skirmish, it should inhibit all interruptions and information. Similarly, a simple optical flow algorithm may be used to determine when the player is scouting a new area. Again, any interruption should be inhibited. On the other hand, when the user is "resurrecting" or waiting, the computer should provide as much information as possible to prepare the user for rejoining the game.

The model created by the HMM location system above can also be used for prediction. For example, the computer can weight the importance of incoming information depending on where it believes the player will move next. An encounter among other players several rooms away may be relevant if the player is moving rapidly in that direction. In addition, if

the player is shot, the computer may predict the most likely next area for the enemy to visit and alert the player's team as appropriate. Such just-in-time information can be invaluable in such hectic situations.

5 Environmental Awareness via Audio and Video

The Environmentally-A-Wearable (EW) uses auditory and visual cues to classify the user's environmental context. Like "the fly on the wall" (except now the fly is on your shoulder) it does not try to understand in detail every event that happens around the user. Instead, EW makes general evaluations of the auditory and visual ambiance and whether a particular environment is different or similar to an environment that the user was previously in. To use sight and sound is compelling because the user and the computer can potentially share perceptions of the environment. An immediate benefit is that the user naturally anticipate what the computer can and cannot observe.

An earlier version of the system [Clarkson and Pentland, 1998a] which analysed audio alone already allowed to differentiate speech from non-speech. Such information can be used by a computer to decide if and how to present information to the user depending if the user is or is not involved in a conversation.

In order to make use of the audio-visual channel we construct detectors for specific events. Events can be simple such as a bright light and loud sounds, or more complicated such as speaker sounds and objects. Given a set of detectors higher order patterns can be observed. For example, a user's audio-visual environment can be broken into scenes (possibly overlapping) such as 'talking to a person', 'visiting the grocery store', 'walking down a busy street', or 'at the office' that are collections of specific events such as 'footsteps', 'car horns', 'crosswalks', and 'speech'. We can recognize scenes by using detectors for low-level events that make up these scenes. This identifies a natural hierarchy in a person's audio-visual environment.

5.1 Sensors

An important question is how exactly do we observe the auditory and visual environment of a person? There are many possibilities from close-talking microphones near the mouth to omni-directional microphones fixed in clothing, and from cameras in eye-glasses that see what the user sees to wide-angle cameras that try to see everything at once. Choosing the appropriate setup is crucial for success. We concluded after some experimentation that in order to adequately sample the visual and aural environment of a mobile person, the sensors should be small and have a wide field of reception. In the EW, the environmental audio was collected with a lavalier microphone (the size of a pencil eraser) mounted on the shoulder and directed away from the user. The EW's video was collected with a miniature CCD camera (1/4" diameter, 2" long) attached to the user's back (pointing backwards). The camera was fitted with a 180° wide-angle lens giving an excellent view of the sky, ground, and horizon at all times. (see Figure 9)

We chose features that are robust to errant noise like people passing by or small wavering in audio frequency. We want our features to respond only to obvious events such as walking into a building, crossing the street, and riding an elevator. Both the video and the audio features were calculated at a rate of 10Hz which is much faster than the rate at which the user's environment changes. This oversampling is advantageous for learning because it provides more data with which to make robust models.

Figure 9: The eyes and ears of the Environmentally-A-Wearable

Video: The visual field of the camera was divided into 9 regions that correspond strongly to direction. From the pixel values (r, g, b) we calculate the luminance $I = r+g+b$ as well as the chromatic channels $I_r = r/I$ and $I_g = g/I$. For each of the 9 regions we calculate 9 features including the three means and the 6 distinct covariances of I, I_r and I_g. Hence, we are collapsing each region to a Gaussian in color space. This approximation lends robustness to small changes in the visual field, such as distant moving objects and small camera movements.

Audio: Auditory features were extracted with 25 Mel-scaled filter banks. The triangle filters of the Mel-scaling give the same robustness to small variations in frequency (especially high frequencies), not to mention warping frequencies to a more perceptually meaningful scale.

5.2 From Events to Scenes

An individual's audio-visual environment is rich with repetition (going to work every morning, the weekly visit to the grocery store) that we can use to extract models for these situations. This means we do not need to make assumptions about which events can occur. We let the data speak for itself by finding similar temporal patterns in the audio-visual data. This can be contrasted with the label-train-recognize approach taken by the speech recognition community.

The audio-visual data has its own set of units, which we call events, that are obtained by clustering the audio-video features in time. The EW clusters similar sequences of features into Hidden Markov Models (HMMs). These HMMs which correspond to events are later used to model scenes [Clarkson and Pentland, 1998b, Rabiner, 1989]. The HMM clustering algorithm can be directed to model the time-series at varying time scales. Hence, by changing the time-scale and repeating the clustering, we can build a hierarchy of events where each level of the hierarchy has a coarser time scale than the one below it.

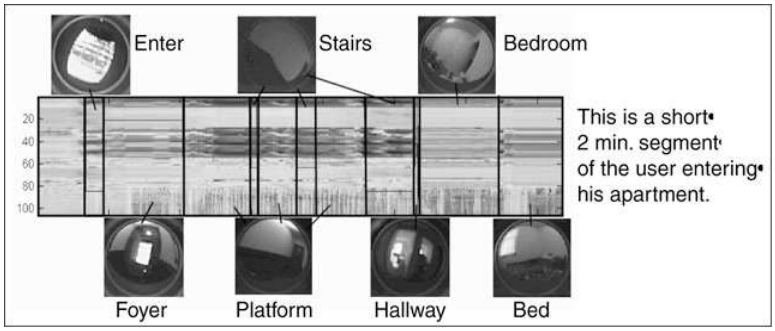

Figure 10: Coming Home: this example shows the user entering his apartment building, going up 3 stair cases and arriving in his bedroom. The system's segmentation is depicted by the vertical lines along with key frames.

For example, when we used a 3 sec time-scale for each event HMM, the emergent events are things like closing doors, walking up stairs, and crosswalks. A specific example of the user arriving at his apartment building is shown in figure 10. The figure shows the features (in the middle), segmentation (as dark vertical lines), and key frames for the sequence of events in question. The image in the middle represents the raw feature vectors (top 81 are video, bottom 25 are audio). Notice that you can see the sound of the user's steps in the audio features as vertical stripes (since audio features are just a form of spectrogram).

The EW takes these extracted events and learns their correlations in time. This allows the wearable to learn to recognize groups of events, which we call scenes. For example, suppose we wanted a model for a supermarket visit, or a walk down a busy street. The event clustering finds specific events like supermarket music, cash register beeps, walking through aisles, for the supermarket, and cars passing, crosswalks, and sidewalks for the busy street. By simply clustering raw audio-video features the system will not be able to capture the fact that events occur together to create scenes. So by clustering events themselves rather than low-level features, EW finds events which occur together and which therefore create a scene.

Figure 11 shows an example scene segmentation on roughly 2 hrs. of the user walking around the city and college campus. We evaluate performance by noting the correlation between our emergent models and a human-generated transcription. Each model plays the role of a hypothesis. A hypothesis is verified when its indexing correlates highly with a ground truth labeling. The table below shows some examples of events that matched closely with the event labeling:

Event label	office	lobby	bedroom	cashier
Correlation Coeff.	0.9124	0.7914	0.8620	0.8325

The following table gives the correlations for some example scenes that matched with the scene labeling:

Scene label	student dorms	Charles river	necco area	sidewalk	video store
Correlation Coeff.	0.8024	0.6966	0.7495	0.7804	0.9802

From these results it is clear that unsupervised clustering of audio/video data is feasible and useful. Models that correlate highly with what humans consider to be meaningful events and scenes emerge from the raw data without prior knowledge engineering. Such information can be used to provide useful context for a variety of applications. Particular interesting is the thought of learning the correlations between this environmental context and the context

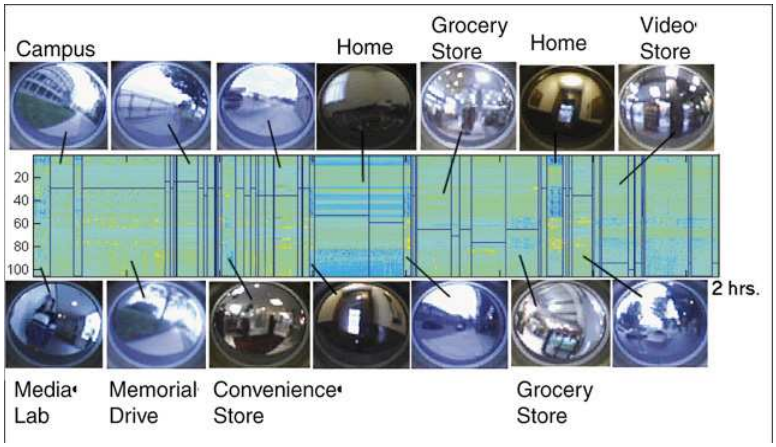

Figure 11: The Scene Segmentation: Clustering the events in time gives a higher-level segmentation of the user's audio-visual history into scenes.

from explicit interaction with the wearable (for example the Remembrance Agent).

6 Conclusion

Wearable computers offer a new opportunity to sense a user's rich environment by becoming a platform for a wide range of cameras, microphones, and other sensors. At the same time, wearables need situationally aware applications more than traditional desktop computers, because wearables often need to operate in environments where the user is engaged in tasks other than interacting with the computer. This chapter has demonstrated several systems ranging from contextually aware applications to systems that give a wearable computer a rich, high-level understanding of the wearer's environment. These systems will in the future become the basis of new contextually aware applications.

References

[Abowd et al., 1997] Abowd, G., Dey, A., Orr, R., and Brotherton, J. (1997). Context-awareness in wearable and ubiquitous computing. In *IEEE Intl. Symp. on Wearable Computers*. IEEE Computer Society.

[Azuma, 1997] Azuma, R. (1997). A survery of augmented reality. *Presence*, 6(4):355–386.

[Baum et al., 1996] Baum, W., Ettinger, G., White, S., Lozano-Pérez, T., Wells, W., and Kikinis, R. (1996). An automatic registration method for frameless stereotaxy, image guided surgery, and enhanced reality visualization. *IEEE Trans. Medical Imaging*, 15(2):129–140.

[Cho et al., 1997] Cho, Y., Park, J., and Neumann, U. (1997). Fast color fiducial detection and dynamic workspace extension in video see-through self-tracking augmented reality. In *Fifth Pacific Conference on Computer Graphics and Applications*.

[Clarkson and Pentland, 1998a] Clarkson, B. and Pentland, A. (1998a). Extracting context from environmental audio. In *Second Interl. Symposium on Wearble Computers*, pages 154–155.

[Clarkson and Pentland, 1998b] Clarkson, B. and Pentland, A. (1998b). Unsupervised clustering of ambulatory audio and video. Technical Report 471, MIT Media Lab, Perceptual Computing Group.

[Darrell et al., 1994] Darrell, T., Maes, P., Blumberg, B., and Pentland, A. (1994). A novel environment for situated vision and behavior. In *Proc. of CVPR–94 Workshop for Visual Behaviors*, pages 68–72, Seattle, Washington. alive ref.

[Feiner et al., 1997] Feiner, S., MacIntyre, B., Hollerer, T., and Webster, T. (1997). A touring machine: Prototyping 3d mobile augmented reality systems for exploring the urban environment. In *IEEE Intl. Symp. on Wearable Computers*, Cambridge, MA.

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

[Horn, 1986] Horn, B. (1986). *Robot Vision*. MIT Press, Cambridge, MA.

[Hull et al., 1997] Hull, R., Neaves, P., and Bedford-Roberts, J. (1997). Towards situaded computing. In *Proceedings of the First Intl. Symposium on Wearable Computers ISWC97*, Cambridge, MA.

[Humphries et al., 1990] Humphries, T., Padden, C., and O'Rourke, T. (1990). *A Basic Course in American Sign Language*. T. J. Publ., Inc., Silver Spring, MD.

[Ishii and Ullmer, 1997] Ishii, H. and Ullmer, B. (1997). Tangible bits: Towards seamless interfaces between people, bits and atoms. In *Human Factors in Computing Systems: CHI '97 Conference Proceedings*, pages 234–241.

[Jebara et al., 1997] Jebara, T., Eyster, C., Weaver, J., Starner, T., and Pentland, A. (1997). Stochasticks: Augmenting the billiards experience with probabilistic vision and wearable computers. In *Proceedings of the First Intl. Symposium on Wearable Computers ISWC97*, Cambridge, MA.

[Kraut et al., 1996] Kraut, R., Miller, M., and Siegel, J. (1996). Collaboration in performance of physical tasks: Effects on outcomes and communication. In *forthcoming ACM Conference on Computer Supported Cooperative Work (CSCW)*, Boston, MA.

[Lamming and Flynn, 1994] Lamming, M. and Flynn, M. (1994). Forget-me-not: Intimate computing in support of human memory. In *FRIEND21: Inter. Symp. on Next Generation Human Interface*, pages 125–128, Meguro Gajoen, Japan.

[Levine, 1997] Levine, J. (1997). Real-time target and pose recognition for 3-d graphical overlay. Master's thesis, MIT, EECS.

[Long et al., 1996] Long, S., Kooper, R., Abowd, G., and Atkeson, C. (1996). Rapid prototyping of mobile context-aware applications: The cyberguide case study. In *MobiCom*. ACM Press.

[Mann, 1997] Mann, S. (1997). Wearable computing: A first step toward personal imaging. *IEEE Computer; http://wearcam.org/ieeecomputer.htm*, 30(2).

[Nagao and Rekimoto, 1995] Nagao, K. and Rekimoto, J. (1995). Ubiquitous talker: Spoken language interaction with real world objects. In *Proc. of Inter. Joint Conf. on Artifical Intelligence (IJCAI)*, pages 1284–1290, Montreal.

[Najjar et al., 1997] Najjar, L., Thompson, C., and Ockerman, J. (1997). A wearable computer for quality assurance inspectors in a food processing plant. In *IEEE Intl. Symp. on Wearable Computers*. IEEE Computer Society.

[Negroponte, 1995] Negroponte, N. (1995). *Being Digital*. Knopf.

[Ockerman et al., 1997] Ockerman, J., Najjar, L., and Thompson, C. (1997). Wearable computers for performance support. In *IEEE Intl. Symp. on Wearable Computers*. IEEE Computer Society.

[Orwant, 1993] Orwant, J. (1993). Doppelganger goes to school: Machine learning for user modeling. Master's thesis, MIT, Media Laboratory.

[Rabiner, 1989] Rabiner, L. (1989). A tutorial on hidden markov models and selected applications in speech recognition. *Proceedings of the IEEE*, 77(2):257–286.

[Rekimoto and Nagao, 1995] Rekimoto, J. and Nagao, K. (1995). The world through the computer: computer augmented interaction with real world environments. *UIST'95*, pages 29–36.

[Rhodes, 1997] Rhodes, B. (1997). The wearable Remembrance Agent: A system for augmenting memory. *Personal Technologies*, 1(1).

[Rhodes and Starner, 1996] Rhodes, B. and Starner, T. (1996). Remembrance agent: a continuously running automated information retrieval system. In *Proceedings of the First International Conference on the Practical Application of Intelligent Agents and Multi Agent Technology (PAAM '96)*, pages 487–495.

[Sawhney and Schmandt, 1998] Sawhney, N. and Schmandt, C. (1998). Speaking and listening on the run: Design for wearable audio computing. In *IEEE Intl. Symp. on Wearable Computers*.

[Schiele, 1997] Schiele, B. (1997). *Object Recognition using Multidimensional Receptive Field Histograms*. PhD thesis, I.N.P.Grenoble. English translation.

[Schiele and Crowley, 1996] Schiele, B. and Crowley, J. (1996). Probabilistic object recognition using multidimensional receptive field histograms. In *International Conf. on Pat. Rec.*, volume B, pages 50–54.

[Schiele et al., 1999] Schiele, B., Oliver, N., Jebara, T., and Pentland, A. (1999). An interactive computer vision system, dypers: dynamic and personal enhanced reality system. In *Intl. Conference on Computer Vision Systems*.

[Schilit, 1995] Schilit, W. (1995). *System architecture for context-aware mobile computing*. PhD thesis, Columbia University.

[Schmandt, 1994] Schmandt, C. (1994). *Voice Communication with Computers*. Van Nostrand Reinhold, New York.

[Sharma and Molineros, 1997] Sharma, R. and Molineros, J. (1997). Computer vision-based augmented reality for guiding manual assembly. *Presence*, 6(3).

[Smailagic and Martin, 1997] Smailagic, A. and Martin, R. (1997). Metronaut: A wearable computer with sensing and global communication campabilities. In *IEEE Intl. Symp. on Wearable Computers*. IEEE Computer Society Press.

[Smailagic and Siewiorek, 1994] Smailagic, A. and Siewiorek, D. (1994). The cmu mobile computers: A new generation of computer systems. In *COMPCON '94*. IEEE Computer Society Press.

[Starner, 1995] Starner, T. (1995). Visual recognition of American Sign Language using hidden Markov models. Master's thesis, MIT, Media Laboratory.

[Starner et al., 1997a] Starner, T., Kirsch, D., and Assefa, S. (1997a). The locust swarm: An environmentally-powered, networkless location and messaging system. Technical Report 431, MIT Media Lab, Perceptual Computing Group. Presented ISWC'97.

[Starner et al., 1997b] Starner, T., Mann, S., Rhodes, B., Levine, J., Healey, J., Kirsch, D., Picard, R., and Pentland, A. (1997b). Augmented reality through wearable computing. *Presence*, 6(4):386–398.

[Starner et al., 1998] Starner, T., Weaver, J., and Pentland, A. (To appear 1998.). Real-time American Sign Language recognition using desk and wearable computer-based video. *IEEE Trans. Patt. Analy. and Mach. Intell.*

[Uenohara and Kanade, 1994] Uenohara, M. and Kanade, T. (1994). Vision-based object registration for real-time image overlay. Technical report, Carnegie Mellon University.

[Vogler and Metaxas, 1998] Vogler, C. and Metaxas, D. (1998). ASL recognition based on a coupling between HMMs and 3D motion analysis. In *ICCV*, Bombay.

[Want and Hopper, 1992] Want, R. and Hopper, A. (1992). Active badges and personal interactive computing objects. *IEEE Trans. on Consumer Electronics*, 38(1):10–20.

[Weiser, 1991] Weiser, M. (1991). The computer for the 21st century. *Scientific American*, 265(3):94–104.

[Wren et al., 1997] Wren, C., Azarbayejani, A., Darrell, T., and Pentland, A. (1997). Pfinder: Real-time tracking of the human body. *IEEE Trans. Patt. Analy. and Mach. Intell.*, 19(7):780–785.

[Young, 1993] Young, S. (1993). *HTK: Hidden Markov Model Toolkit V1.5*. Cambridge Univ. Eng. Dept. Speech Group and Entropic Research Lab. Inc., Washington DC.