APPLIED RESEARCH

Asst Technol 2001;13:23–39

The Use of Artificial Intelligence in the Design of an Intelligent Cognitive Orthosis for People with Dementia

*†Alex Mihailidis, M.A.Sc., P.Eng., *Geoffrey R. Fernie, Ph.D., P.Eng., and †Joseph C. Barbenel, Ph.D., C.Eng.

**Centre for Studies in Aging, Sunnybrook and Women's College Health Sciences Centre, Toronto, Ontario, Canada; and †Bioengineering Unit, University of Strathclyde, Glasgow, Scotland, United Kingdom*

Dementia often reduces a person's ability to perform activities of daily living because he or she becomes confused and cannot remember the sequence of steps to perform. The current solution is to have a caregiver continually supervise and assist the person using verbal reminders or cues. This loss of privacy and increased dependency may cause the affected person to become embarrassed and agitated. We propose that this situation might be improved by using a computerized device that monitors progress and provides the reminders needed. The COACH is a first prototype of such a device. It uses artificial intelligence to observe a user, learn from his or her actions, and issue prerecorded cues of varying detail. The device was developed using a personal computer and a video camera that unobtrusively tracked the user. Preliminary testing with subjects who simulated confused behavior as they washed their hands showed that the device was performing its functions with an efficacy of approximately 95%. More extensive clinical testing and evaluation has begun, and the results will be reported in future publications.

Key Words: Cueing—Artificial intelligence—Activities of daily living—Dementia—Cognitive—Orthosis.

The COACH—Cognitive Orthosis for Assisting aCtivities in the Home—is a prototype of an intelligent cognitive orthosis being developed to assist people with dementia complete activities of daily living (ADL) with less dependence on a caregiver. It was developed using artificial intelligence to observe a user and his or her environment, learn from his or her actions, automatically adapt to individual preferences, and issue prerecorded verbal cues when necessary.

The first prototype has been completed, and preliminary results have been obtained with more in-depth clinical evaluations to follow. This article describes the rationale for this device, its design, and the results of preliminary trials.

Address correspondence and reprint requests to Mr. Alex Mihailidis, Centre for Studies in Aging, Sunnybrook and Women's College Health Sciences Centre, 2075 Bayview Avenue, Toronto (Ontario) M4N 3M5, Canada.

Why Are Cognitive Devices Needed?

Dementia can be defined as a condition of acquired cognitive deficits, sufficient to interfere with social or occupational functioning in a person (Patterson, 1999). It is estimated in Canada that a third of people over the age of 85 have dementia, with this number increasing to nearly half in the United States. Over the next 30 years, as the population ages, it is expected that the prevalence of dementia will triple (American Alzheimer Association, 1998; Health & Welfare Canada, 1991).

Older persons need to feel in control of their environment and activities of daily living (ADL) for their sense of well being and continued intellectual stimulation (Chute & Bliss, 1988, 1994). However, because of the effects of dementia, this sense of control can be lost. Dementia may reduce a person's ability to independently perform ADL tasks such as washing or grooming because he or she cannot remember the sequence of steps required during the activity. This confusion often results in mishaps such as not dressing properly or not using the toilet correctly. The current solution is to have a caregiver, often a family member, supervise and provide verbal reminders of the sequence required for successful task completion. This may result in a loss of dignity and may cause embarrassment and agitation for everyone involved (Cockburn & Collin, 1988; Harrell, Parente, Bellingrath, & Lisicia, 1992). This loss of autonomy can be especially upsetting to family caregivers when they must help with toilet-related activities.

Targeted interventions, task analysis (i.e., subdividing an ADL task into individual steps), and cueing have been used for several years in an attempt to ease the stress and difficulty of caring for a person with dementia or other cognitive disabilities (Agostinelli, Demers, Garrigan, & Waszynski, 1994; Giles, Ridle, Dill, & Frye, 1997). However, these and other current techniques still require a caregiver to remain in the room.

Many researchers, including the authors, have suggested that dependence on a caregiver can be reduced and dignity might be improved using an automated device that provides necessary reminders to the user and monitors his or her progress.

A Brief History of Cognitive Orthoses

Devices have been used to assist people with cognitive disabilities (such as learning disabilities and traumatic brain injury) in completion of ADL and vocational tasks for close to 20 years and have most often been referred to as cognitive orthoses or cognitive prostheses (Bergman, 1998).

At first, computer applications were developed primarily as a retraining tool. This was based on the assumption that repetitive practice would promote remediation of brain dysfunction and lead to cognitive recovery (Bergman, 1998). However, there has been little evidence that individuals with cognitive disabilities generalize from this type of practice to independent completion of ADL tasks (Chute & Bliss, 1994, 1988). Consequently, interest shifted from using computers as remediation tools to using them as compensatory tools in actual life situations, i.e., increasing a person's performance by using the computer to support areas of cognitive weaknesses. Early attempts to use the computer for this purpose were limited, but growing evidence indicates that the computer has promise (Bergman, 1998).

Previous Research Devices

Several researchers have developed prototypes of computerized devices and used them in clinical trials to guide subjects through an ADL or vocational task. Chute and Bliss (1988, 1994), Kirsch et al. (1988), Steele, Weinrich, and Carlson (1989), Kirsch, Levine, Lajiness, O'Neill, and Schneider (1992), Cavalier and Ferretti (1993), Napper and Narayan (1994), and LoPresti, Friedman, and Hages (1997) showed subjects were able to complete various tasks with less dependence on a caregiver or job coach when a computerized device was used. These examples have been described in more detail in a previous article by the authors (Mihailidis, Fernie, & Cleghorn, 2000). Only the device by LoPresti et al. used some type of automatic feedback to determine whether the user had completed the required task/step. This was accomplished using several switches and sensors that were placed on different parts of the user's work environment and on the tools that were required. These transducers were connected to a palm-top computer (LoPresti et al., 1997). These studies involved subjects who had a traumatic brain injury, some form of mental retardation, and/or a learning disability.

Only studies by the authors and Bourgeois (1990) have included subjects with dementia. Bourgeois tested subjects with Alzheimer's disease (the most prevalent dementia) but used a noncomputerized cognitive orthosis. Cue cards with various conversational phrases were placed inside a wallet to prompt a user during daily conversations (Bourgeois, 1990). In a pilot study preceding this article, the authors showed that a user with moderate to severe dementia completed an ADL task (handwashing) in response to a computerized cognitive orthosis that used a recorded voice for cueing. Similar to LoPresti et al. (1997), this device monitored which handwashing steps were being completed using several switches and sensors attached to various parts of a sink (Mihailidis et al., 2000). These two studies plus those by Chute (1994, 1988) have included older adults as participants. Each study found that the older adults were able to use the various devices to decrease their dependence on a caregiver while performing required tasks. The study by Mihailidis et al., which used older adults with moderate to severe dementia, observed that the participants did not perceive a difference between the prerecorded verbal prompts played by the computerized device and those given by a live caregiver.

Several other researchers have been using computers to monitor ADL tasks in the home. The purpose of these devices is not to help a person through a particular task but to monitor the person's actions. These devices use several sensors and switches attached to various objects and parts of the user's environment to detect which task the person is completing. If these devices detect an unexplainable change in the person's normal routine, then external assistance is called. Trials with several subjects have shown that this method of tracking a person's actions is a good way of monitoring the state of a person's health and independence (Bai, Zhang, Cui, & Zhang, 2000; Nambu, Nakajima, Kawarada, & Tamura, 2000; Ogawa, Ochiai, Shoji, Nishihara, & Togawa, 2000).

The use of cueing and verbal prompting in assisting people with cognitive disabilities to complete ADL tasks has been studied previously, and two primary types of cueing have been identified: (1) a system of least prompts (SLP) and (2) time-delayed procedures (Giles et al., 1997). The SLP, or the increasing assistance procedure, involves the presentation of a prompt hierarchy that is arranged from most general to most specific. The person is cued progressively through this hierarchy for a particular step. The time-delay procedure plays cues after a defined interval of time has elapsed after the occurrence of a stimulus. After a cue is played to a user, he or she is given a certain amount of time to complete the required step before another cue is played (Giles et al., 1997). Several studies have concentrated on the use of the time-delay procedure, with limited studies being completed using the SLP technique. The studies that were conducted using the SLP technique, however, demonstrated that this type of cueing was effective, specifically with subjects who had anoxic brain damage (Giles et al., 1997). In addition, Giles et al. (1997) showed that the time-delay procedure was effective in assisting subjects with severe brain injury to redevelop self-care skills, such as washing and dressing tasks. There have been no studies comparing SLP techniques with time-delay procedures (Giles et al., 1997). It is unknown whether these techniques have been tested with older adults.

Beck, Heacock, Rapp, and Mercer (1993) described a variety of strategies as interventions that may help increase the ability of a person with a cognitive impairment to perform ADL. Seven levels of assistance were identified: stimulus control, verbal prompting, modeling/gesturing, physical prompting, physical guidance, occasional physical guidance, and complete physical guidance (Beck et al., 1993).

Engelman, Altus, and Mathews (1999) examined the effects of caregivers using verbal prompts to increase the completion of ADL by older adults with dementia. Five residents aged 85–95 with dementia were observed twice per week in the morning or afternoon over 20 weeks. It was found that, during the baseline period, the residents demonstrated appropriate completion of ADL during 41% of the time in the morning and 31% of the time in the afternoon. During the intervention period (i.e., trials using prompting), the residents demonstrated appropriate completion of ADL during 81% of the time in the morning and 72% of the time in the afternoon. During the follow-up period, residents demonstrated appropriate completion during 83% of morning ADL and 68% of afternoon ADL. This study concluded that increasing prompts and praise by caregivers may increase the level of ADL completion of older adults with dementia (Engelman et al., 1999). The use of verbal prompting and touch was also shown to be effective in increasing the nutritional intake of several elderly residents in a long-term care facility (Lange-Alberts & Shott, 1994).

Commercial Devices

MASTERY Rehabilitation Systems, Inc., has developed a cognitive orthosis called Essential Step, which is a software package that can be modified as a user's rehabilitation progresses. This software is a compensatory system that can support users in a variety of daily tasks necessary for work, school, and daily living (Bergman, 1996). While a user is completing a task, the software provides him or her with cues and information on-screen or using a computer-generated voice. The user is required to push a function key to specify that the task has been completed (Bergman, 1996). This software was tested with 54 people having memory deficits secondary to neurological and/or developmental etiologies (e.g., traumatic brain injury). These subjects demonstrated rapid skill acquisition in individual trials with the software. It was also demonstrated that this tool was successfully integrated into ADL tasks at home, school, and work over a long period of time—the longest being 10 years (Bergman, 1997). As with the previous efficacy studies, this tool was not tested with people who have dementia.

ISAAC, by Cogent Systems, Inc., is a handheld, fully individualized cognitive device "specifically designed for individuals with a wide range of cognitive disabilities" (COGENT, 1998). It allows for the organization and delivery of individualized procedural and personal information and prompts. A rehabilitation professional initially programs the device using editing software on a standard personal computer and then downloads the information to the hand-held unit. Information and prompts can be delivered as text, checklists, graphics, or synthesized speech and are accessed by the user through a graphical user interface. Prompts to initiate a required task can also be delivered to a user based on set conditions, such as time. The user is required to respond to the cues through a pressure-sensitive touch screen (COGENT, 1998). Clinical trials with subjects have not been presented.

PEAT—The Planning and Execution Assistant and Training System by Attention Control Systems, Inc.—is a hand-held electronic calendar and address book that features automatic cueing to start and stop daily activities. PEAT uses automatic planning software to compensate for executive function deficits. It automatically generates daily plans, monitors plan execution, and replans when the situation changes using an artificial intelligence tool called PROPEL—PROgram Planning and Execution Language (Levinson, 1997). PEAT maintains a library of hierarchical programs (scripts) that describe ADL tasks. Some users rely on their caregivers to write and enter the scripts, while other users are able to independently write and enter these scripts. Using manually entered appointments and the scripts from the ADL library, PEAT generates the best plan to complete all of the required tasks and assists with plan execution by using visual and audible cues to prompt the user through each step. The user is required to provide input to the device when a task is completed or when he or she requires more time to complete the task (Levinson, 1997). Clinical trials have not been reported to date.

Drawbacks of Past Devices (Commercial and Research)

Many past devices have been unable to provide the system feedback about task completion or error occurrence automatically (Vanderheiden, 1998). A majority of previous-generation computerized orthoses (both research and commercial) relied on input from the user for feedback (e.g., pushing OK after a step). This feedback—and for some devices, the expiration of a time limit—was the only information used to determine whether corrective action or replanning was required. Such responding may be achievable for a person with a less severe cognitive disability but is less likely to be completed by a person with advanced dementia because he or she may lack the required planning and initiation skills. Such users may neither remember what step they had been asked to perform nor the need to indicate that the step had been completed (Vanderheiden, 1998). This is one of the primary reasons why many of the hand-held devices may be inappropriate for people with dementia (or other severe cognitive disabilities), even though the devices have sophisticated algorithms.

The few devices that obtained input and feedback via transducers, such as that of Mihailidis et al. (2000), LoPresti et al. (1997), and those for ADL monitoring, have been restricted in scope because of the limitations of the hardware used and the units' lack of intelligent algorithms allowing them to adapt automatically to a user's requirements. These devices used switches and sensors that detected whether a step was completed. For instance, an infrared switch could detect whether or not a user's hands were in a certain location. Even though these switches and transducers provided reliable and useful input to their devices, they did not lend themselves to being easily generalizable or adaptable to tasks other than those for which they were developed.

Many devices were expensive to develop because they required detailed user modeling and high customization for each user (Francik, 1999). Previous researchers have maintained that, for a cognitive device to be effective for a particular user, it must be developed as a one-of-a-kind device (Cole, 1999). To adapt to individual users, most devices required manual reprogramming by someone who was knowledgeable about the device and its software. Again, this level of technical sophistication cannot be expected from a person with dementia or from his or her caregiver. An effective and practical device should be able to customize essential features for a user through its own algorithms based on individual performance.

Essential Features of a Successful Cognitive Orthosis

Research conducted by Levine and Kirsch (1985) identified two features that contribute to the functionality of cognitive devices. These are that (1) the device must determine whether its user deviates from the task at any point or has done something out of sequence and (2) the device must be customized to meet individual needs, interests, and preferences. In addition, several program features, including the amount of guidance provided and the types of cues issued, need to be modified. The ability to detect errors is very important. The device must be able to determine whether its user deviates from the task at any point and then must be able to correct the error(s) (Levine & Kirsch, 1985). These features have been used by previous researchers as guidance in developing the algorithms for their cognitive orthoses and to ensure that essential features have been included in their devices. In addition to following these guidelines, however, the true functionality of any cognitive orthosis needs to be determined through clinical trials, feedback from users, and feedback from caregivers. Unfortunately, many people who have dementia will not be able to provide feedback. As a result, data collected from user trials and feedback from caregivers must be relied upon to determine device functionality.

Typically, up to 1,000 different parameters need to be customized for each user for a cognitive device to be effective. The instructions provided by the device must be adjusted on a regular basis as the user's capabilities change. Customization needs to reflect a number of factors, including (1) user priorities for activities, (2) abilities in the context of the environment where the activity is performed, (3) functional deficits that require support, and (4) features that make the system user friendly for a user who has a cognitive impairment (Cole, 1999).

For a cognitive device to be effective, it must be able to adapt automatically. The majority of earlier devices did not have this self-adaptive capability. This limitation made them difficult to set up for different users, environments, and activities and unacceptable for use by people with severe cognitive disabilities, such as dementia. However, today's software development tools and artificial intelligence algorithms have changed the way programs are designed and the functions they can perform (Cole, 1999). Artificial intelligence (AI) techniques and advanced hardware have the adaptability and the self-learning capacity to create a cognitive device that incorporates these and other important design features more effectively than in the past.

ARTIFICIAL INTELLIGENCE

Artificial intelligence techniques employ algorithms that can be used to make a computer program act more like a human when performing cognitive tasks such as decision making or planning (Russell & Norvig, 1995).

In AI, the term "agent" is used to describe something that perceives and acts in an environment (e.g., a computerized device). An agent consists of hardware and an agent program. For example, a personal computer is the hardware and the software that incorporates the AI algorithms is the agent program. The agent program is responsible for mapping a percept (information or an observation from the environment) to an action (Russell & Norvig, 1995). Although there are several different types of agents, a cognitive device is a goal-based agent because it acts to achieve a specific set of goals, such as completing an ADL.

There are many different types of AI techniques and algorithms. For the design of this device (the COACH), two techniques were applied: (1) artificial neural networks which, among other things, are used in classification, and (2) planning algorithms. A brief overview of the algorithms follows.

Artificial Neural Networks

Artificial neural networks (ANNs) attempt to model the operation of the brain using mathematics. They learn associations between input data patterns and output data patterns by first learning the correct associations for a set of training data and then applying what has been learned to new inputs. The trained ANN can then be used to classify these new data into prelearned categories (Russell & Norvig, 1995). This robust ability to classify new data makes an ANN an ideal algorithm for the development of an adaptable device. The ability to train an ANN with new data also makes it ideally suited for use in the agent program because the device can be trained for any specific ADL by simply providing it with the required data.

There are many different kinds of network structures, each with different uses and properties. The development of the COACH required a network structure that can be used for classification, such as a probabilistic neural network (PNN).

A PNN typically comprises an input level, an output level, and two hidden levels (pattern units and summation units); each level is responsible for some type of computation. It learns quickly from examples in one pass and asymptotically achieves the Bayes optimal decision boundaries, i.e., it uses the Bayes strategy to decide to which category the originally presented input belongs (Specht, 1992). The Bayes optimal decision rule states that X should be classified into population i if

$$h_i c_i f_i(X) > h_j c_j f_j(X), \qquad (1)$$

where h is the prior probability of an unknown sample being drawn from a population, c is the associated cost of misclassification, and $f(X)$ is the probability density function (PDF) (Masters, 1993). Normally, the prior probabilities are treated as being equal and can be ignored. The same is true for the costs. This leaves the problem of determining the PDF, which is typically unknown.

The PDFs of the populations can be estimated using a set of training data and Parzen's method (Parzen, 1962). This method provides a tool for constructing a Bayes optimal classifier even when the PDFs are unknown. The PDF estimator is a scaled average of that function across the training set (Masters, 1993). For a population with sample size n, the estimated density function is

$$G(X) = \frac{1}{n\sigma} \sum_{i=0}^{n-1} W \sqrt{\frac{x - x_i}{\sigma}}, \tag{2}$$

where the weighting factor, W, is a Gaussian function, and σ is a scaling parameter that controls the width of the area of influence in subsequent classifications (Masters, 1993). Using the density function, which is now known, new inputs to the ANN can be classified into their respective categories using Bayesian confidence measures represented by

$$P[S \mid X] = \frac{G_A(X)}{\sum G_K(X)}. \tag{3}$$

This model is quite specialized and unique when compared with other ANNs. As mentioned, the primary advantages of a PNN are that usually little or no training time is required and training for another ADL task is easy to accomplish. This allows for training of the network for either a new ADL task or the retraining or updating of the same one to be accomplished on-line and as close to real time as possible. At worst, σ in the denominator of equation (2) may have to be optimized to achieve the best results. Its value can be crucial to good performance of the network; a value too small will result in very specific and limited classifications, and a value too large will result in overgeneralized classifications. There is no mathematically rigorous method for choosing the best value for σ, and optimization often relies on trial and error (Masters, 1993). However, to train another type of network to do the same classification would be much more time consuming. The main drawback of using a PNN is that the training examples usually have to be highly representative of the data that will be subsequently presented to the network (Masters, 1993).

Planning

A plan is a set of steps that, when executed together, correctly accomplishes a particular goal. Using percepts from the environment, a planning agent constructs a plan that achieves its goals and then executes each of the steps in the plan (Russell & Norvig, 1995).

For the development of this device, a plan recognition algorithm and a situated planning algorithm were used. Situated planning combines a typical planning agent with an execution monitoring agent (Russell & Norvig, 1995). In this article, these algorithms will collectively be known as the plan recognition algorithm (PRA).

A user's actions are explained by a plan if the steps match those in the plan definition (Franklin, 1998). Given a set of steps, plan recognition is accomplished by finding a plan that explains the steps being completed. Once a plan has been found, it is used to guide a user through the remaining steps. A useful plan recognition agent must be able to handle various forms of uncertainty. In particular, a given set of observed actions often may not identify a particular plan uniquely, yet many important conclusions can still be drawn and predictions about future actions still be made (Kautz & Allen, 1986). Kautz and Allen (1986) developed a new view of plan recognition. They proposed that it be viewed as ordinary deductive inference based on a set of observations, an action taxonomy, and one or more simple constraints. Action taxonomy is an exhaustive description of the ways in which actions can be performed and the ways in which any action can be used as a step of a more complex action. The approach assumes that all the possible reasons for performing an action are known (Kautz & Allen, 1986). The plan recognition algorithm used in this device (as part of the PRA) is based on this approach but has been greatly simplified and implemented using a different algorithm than Kautz and Allen's.

A situated planning agent is a more comprehensive approach to plan execution because it involves incremental modifications to the plan, including execution of steps, as the environment evolves (Russell & Norvig, 1995). This execution monitoring and replanning are necessary for two reasons: (1) the user may change the sequence of the steps from those defined in the current plan, although the same outcome will be met, or (2) the user has made an error and corrective action needs to be added to the plan.

Some Previous Uses of AI

As previously described, PEAT used a sophisticated planning and replanning agent to assist its users through ADL tasks (Levinson, 1997). The Intelligent Classroom by Franklin (1998) used a combination of plan recognition, planning, and execution monitoring. This device used information obtained from the environment to infer what a speaker was trying to do and then controlled the setting of the lights, played videos, displayed slides, or did whatever was appropriate (Franklin, 1998). Mozer (1998) used ANN to develop a home that essentially programmed itself by observing the lifestyle and desires of the inhabitants and learned to anticipate and accommodate their needs. The device controlled the heating, lighting, ventilation, and water heating by using ANN and decision-making algorithms that learned when these various controls should be turned on or off, or be adjusted (Mozer, 1998). Finally, Kiana, Snijders, and Gelsema (1998) used a PNN to recognize daily life motor activity classes (such as sitting, lying, standing, walking, etc.). Training data representing these various states were presented to a PNN, and it was found that the PNN was able to classify 95% of the presented cases of daily life activities correctly (Kiani et al., 1998).

THE COACH: AN OVERVIEW

The COACH consists of hardware that tracks the actions of the user, an agent program that uses the AI techniques described earlier, plus other innovative algorithms, to analyze inputs and make intelligent decisions. The device has the potential to be used for most ADL tasks, but this article describes COACH's handwashing routine.

A video camera located within the environment, in combination with associated hardware, finds the two-dimensional (x and y) coordinates of the user's hand. These coordinates are input to the agent program. The program then analyzes this input and classifies the data into a corresponding category and assigns it a step identification number (each step in the ADL is defined by a set of coordinates corresponding to the location of the user's hand). Once the program determines the step the user is completing, it finds which plan he or she is trying to complete by conducting a search through a preexisting plan library, or action taxonomy. If the user changes the sequence of the steps required for completion while still reaching the final goal, the program will adapt itself to guide the user through the new sequence. If a match cannot be found, the program attempts to predict which plan the user is trying to complete and hence which step he or she should be doing. If the user makes an error, such as completing a wrong step or performing a step out of sequence, the program selects a prerecorded verbal cue and plays it over a set of speakers. If necessary, the device will repeat the cue after an interval. The level of description provided in the cue can be adjusted as required. If the user does not respond to any of the cues issued, the device stops and calls for a caregiver via visual and audible alarms on a graphical user interface (GUI) shown on a computer monitor outside the environment. Information about the user's progress and actions taken by the device are also displayed on the GUI.

DESIGN

The objective was to create a device that used artificial intelligence to (1) intelligently and rationally guide a person with dementia through an ADL task and (2) adapt automatically to different users and different tasks.

The Design Process

The development of a working prototype has taken approximately 2 years. Design criteria to be incorporated in this device were first developed using results and observations from previous researchers and the pilot study conducted by the authors (Mihailidis et al., 2000). Some of these criteria were

- The device must be easy to set up and use for more than one ADL and more than one user.
- It must maintain a memory of each individual's performance, settings, tendencies, and preferences.
- It must track the user in an automatic, nonobtrusive manner and it must not greatly modify the existing environment.
- It must adapt the sequence of steps to be completed for a specific ADL to meet the sequence with which the user feels most comfortable.
- Rather than having one cue available for each step, the device should be able to provide more detail on how to complete the step with each subsequent cue.
- This change in the cue detail should be reversible (i.e., the level of description provided in the cue should be reduced if the user's performance improves).
- The device should be able to skip automatically to the most detailed cue if the user is quickly declining in his or her performance.

Next, algorithms that incorporated these and other functions were developed using Matlab v.5.3 from Mathworks, and the graphical user interface for the device was developed using LabView v.5.1 from National Instruments. The final step was to develop the hardware that tracks the user within the environment and to ensure that the hardware and agent program functioned properly with each

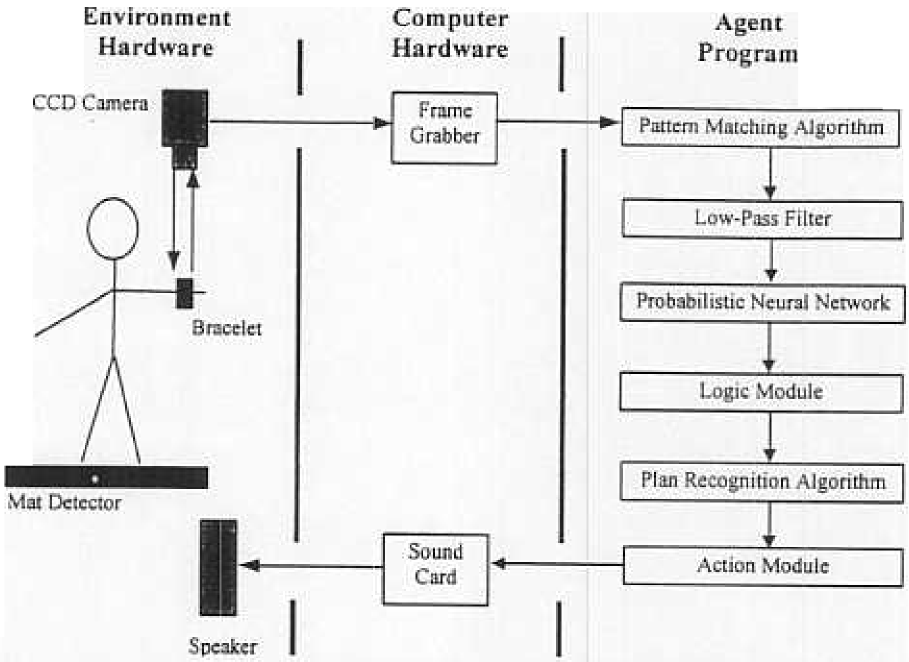

FIG. 1. The system consists of hardware located in the environment and a personal computer that runs the agent program. A CCD video camera is installed above the workplace; it tracks the position of a bracelet worn by the user. A mat is also placed on the floor and monitors whether the user is standing in front of the workplace, and a loudspeaker plays prerecorded verbal prompts to the user. The agent program consists of six algorithms that track the position of the user, determine which step he or she is completing, and provide suggestions for corrective actions.

other when incorporated into the final device. LabView v.5.1 was used to develop the controls and data-acquisition tools for the tracking hardware.

This process resulted in the overall system design shown in Figure 1. The device hardware consists of a charge-coupled device (CCD) camera, bracelet, mat detector, speakers, and a personal computer. The agent program consists of six main algorithms.

Steps and Cues

Task analysis and observations of people with dementia washing their hands were used to subdivide this activity into six steps: (1) turning the water on, (2) rinsing hands, (3) using the soap dispenser (this dispenser was typical of those used by the patients on the Cognitive Support Unit at Sunnybrook & Women's College Health Sciences Centre, the primary site for future clinical testing), (4) rinsing off soap, (5) turning the water off, and (6) using the towel. If a user requires these steps to be broken down further, additional steps can be added by adjusting the preexisting plan library and recording the extra verbal cues. A new plan can be added quickly on-line by a caregiver or clinician by manually retraining the device using a series of prompts presented by the device on the graphical user interface (GUI). These prompts allow the device to determine the required new step and how this new step fits into the existing sequences of handwashing steps. We are planning to automate this feature in future prototypes by applying learning algorithms similar to those used to teach computers to play games, such as electronic chess.

The COACH uses verbal and simple visual cues to provide assistance to a user. Simple visual cueing is used in conjunction with the verbal prompts as recommended by Beck (1993). The cold and hot water taps were painted different colors (the cold tap was painted blue and the hot tap was painted red), and these colors are referred to in the associated verbal prompts. The color of the soap bottle also is referred to in the associated verbal prompts.

The types of verbal cues, specifically the amount of detail and the timing of the cues, were determined using a combination of the system of least prompts (SLP) and time-delay procedures. The device plays a hierarchy of cues, starting with the most general description of the step to be completed. It cues a user if he or she did not complete or start a step after a set amount of time has elapsed. The device has the capability to adjust the number of levels of detail required for each user. Even

TABLE 1. Verbal cues used during preliminary testing of the COACH

Step identification	Level of detail	Recorded verbal cue
	1	Turn the cold water on.
	2	Turn the cold water on using the tap in front of you.
	3	[Name],[a] turn the water on using the blue tap in front of you.
	1	Wet your hands.
	2	Put your hands under the water and wet them.
	3	[Name], pur your hands under the water and wet them.
	1	Use the soap.
	2	Use the soap in front of you.
	3	[Name], use the pink soap from the bottle in front of you.
4	1	Rinse your hands.
	2	Put your hands under the water and rinse the soap off of them.
	3	[Name], put your hands under the water and rinse the soap off of them.
5	1	Turn the water off.
	2	Turn the water off using the tap in front of you.
	3	[Name], turn off both the blue and the red taps in front of you.
6	1	Dry your hands.
	2	Dry your hands using the white towel.
	3	[Name], dry your hands using the white towel on your left.

[a] Either the user's first or last name could be used, depending on which one the caregiver usually uses.

though the efficacy of the SLP technique has not been well documented with older adults who have dementia, this cueing technique was chosen because it allows for several different cues to be played to a user for a particular step instead of repeating the same cue until the user responds. Mihailidis (1998) showed that the invariant repetition approach was not very effective. The time-delay procedure was used to ensure that the user completed each required step in a timely manner, and it has been proven to be effective in assisting people to complete tasks similar to handwashing (Giles et al., 1997). The amount of time allowed to complete a step is adjustable for each user. The structure and wording of the cues were based on recommendations outlined by the author in previous research (Mihailidis, 1998). In general, the cues should be as simple as possible. They should only describe one step at a time and need not necessarily be in a familiar voice. For this device, a generic male voice (i.e., a man who was not known by the subjects) was used to record the cues. In addition to these general guidelines, the wording of the cues was based on prompts offered by caregivers in observed handwashing scenarios. From these observations, it was determined that many of the cues and the way they were worded were very similar from one subject to another. As a result, it was decided to record three different cue details for each step, where these detail levels were similar for each subject. The first level of detail is a general cue about the step to be completed. The second level provides more detail (such as the location of the water tap and the color of the soap bottle) about the step to be completed. The third level of detail is similar to the second level but addresses each subject by name in order to secure their attention prior to provision of instructions. The cues recorded for the subjects are outlined in Table 1. Note that, for step 1, the subjects are only cued to turn on the cold water to ensure that they do not burn themselves by turning on only the hot water.

DEVELOPMENT

Device Hardware

Some previous devices, including that used in the authors' pilot study, used transducers to monitor each step. For example, microswitches were placed at the sink faucet and soap dispenser. It was decided that the use of such transducers was unacceptable for this new device for two reasons: (1) the environment had to be changed to accommodate the transducers, cables, etc., and (2) a new set of transducers would have to be designed for each different activity.

As an alternative, the use of nonobtrusive and nonspecific systems that could track a user or a simple marker worn by the user were explored. The primary systems tested were electromagnetic, ultrasonic, and optical trackers. These tests re-

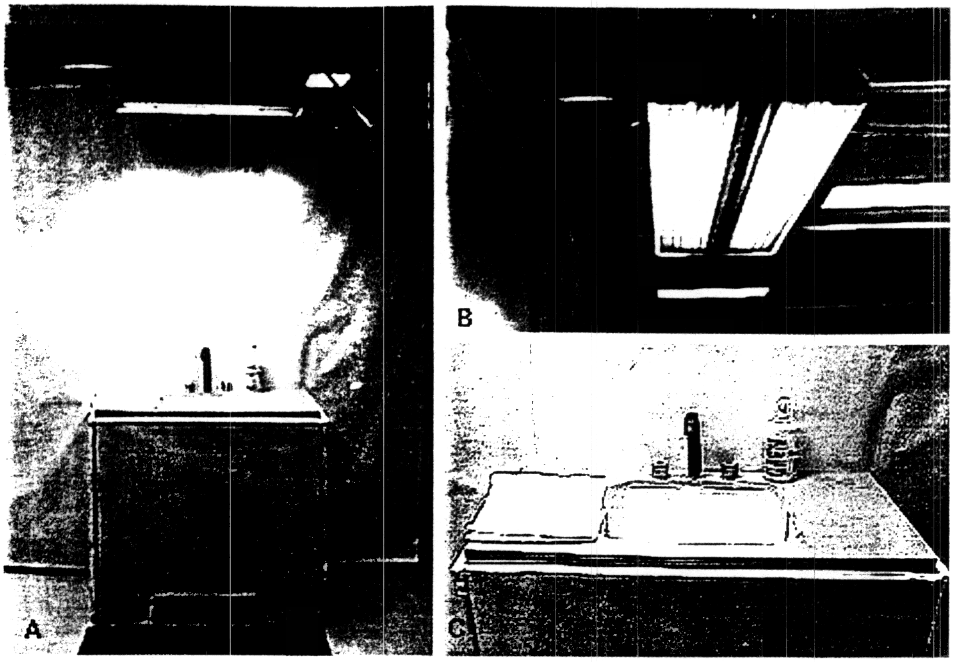

FIG. 2. A mock set-up of a sink was used to test the efficacy of the device before using it with actual subjects. **A.** The overall set-up consists of the sink counter top, the mat detector located on the floor in front of the workspace, and a lighting fixture that simulates the lighting conditions in the actual test environment. **B.** The CCD video camera is installed above the work area using a simple bracket that was attached to the ceiling. **C.** The mock workspace consists of all of the items required for handwashing (faucets, towel, and soap dispenser), positioned similar to the actual environment.

sulted in selection of an optical tracker, specifically a video camera, for the development of this cognitive device.

A monochrome CCD video camera (Panasonic WV-BP330) was mounted above a mock set-up of a sink and counter, which simulated the actual test environment (including lighting conditions). This set-up (Fig. 2) was used to test the device and its components under various conditions and to refine its operation. The camera was placed as close to the ceiling as possible to reduce parallax, and a lens was selected to provide coverage of the field of interest.

The camera was connected to a National Instruments IMAQ-1408 frame grabber card, which was installed inside the personal computer (Pentium III, 600 MHz, 128 MB RAM) that ran the agent program. This computer was located outside of the environment. The frame grabber was a high-accuracy, monochrome board, which supported various video signals. The video camera used an RS-170 signal. The board featured an eight-bit flash analogue-to-digital converter that converted video signals to digital formats (National Instruments, 1999). This output and accompanying pattern matching algorithms (which will be described later in this article) provided the two-dimensional (x and y) coordinates of the user's hand. A third dimension (depth) was not taken into account in this device but will be included in future prototypes.

A bracelet worn on the wrist of the user's dominant hand is the tracking target. The bracelet (Fig. 3) has a printed pattern of three black rings with an outer diameter of 1.75 inches (4.45 cm) and an inner diameter of 0.75 inches (1.91 cm). The rings provided high accuracy when tracked and allowed the highest sampling rate of all of the shapes tested. A sampling rate of approximately four points/second proved to be sufficient for tracking a user in real time during the preliminary trials. The pattern was repeated along the entire length of the bracelet to avoid occlusion when the user's hand was turned over.

A mat consisting of four strip switches wired in parallel was placed on the floor in front of the mock set-up. This mat was connected to a National Instruments DIO-24 data acquisition card (DAQ) installed in the personal computer. The mat allows the agent program to detect when the user is standing at the sink and if he or she walks away in the middle of the activity. The DAQ card converts the signal from the mat (the presence or absence of +5 volts) into a digital binary signal. If this binary signal is on, then the person is standing at the sink. The mat detector was required because the video camera's field of view only covered the

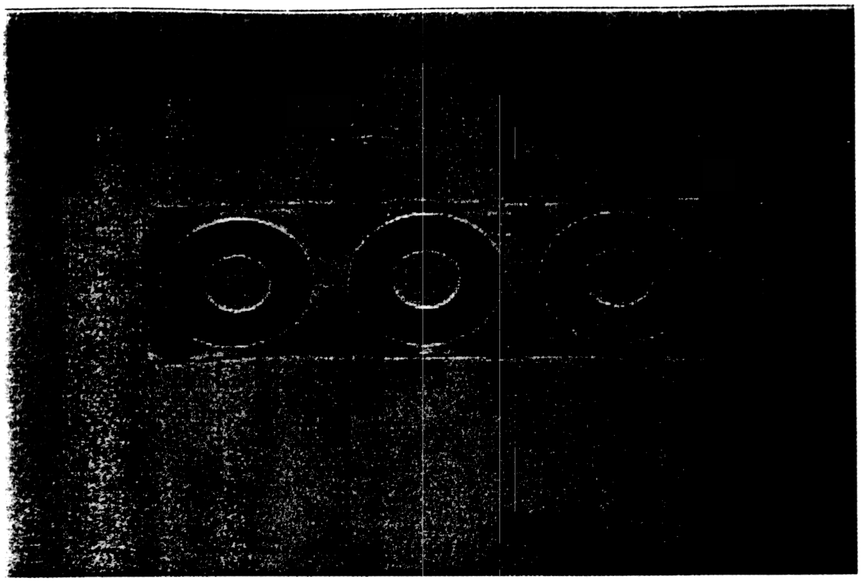

FIG. 3. The bracelet used to track the position of the user's hand has a printed pattern of three black rings with an outer diameter of 1.75 inches (4.45 cm) and an inner diameter of 0.75 inches (1.91 cm). The bracelet is made from cotton material and uses Velcro fasteners.

immediate sink area in order to avoid confusion with extraneous objects in the surrounding environment or with patterns on the user's clothing. Therefore, the device was not able to use the video image to detect whether the user was standing at the sink. This problem will be corrected in the next prototype.

A set of speakers was connected to an external amplifier, which in turn was connected to the audio output on the computer's sound card.

Even though the COACH requires various hardware to be installed, it is less obtrusive than earlier devices because its hardware is not visible to the user. A majority of users who did not have any type of cognitive disability did not notice the hardware until they were made of aware of it by the researchers. As well, this hardware does not modify the existing environment to the extent required for earlier devices, such as those by LoPresti et al. (1997) and Mihailidis et al. (2000).

Agent Program

The agent program is essentially the brain of the COACH. It receives information from the hardware, determines what the user is doing, and makes decisions about whether corrective action is required. It consists of six subfunctions: (1) a pattern-matching algorithm, (2) a low-pass filter, (3) a probabilistic neural network (PNN), (4) a logic module, (5) a plan recognition module, and (6) the action module (AM). In addition, the program provides information about the user's progress, device settings, etc., through a graphical user interface.

The subfunctions, except for the pattern-matching algorithm, were developed in Matlab using vector operations, which allows normally complex operations to be completed with more ease and speed than a nonvectored algorithm. A vector, or a particular element of one, defined all events, inputs, and records. For example, vectors were used to store lists of cues played and steps completed. All of the verbal cues were assigned cue identification numbers when they were recorded. When the device needs to determine if a particular cue has already been played to the user, a simple search of a vector whose elements are the cues already issued occurs. Vectors are necessary to store this type of information because the device needs to track sequences of events, not just samples of events in time (i.e., scalar events). This programming technique also allows a better real-time device to be developed, as time delays between input and action can be reduced. The algorithms were programmed to receive input that indicates the user's identification as well as which ADL is being completed. Using these inputs, the program only accesses relevant variables, data, and memory. This allows for individualized preferences and habits to be stored and recalled by the device as necessary.

Pattern Matching

An algorithm that uses pattern-matching techniques was developed to track the position of the

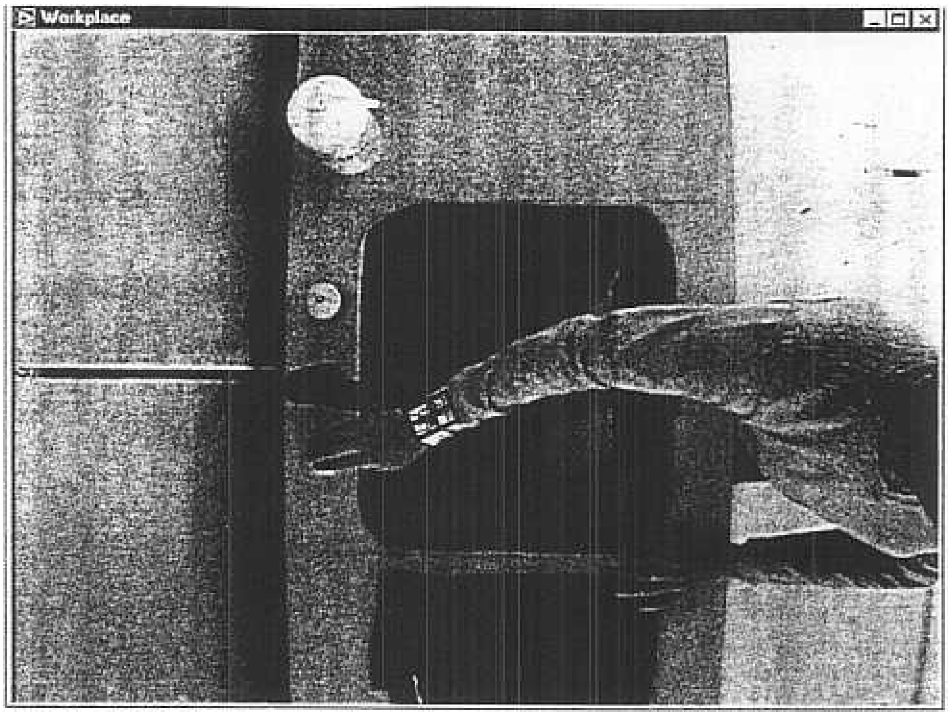

FIG. 4. The pattern-matching algorithm was programmed so that, when a match was found, it was illustrated on the video image using a red target (which is represented by a black target in this figure). The two-dimensional coordinates (x and y) of this target is calculated and provided to the remainder of the agent program.

user's bracelet. It was programmed using National Instrument's IMAQ Vision, which is a library of image-processing functions available in LabView. This algorithm was first trained by providing it with a sample image of the pattern on the bracelet. From this sample, a template of the pattern was stored in the memory of the device and was used to make matches of the same pattern in subsequent images provided by the camera. These matches were found using normalized cross-correlation techniques (National Instruments, 1999). Once a match was made within the new image, the x and y coordinates of the match were calculated and provided to the remaining parts of the agent program. The algorithm was programmed so that, when a match was found, it was illustrated on the black-and-white video image using a computer-generated red target as shown in Figure 4. This algorithm was also programmed using several image-understanding techniques. One important technique was the use of shift-invariant searches versus rotation-invariant searches. A shift-invariant algorithm looks for a change in the position of the object that it is trying to match, whereas a rotation-invariant algorithm looks for changes in both position and orientation (National Instruments, 1999). Obviously, a shift-invariant algorithm would be much quicker than a rotation-invariant one and therefore was implemented in the algorithm. This was why a symmetric shape, such as a ring, resulted in a much higher sampling rate than the other shapes tested.

Low-Pass Filter

In addition to calculating the spatial coordinates of the user's bracelet, the speed with which the bracelet moved was also calculated. This was accomplished in an algorithm that used the equation

$$V = \frac{\sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2}}{t_2 - t_1} \text{ pixels/second,} \quad (4)$$

where (x_1, y_1) are the previous coordinates of the bracelet, (x_2, y_2) are the current coordinates of the bracelet, and $(t_2 - t_1)$ is the total time it took for the user to move from the previous position to the current one. This additional input allows a rudimentary low-pass filter to be applied to data passed to the parts of the program that analyze the input and make decisions. This limit ensures that data points are not taken while the user's hand is in transition from one area to another or from random errors in the pattern-matching algorithm (both of which may reduce the performance of the overall device).

Probabilistic Neural Network

After the pattern matching algorithm has calculated the x and y coordinates for the matched pattern, the next algorithm is the probabilistic neural network (PNN). As previously described, this type of ANN uses probability theory to learn which steps correspond with the various inputs from the environment. The network classifies these inputs into step identification numbers. This part of the agent program was developed using a standard algorithm outlined by Masters (1993). It used all of the equations and methods outlined previously, with some modifications by the authors, such as the following:

- A comparison of the Bayes confidence value for a specific input with a specified confidence limit is used to determine whether the device will proceed. If the calculated value is greater than the cut-off value, the device will proceed with the classification made and will complete the remainder of the program. If the value is below the cut-off, the device stops and, via a message on the GUI, asks if this new input should be added to the training data. However, this situation rarely occurs if a comprehensive training set is collected initially.
- Two validation techniques, jackknifing and K-fold cross-validation, were also added to the standard PNN algorithm. These measures calculate the accuracy of the PNN based on the training data collected. Both of these methods divide the training data into two sets. They use one set to train the PNN and the other set to test its classification accuracy. The jackknifing technique, which is normally used for smaller training sets, extracts a sample set (i.e., the one to be used to test the network's accuracy) of size $n = 1$, and the K-fold technique extracts a sample set of any specified size. The K-fold technique also uses a standard t-test to calculate the mean accuracy and confidence intervals (Masters, 1993).
- A defining system within the PNN code was used to assign step identification numbers. During the initial training of the device, all correct steps were assigned step identification numbers between 1 and 99, all steps not related to the activity were assigned numbers between 100 and 999, and all steps not related to the activity but that do not cause an error in the overall activity were assigned numbers greater than 1,000. This allowed for steps to be easily defined and categorized (which is important in later parts of the program) and for new steps to be added easily.

Logic Module

An algorithm that applies basic logic rules to the output of the PNN was developed in order to take into account any ambiguities that may exist in the training data. These ambiguities result because the position of the user's hand may not be uniquely related to a specific step. For example, the steps of turning the water on and turning it off have the same spatial coordinates. The only thing that distinguishes these two steps from each other is their positions in the overall plan and the successful completion of the previous steps that were prerequisites to the current step. This algorithm applies these rules by using prerequisites for each ambiguous step. For the example of turning the water off, a rule has been programmed that basically says "in order for the step being completed to be interpreted as turning the water off, the step of turning the water on must have already been completed." Using this rule, the algorithm then searches a vector that keeps track of all of the completed correct steps by the user (the step vector) and sees if this prerequisite exists. If the prerequisite has not been completed, the algorithm changes the initial step identification number to the one that is causing the ambiguity, in this case, turning the water on. Currently, this algorithm has to be manually programmed for each new ambiguity and ADL step. We plan to automate this feature in the next prototype.

Plan Recognition Algorithm

The next stage is the plan recognition algorithm (PRA), which includes a situated planning algorithm. Using the output from the PNN and the logic module, the PRA determines which plan the user is completing by comparing a vector that keeps track of the steps already completed (the step vector) with a preexisting database of possible plans that could be executed (the action taxonomy). This database contains several row vectors. Each vector constitutes a plan and is a different sequence of step identification numbers. This algorithm was developed using simple vector searches instead of the more complex methods used in previous devices. It uses these simple searches to find the step vector within one of the database plan vectors. If a match cannot be found, the algorithm will attempt to predict which plan the user is attempting. This is done by removing one at a time the most recently added step to the step vector until a match can be found.

Once a plan has been found, the algorithm becomes a situated planning agent. The algorithm guides the user through the remaining steps in the selected plan. If the user deviates from this plan by either completing an incorrect step or a step out of sequence, the next algorithm is called to correct the user's actions using a verbal prompt. However, if the user deviates from the plan and this deviation may still lead to the goal being met, the PRA will find another plan from the taxonomy that explains this change or will add this new plan to the action taxonomy. If the user completes an irrelevant task, the PRA will wait a set amount of time and then start the monitoring process again.

Action Module

The final algorithm is the action module (AM), which is responsible for selecting and playing a prerecorded verbal cue when the PRA indicates that this is necessary. Several different verbal cue details can be played to the user for a particular step before assistance from a caregiver is requested. The AM has the ability to select varying levels of cue detail depending on (1) the user's past performance of the step and (2) how many errors the user has made while attempting the current step. The first option is used to select the starting level of cue detail, and the later one is used to increase progressively the cue detail until the user successfully completes the step. The selection of the required cue detail is based on an individualized performance history, which allows the device to learn about the user's abilities, tendencies, and habits. The performance history is a running average of the user's success rates for each individual step in the overall activity and is stored for each user in the memory of the device. Depending on the success rate for a particular step, one of the prerecorded verbal cue details will be selected and played over the speakers hidden within the environment. If the user's success rate has increased or decreased, the selected cue detail is decreased or increased, respectively. The performance history is constantly updated to reflect the most recent performance of the user and hence the most recent successful cueing strategy for the user. This algorithm can also skip to the most detailed cue if appropriate; this is decided by calculating the slope of the user's performance over the past five attempts. A rapid decline in the user's performance over the last few trials can be identified as a negative slope, and if this exceeds a preset limit, the device will play the most descriptive cue right away. When the user's performance begins to improve, the slope will increase above the limit, and the device will return to selecting cues based on the user's performance history.

Other Algorithms and the Graphical User Interface

Algorithms that maintain statistics and reports on each user's performance have also been developed. These statistics include the running averages used in the individualized performance histories, a histogram that shows the frequency of cues issued for each required step, and the total time it takes the user to complete the activity. Each event that occurs, such as cue played, steps completed, and call to the caregiver, are time stamped and stored in a data file for each user.

The device has a remote display with a graphical user interface (GUI), which displays information about the user's performance, actions taken by the device, and the device's settings, as illustrated in Figure 5. The GUI also uses visual and audible alarms to call the caregiver into the environment when the device has failed to assist the user.

EVALUATION

Preliminary Results

Subjects who simulate confused behavior (surrogate users) are being used to purposely make the device fail in an attempt to discover device features that need to be changed before the clinical trials. This preliminary phase is also being used to estimate the performance of the device, including the PNN.

The accuracy of the training set (n = 182) was tested using the jackknifing technique. It had a classification accuracy of 84%. The reason for this relatively low value was the ambiguities that existed between inputs (as previously described). These ambiguities caused confusion within the network and resulted in some wrong classifications. However, since the logic module has been developed to correct any errors resulting from ambiguities, this poor accuracy does not affect the overall performance of the device. The jackknifing technique was run using various values for σ in the PNN to optimize this parameter (values ranging from $\sigma = 0.01$ to $\sigma = 1.0$). From these tests, a σ value of 0.3 was found to produce the best results from the PNN and this specific training set. It is unlikely that this value will have to be changed as new data are added to the training set because of the robustness of the PNN and the relative insensitivity of a PNN with respect to changes in σ (Masters, 1993).

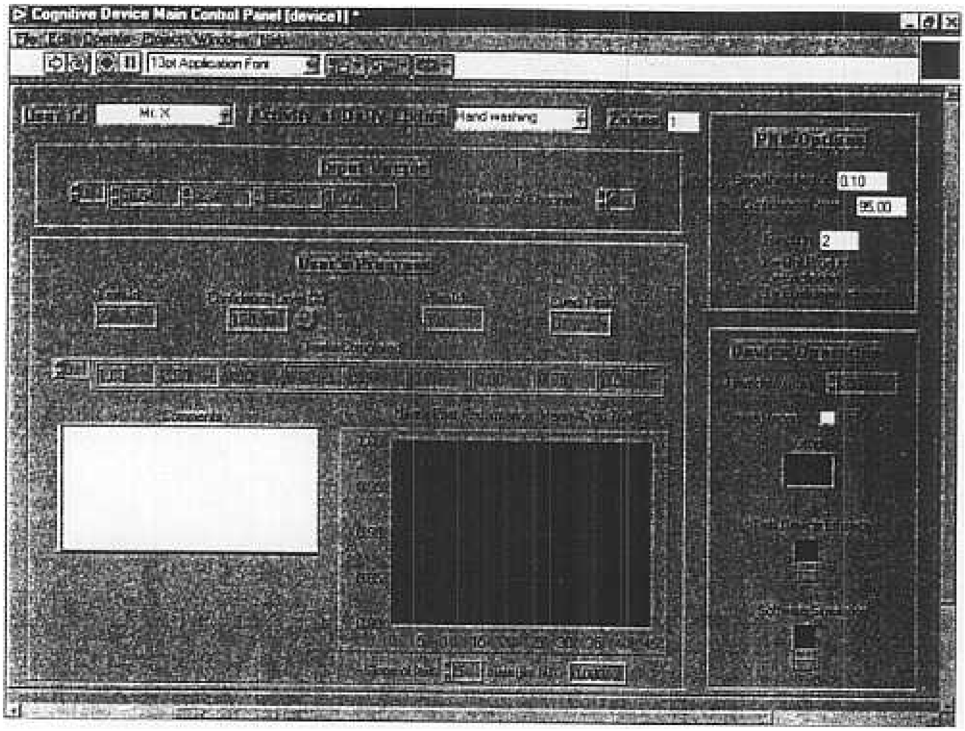

FIG. 5. The device has a graphical user interface (GUI), which displays information about the user's performance, actions taken by the device, and the device's settings. The GUI is also used to call the caregiver into the environment when the device has failed to assist the user. This GUI was developed using LabView v.5.1. and is displayed on a standard VGA monitor located outside of the washroom.

The efficacy of the device—a measurement of how well the device does what it is supposed to do—needs to be studied (Salminen & Petrie, 1998). The efficacy of the device is measured using signal detection theory techniques, which look at the number of correct and incorrect actions that the COACH performs. With respect to these types of measurements, preliminary data showed that the device has an efficacy of approximately 95%. It is hoped that the efficacy of the device will be further improved by removing the remaining bugs.

Finally, the performance of the low-pass filter was tested by running trials both with and without the velocity calculation algorithm in the agent program. Without the filter, the device was much poorer in its operation. The presence of extraneous data points, which would normally be filtered out, caused the device to take unnecessary actions, such as playing nonrequired cues.

Future Clinical Evaluation

Extensive clinical testing of the device in a handwashing scenario has begun. A single-subject research design (SSRD) will be used with 10 subjects who have moderate to severe dementia (Barlow & Hersen, 1984; Kratochwill, 1978). Data will be collected with respect to the user's performance of handwashing with and without the device and with respect to the operation of the device itself.

DISCUSSION

Preliminary data suggest that the COACH has the ability to monitor a user during handwashing. It can adapt its settings and cueing strategies for an individual user without the need for reprogramming or long training periods. It can learn about the behavior of a new user and adjust the required parameters automatically and in real time.

The preliminary trials have demonstrated that the COACH acts in an appropriate manner for the situations encountered with the surrogate users. It was easily trained for handwashing. It successfully determined which task the user was attempting and, when needed, played timely verbal cues. The algorithm that played varying levels of cue detail also seemed to select cues appropriately based on the individual performance of the surrogate users. Compared with previous cognitive devices, the COACH seems to be able to elicit information from the environment and user more quickly. It does not require input from the user or a caregiver to be able to adapt itself to the individual's preferences.

This has allowed the device to handle variations in the way surrogate users have completed the ADL and in the mistakes made. The device has adapted plans according to the preferences of each user and changed the cue details for those users who had more difficulties completing the activity than others. The use of AI and advanced hardware also has allowed us to move away from the notion of manually constructing one-of-a-kind systems for each user. The only manual customization that is required for the users of the COACH is to record cues for each person, which takes approximately 20 minutes.

These trials have also shown that the COACH requires some improvements. Changes to existing algorithms and strategies that are more intelligent are needed to handle unforeseen situations that might arise. For example, the use of the soap may be repeated during the later parts of handwashing. The device must be able to recognize this error, determine how far into the sequence this error has occurred, and then reguide the user through the steps associated with rinsing off the soap. This may involve repeating certain prerequisite steps such as turning the water back on. Currently, the COACH can recognize this error but calls a caregiver to enter the environment and assist the user through the required prerequisite steps. The next prototype will use a more intelligent and automated algorithm to recognize and correct an error of this sort. There are also some inefficiencies in the agent program that need to be corrected in future versions of the device; currently, these inefficiencies cause a slight time delay between certain actions taken by the device, such as playing a cue after detection that the user left the sink without completing the entire activity. Finally, three questions need to be answered about the device's tracking hardware: (1) Will an individual with a cognitive disability accept wearing a bracelet, especially one of the size described earlier? (2) Will a bracelet for each hand be required or will one on the user's dominant hand be sufficient? (3) Will one camera be sufficient for more complex ADL tasks? These questions cannot be answered until the clinical evaluation has been completed. It is suspected, however, that one bracelet will be sufficient but that more than a single camera will be required for more complex ADL tasks such as using the toilet in order to avoid problems created by obscured views.

At the time of writing, it is not possible to prove the true efficacy of the device and its cueing strategies since the tests with subjects who have dementia have not been completed. These preliminary trials demonstrated that the device is working properly and that, with some minor modifications, we are ready to proceed to the final stage of this research—testing with 10 subjects who have moderate to severe dementia. During these clinical trials, the efficacy of the device will be studied. We will observe how the subjects respond to the issued cues, whether the subjects exhibit any type of emotional response, such as becoming annoyed by the COACH's cues, and whether the subjects' performances with the device improve, decline, or remain the same over time.

IMPLICATIONS

A successful cognitive orthosis could have a positive effect on the lives of some people with dementia as well as on their caregiver or family members. Obviously, such a device will not replace the caregiver, but could provide an alternative to the need for continual monitoring and assistance during common tasks. Cognitive orthotics might also be applied to increasing the activity of older adults in other ways, such as assisting a cognitively disabled person to use a powered wheelchair more effectively or to use common household appliances more safely. If these goals can be achieved, cognitive orthoses have significant roles to play in improving the quality of life of people with cognitive disabilities.

Acknowledgments: This research is funded by the Alzheimer Society of Canada, Lifeline Systems, Inc., and the Ontario Rehabilitation Technology Consortium (ORTC).

REFERENCES

Agostinelli, B., Demers, K., Garrigan, D., & Waszynski, C. (1994). Targeted interventions: Use of the Mini-Mental State Exam. *Journal of Gerontological Nursing, August,* 15–23.

American Alzheimer's Association. (1998). *Alzheimer's disease fact sheet* [On-line]. Available: http://www.alz.org.

Bai, J., Zhang, Y., Cui, Z., & Zhang, J. (2000). *Home telemonitoring framework based on integrated functional modules.* Paper presented at the World Congress on Medical Physics and Biomedical Engineering, Chicago.

Barlow, D. H., & Hersen, M. (1984). *Single case experimental designs: Strategies for studying behaviour change* (2nd ed.). New York: Pergamon.

Beck, C., Heacock, P., Rapp, C. G., & Mercer, S. (1993). Assisting cognitively impaired elders with activities of daily living. *American Journal of Alzheimer's Care and Related Disorders and Research, 8*(6), 11–20.

Bergman, M. (1996). Computer orthotics: Fostering self-sufficiency in people with cognitive challenges. *Disability Today, Fall,* 54–55.

Bergman, M. M. (1997). *People with cognitive challenges can enjoy rapid success in acquiring skills and managing their lives: The exciting breakthrough of cognitive orthotics.* Paper presented at the Technology and Persons with Disabilities, California State University, Northridge (CSUN).

Bergman, M. M. (1998). A proposed resolution of the remediation-compensation controversy in brain injury rehabilitation. *Cognitive Technology, 3,* 45–51.

Bourgeois, M. S. (1990). Enhancing conversation skills in patients with Alzheimer's disease using a prosthetic memory aid. *Journal of Applied Behaviour Analysis, 23,* 29–34.

Cavalier, A. R., & Ferretti, R. P. (1993). *The use of an intelligent cognitive aid to facilitate the self-management of vocational skills by high school students with severe learning disabilities.* Paper presented at the RESNA 1993 Annual Conference, Arlington, VA.

Chute, D. L., & Bliss, M. E. (1988). *Prosthesis ware: Personal computer support for independent living* [On-line]. Available: http://www.homemods.org/library/life-span/prosthesis.html [June 13, 2000].

Chute, D. L., & Bliss, M. E. (1994). ProthesisWare. *Experimental Aging Research, 20,* 229–238.

Cockburn, J., & Collin, C. (1988). Measuring everyday memory in elderly people: A preliminary study. *Age and Ageing, 17,* 265–269.

COGENT. (1998). *ISSAC—Equipping people for independence* [On-line]. COGENT Systems, Inc. Available: www.cosys.com/index.html [October 26, 2000].

Cole, E. (1999). Cognitive prosthetics: An overview to a method of treatment. *NeuroRehabilitation, 12,* 39–51.

Engelman, K. K., Altus, D. E., & Mathews, R. M. (1999). Increasing engagement in daily activities by older adults with dementia. *Journal of Applied Behaviour Analysis, 32,* 107–110.

Francik, E. (1999). *Telecommunications problems and design strategies for people with cognitive disabilities* (annotated bibliography and research recommendations). Oakland, CA: World Institute on Disability.

Franklin, D. (1998). *Cooperating with people: The intelligent classroom.* Chicago: University of Chicago Press.

Giles, M. G., Ridle., J. E., Dill, A., & Frye., S. (1997). A consecutive series of adults with brain injury treated with a washing and dressing retraining program. *The American Journal of Occupationl Therapy, 51,* 256–266.

Harrell, M., Parente, F., Bellingrath, E. G., & Lisicia, K. A. (Eds.). (1992). *Cognitive rehabilitation of memory: A practical guide.* Gaithersburg, MD: Aspen.

Health and Welfare Canada. (1991). *Designing facilities for people with dementia.* Ottawa: Department of National Health and Welfare.

Kautz, H. A., & Allen, J. F. (1986). *Generalized plan recognition.* Paper presented at the Fourth National Conference on Artificial Intelligence, Philadelphia, PA.

Kiani, K., Snijders, C. J., & Gelsema, E. S. (1998). Recognition of daily life motor activity classes using an artificial neural network. *Archives of Physical and Medical Rehabilitation, 79,* 147–154.

Kirsch, N. L., Levine, S. P., Lajiness, R., Mossaro, M., Schneider, M., & Donders, J. (1988). *Improving functional performance with computerized task guidance systems.* Paper presented at the ICAART 1988 Annual Conference, Arlington, VA.

Kirsch, N. L., Levine, S. P., Lajiness-O'Neill, R., & Schneider, M. (1992). Computer-assisted interactive task guidance: Facilitating the performance of a simulated vocational task. *Journal of Head Trauma Rehabilitation, 7*(3), 13–25.

Kratochwill, T. R. (1978). *Single subject research: Strategies for evaluating change.* New York: Academic.

Lange-Alberts, M. E., & Shott, S. (1994). Nutritional intake: Use of touch and verbal cueing. *Journal of Gerontological Nursing, 20*(2), 36–40.

Levine, S. P., & Kirsch, N. L. (1985). *COGORTH: A programming language for customized cognitive orthotics.* Paper presented at the RESNA Annual Conference, Arlington, VA.

Levinson, R. (1997). PEAT: The Planning and Execution Assistant and Training System. *Journal of Head Trauma Rehabilitation, April.* Available: www.brainaid.com/publicat.htm.

LoPresti, E. F., Friedman, M. B., & Hages, D. (1997). *Electronic vocational aid for people with cognitive disabilities.* Paper presented at the RESNA 1997 Annual Conference, Arlington, VA.

Masters, T. (1993). Probabilistic neural networks. In *Practical neural network recipes in C++* (vol. 1, pp. 202–222). New York: Academic.

Mihailidis, A. (1998). *The use of an automated cueing device to assist people with dementia during handwashing.* Unpublished master's thesis, University of Toronto, Toronto.

Mihailidis, A., Fernie, G. R., & Cleghorn, W. L. (2000). The development of a computerized cueing device to help people with dementia to be more independent. *Technology and Disability, 13,* 23–40.

Mozer, M. C. (1998). The neural network house: An environment that adapts to its inhabitants. In *Proceedings of the AAAI Spring Symposium* (pp. 110–114). Menlo Park, CA: AAAI.

Nambu, M., Nakajima, K., Kawarada, A., & Tamura, T. (2000). *A system to monitor elderly people remotely, using the power line network.* Paper presented at the World Congress on Medical Physics and Biomedical Enginering, Chicago.

Napper, S. A., & Narayan, S. (1994). *Cognitive orthotic shell.* Paper presented at the RESNA 1994 Annual Conference, Arlington, VA.

National Instruments. (1999). *IMAQ PCI/PXI – 1408 user manual.* Austin, TX: National Instruments Corporation.

Ogawa, M., Ochiai, S., Shoji, K., Nishihara, M., & Togawa, T. (2000). *An attempt of monitoring daily activities at home.* Paper presented at the World Congress on Medical Physics and Biomedical Engineering, Chicago.

Parzen, E. (1962). On estimation of a probability density function and mode. *Annals of Mathematical Statistics, 33,* 1065–1076.

Patterson, C. (1999). Focusing on Alzheimer's disease. *The Canadian Journal of Diagnosis, December,* 62–74.

Russell, S., & Norvig, P. (1995). *Artificial intelligence: A modern approach.* Upper Saddle River, NJ: Prentice Hall.

Salminen, A.-L., & Petrie, H. (1998). *Evaluating assistive technology prototypes: Laboratory or real life contexts?* [On-line]. Available: www.dinf.org/tide98/70/salminen_petri.html.

Specht, D. (1992). Enhancements to probabilistic neural networks. *IEEE,* 761–768.

Steele, R. D., Weinrich, M., & Carlson, G. S. (1989). *Recipe preparation by a severely impaired asphasic using the VIC 2.0 interface.* Paper presented at the RESNA 1989 Annual Conference, Arlington, VA.

Vanderheiden, G. (1998). *Assistive techniques and devices for persons with cognitive and language impairments.* Madison, WI: Trace Research and Development Centre, University of Wisconsin.