Integrating Sensing and Cueing for More Effective Activity Reminders

Joseph Modayil, Rich Levinson, Craig Harman, David Halper, Henry Kautz

Abstract

We are investigating how sensors can improve a portable reminder system (PEAT) that helps individuals accomplish their daily routines. PEAT is designed for individuals who have difficulty remembering when to perform activities because of cognitive impairments from strokes or other brain injuries. These impairments can cause a significantly reduced quality of life for afflicted individuals and their caregivers. PEAT provides assistance by planning a schedule of activities for a user, and by cueing the user when activities should begin or end. One limitation of PEAT is that it requires the user to manually indicate when an activity starts or stops, which causes unnecessary cues for a user who needs only occasional reminders. By incorporating feedback from reliable sensors, the software can automatically infer which activity the user is performing. With this information, we expect that PEAT will be able to cue the user more effectively, by not cueing the user when sensors indicate that activities have already started or stopped, and by providing compliance cues that remind the user when steps of an activity have been forgotten. We present a description of the system, implemented scenarios, and a discussion of potential benefits and pitfalls with this approach.

Introduction

Impaired cognitive function presents a significant challenge for many elderly persons. Common failures include failing to start an activity, stalling after an activity, forgetting a required component, or performing the activity incorrectly. Although everyone experiences occasional lapses, when these lapses become chronic then activities of daily living (ADLs) become infeasible without regular assistance from a caregiver. This presents a significant burden to the patient, the caregiver, and the health-care system.

PEAT (the Planning and Execution Assistant and Trainer) is a cognitive orthosis which runs on a cell phone and helps compensate for executive function impairment. Executive functions refer to cognitive abilities that are required for goal-directed behavior. People rely on executive functions for planning and carrying out daily activities, staying on track by inhibiting distractions, adjusting to novel situations and recovering from performance errors (Levinson 1995a). Many individuals with mild cognitive impairment can still function independently, given appropriate external cues. The role played by a caregiver or PEAT is to compensate for the user's impaired executive function by monitoring the user's behavior and providing reminders when required to ensure that activities are performed correctly. We can think of the performance of activities as a two level system, with a set of low-level reactive behaviors capable of complex environmental interaction through sensory motor coordination, and with a high-level deliberative supervisor that recognizes behaviors, enforces constraints, and intervenes when appropriate.

In previous work, the primary focus of PEAT has been on scheduling and enforcing temporal constraints between activities (Levinson 1997). PEAT's perception of what activity the user is performing is limited to self-reports where the user indicates the current activity on the GUI. The purpose of this work is to explore how PEAT can incorporate sensors to automatically infer the current activity and thus provide more appropriate cues.

Augmenting PEAT with sensors provides several potential benefits to the user. First is a reduction in irrelevant cues for starting and stopping activities when an activity classifier can infer the user's current activity. Second, the sensors permit therapy compliance cueing where PEAT can remind a user to use an object as part a task (for example, to use a cane while getting the newspaper). Third, PEAT can cue the user when they are stalled or perseverating in a task. Fourth, PEAT can assist with performance errors such as going to the wrong location or picking up the wrong object. Finally, there is the possibility for therapeutic monitoring, where logs from PEAT can be used to infer how often the user used the cane while walking, or how much time the user spent outside.

To the best of our knowledge, this is the first portable

(a) (b) (c)

Figure 1: Several technologies are integrated in this project. (a) PEAT runs on a cell phone. PEAT cues the user using dialog boxes. (b) A wearable RFID reader is used to detect handled objects. (c) A pressure mat detects that someone is at a known location. The sensors transmit data through wireless interfaces.

reminder system to "close the loop" by using sensors to detect what the user is doing instead of forcing the user to directly respond to the device. Moreover, we demonstrate that these improvements are feasible in real-time with the limited computation and memory available on a cell phone. The following sections provide a more detailed description of PEAT and our progress on this project, along with some implemented scenarios. We conclude with a discussion of both related work and open issues.

Technologies

The PEAT software is sold by Attention Control Systems and runs on Windows Mobile platforms, including cell phones (Figure 1). The intended users are people with impaired executive functions who have difficulty remembering when and how to perform daily activities. Primarily, PEAT provides assistance by maintaining a schedule of a user's activities and automatically cueing the user when activities need to be started, resumed, or completed. A distinctive aspect of PEAT is the use of reactive planning to adjust a user's schedule when an activity takes an unexpected amount of time, or the user updates the calendar. PEAT monitors activity durations and warns the user about approaching deadlines and scheduling conflicts.

PEAT represents an instance of an activity as a task. Each task has temporal properties such as a start time and expected duration. As the user progresses through their day and updates the status of the current task, PEAT reactively updates the day's schedule, and advises the user when tasks should start or stop, when conflicts arise, or when decisions must be made.

A federally-funded, three-year randomized controlled trial to evaluate PEAT's efficacy with 100 subjects has recently been completed (Fisch *et al.* 2007), and a similar study is beginning at the University of Maastricht in the Netherlands. Preliminary results show that PEAT provides benefits over a traditional therapy of pen and paper schedules.

In order to provide effective cues to users with cognitive limitations, the original version of PEAT provides all cues in modal dialog boxes that simplify the user interface by requiring the user to respond before permitting any further interaction with the device. However, this is the only manner in which PEAT is able to track the user's activities. Thus, to benefit from PEAT, the user is forced to respond to two cues for every scheduled activity to inform PEAT of progress. This cueing may be reassuring to some users (PEAT is managing the schedule), but may be annoying to users who make only occasional mistakes (the cost of responding to unnecessary cues outweighs the perceived benefit from when PEAT catches an error). Moreover, the user's executive function could deteriorate through excessive reliance on reminders from PEAT. Reducing the number of unnecessary cues that PEAT generates could reduce the risk of learned dependency.

We are extending PEAT with sensors in order to automatically detect when a user is performing an activity. The sensors provide input to activity classifiers that infer which activity the user is currently attempting to perform. A variety of sensors are currently available, including RFID, pressure mats, and GPS. Each sensor has distinct characteristics with respect to availability, coverage and error rates. We are focusing on sensors with a low false positive error rate. The sensor coverage is effectively disjoint for GPS signals (which are only available outdoors) and pressure mats (which are intended for indoor use). Hence, although multiple sensors are available, they may not all be available simultaneously or for a single activity. The absence of a sensor observation can be managed by the introduction of a missing observation symbol into the sensor stream. The

GPS has two symbols for missing observations, one to handle the case when GPS is turned off, and one for when GPS is turned on but no signal is being received.

We have implemented interfaces to PEAT for several sensors. The RFID reader is a prototype developed by Intel Research Seattle and has a bracelet form-factor. The RFID reader scans RFID tags and then transmits the identifiers in real time through a wireless ZigBee network. RFID tags are placed on several objects in the environment, and whenever the bracelet is within a few inches of a tag, the reader is able to scan the tag's unique identifier. The pressure mats come from Trossen Robotics and are connected to PEAT through a Bluetooth network. The pressure mats provide strong evidence that someone is working in a certain area. The iPaq cell phone running PEAT has on-board GPS capabilities. The GPS provides a secondary source of location information.

Theory

PEAT compensates for a user's impaired executive function by monitoring the execution of user activities and intervening when required to enforce constraints on sequences and deadlines. Previously, PEAT emphasized managing constraints and generating effective cues. PEAT's perception of the user's behavior was limited to self-reports.

The introduction of sensors necessitates some changes to PEAT. The structure of the modified PEAT system can be represented by the tuple

$$<X, AC, CM, CG>$$

consisting of a *state vector* X and separate modules for *activity classification*, *constraint management*, and a *cue generation*. The state contains the latest sensor readings, the current attempted activity, and the activity schedule. By incorporating sensors into the platform, the new activity classification module provides the perception, while the constraint manager compares the activity that the user is attempting to perform with the activity's constraints. When errors are detected, a cue is generated based on the user's cueing preferences. Note that the activity classifier does not need to detect errors in the performance of a task, nor does it need to generate appropriate cues, as these functions are provided by the constraint manager and the cue generator respectively. This provides a clear separation of concerns between the modules.

State Vector

The state vector contains multiple types of information. It contains the most recent observations from each sensor. For example, the last object touched was the coffee mug and the last location was the kitchen. It contains the $CurrentActivity$ as estimated by the activity classification module. It also contains the current schedule of activities.

Activity Classification

Activity classification (AC) using RFID sensors can be quite accurate (Patterson *et al.* 2005). Moreover, positioning sensors such as GPS or pressure mats provide a valuable independent source of information, when they are available.

Given observation streams from sensors, activities can be classified with a hidden Markov model (HMM) (Rabiner 1989). In a simple HMM for activity recognition, there is one state per activity, and each activity has some probability of generating the current sensor observations. Formally, given a set of activities A and a set of observations O, the probability of having performed activity a_t at time t after seeing the observation o_t only requires knowing the probability distribution over the activities at the previous time step.

$$P(a_t|o_{1,\ldots,t}) \propto P(o_t|a_t) \sum_{a_{t-1} \in A} P(a_t|a_{t-1}) P(a_{t-1}|o_{1,\ldots,t-1})$$

At each time-step, the best estimate of the current activity is updated in the state vector. The estimate is qualitative, either indicating that the activity classifier is confident that the activity is currently being attempted or indicating that the current activity is ambiguous when the classifier is not confident.

$$CurrentActivity \in A \cup \{Ambiguous\}$$

The classification is ambiguous when the probability of the most likely activity is falls below a threshold.

Although an HMM is commonly used with the Viterbi algorithm to find the most likely sequence of states to explain a sequence of observations, PEAT needs to know what the user is *currently* attempting. This classification must be provided online and in real time for timely and effective interventions. Hence, HMM filtering is used instead of the Viterbi algorithm. Since the observation and activity sequences do not have to be stored, the filtering algorithm requires only a constant amount of memory.

The constraint manager also primes the activity classifier to expect certain activities based on the user's schedule. Priming resets the internal HMM model with a new probability distribution over the plausible activities at the next time step, but it does not change the $CurrentActivity$ in the state vector until new observations are received. Priming is useful for combining the knowledge from the user's schedule that only a few activities are likely to occur with the capability for recognizing unlikely activities given sufficient evidence. Priming also allows the activity classification to be robust to arbitrary reorderings of expected activities.

We further have developed the *Interleaved HMM* (IHMM) (Bai, Modayil, and Kautz 2008) as a better variant of an HMM for the classification of interleaved activities. The IHMM augments the state representation in a simple HMM (one state per activity) with a richer HMM state representation that stores the last observation seen in each activity (one state for each observation symbol for each activity). When activities are interleaved, the IHMM can better predict the next observation based on the last observation for the new activity. The IHMM has a very large state space, but an effective approximation reduces the portion of the state space considered at each time step to be comparable to that used for the simpler HMM. The introduction of explicit representations for the interleaving of activities improves the accuracy for both the Viterbi algorithm and the filtering algorithms.

Constraint Management

The constraint manager (CM) relies on the $CurrentActivity$ in the state vector, along with the current schedule of activities. Even though a user may be attempting to perform an activity, the user may not be performing the activity properly. The CM examines the state vector to determine when an activity's constraints are violated and how to respond.

PEAT represents flexible time constraints between activities. Each activity has an earliest start time, latest stop time, minimum duration and an expected duration. The CM manages the schedule of activities and informs the user when significant changes must be made.

Groups of activities may be collected to form routines, and routines can be hierarchical. The use of hierarchical routines simplifies the user interface, but all planning and constraint satisfaction occurs with a flat representation. The components of a routine may be unordered, ordered sequences, or disjoint choices.

In addition to the above temporal constraints, the activity descriptions have been extended to include sensor-based constraints for preconditions, invariants, and termination conditions. At each time step, the constraints of activities are compared with the data in the state vector. We list examples of these constraints for some activities below.

- Activity:GetNewspaper, Preconditions: touch cane
- Activity:MakeCereal, Invariants: location kitchen
- Activity:GetNewspaper, Termination: touch mailbox

These constraints are currently used for updating the state of the system and for cue generation.

At any point in time, the CM only performs inference over a limited set of tasks (instances of activities). By comparing the sensor constraints of the tasks to the state vector, the CM can update the status of each task to one of pending, active, paused, or complete. If the AC asserts that the user is trying to perform an activity whose preconditions are false, the CM will automatically cue the user to achieve the preconditions. In this manner, PEAT helps the user recover from performance errors during plan execution.

Cue Generator

While it is useful to cue a user, the user may not want reminders if they start or stop activities manually. The user may not want repeated reminders about the the same issue. Perhaps the user does not want to use their cane that day. Perhaps the user is currently talking on the phone and should not be interrupted at the current time.

The purpose of the cue generator is to provide appropriate cues based on the user's current state as measured by previous interactions and other sensors. In the current implementation, cue generation is based purely on the user's preferences, but we expect to extend this capability to account for the user's cognitive state.

The addition of sensors offers new possibilities for cueing. One is to provide confirmation when the activity classifier detects that the user has started a new activity, without requiring a user response. The confirmation can be as subtle as a vibration, or more blatant such as an audible tone, a pre-recorded sound segment or context-specific synthesized speech. This provides feedback to the user that the system is operating correctly without requiring the user to respond to PEAT. In future work, the user may provide feedback to PEAT to improve the probabilistic activity models.

Module interfaces

The constraint manager provides top-down priming to the activity classifier with a list of activities that are likely causes of the next observation. This constrains the difficulty of activity classification. Even though an individual may have hundreds of different activities in a day, PEAT knows that at any point in time only a few of these activities are likely. However, given sufficient evidence the AC can overcome the strong prior belief to correctly infer that a user is performing some unusual activity.

Conversely, the AC module infers which activity the user is attempting to perform, and PEAT can use this to reduce cues to the user. In particular, cues to start an activity are not required if the state vector indicates that the user has already started the activity and all preconditions are satisfied. Also, if the AC infers that a user has moved on to the next activity, then PEAT can change the status of the previous activity to paused and PEAT does not need to cue the user to stop the previous activity if termination conditions are satisfied.

Example Scenarios

Currently we have a preliminary implementation of the integrated system. The sensors and cell phone commu-

Figure 2: This is an example of therapy compliance with a simulated user. On receiving a sensor reading from the pressure mat at the doorway, PEAT recognized that the user was getting the newspaper. Although the activity of getting the newspaper has a precondition of using the cane, the user had not recently touched the cane. PEAT generated a cue to use the cane, and then the user retrieved his cane.

nicate via wireless networks using a laptop base station. The probabilities for the activity classifier are generated manually, although they can be learned. For our initial implementation, we have combined both location observations and object observations into a single observation stream for the activity classifier. We have removed the missing-observation sensor readings, so the activity classifier only performs updates when new sensor data is received. We have implemented and tested three example scenarios.

The scenarios are based on a sample morning routine (Figure 2). In the morning routine, there are three activities (making tea, making cereal, and getting the newspaper), two locations (kitchen and doorway), and several objects (cane, mailbox, tea, mug, spoon, teapot, bowl, cereal, fridge, milk). Some objects are shared between activities and two activities shared the same location.

Baseline. In the baseline scenario, the user does not wear the bracelet, and goes through a morning routine with PEAT. Because the sensors are not available, PEAT cues the user at scheduled times when activities are expected to start or end. This leads to six cues to the user (one cue each at the beginning and the end for each of three activities).

Reduced Cueing. The activity classification module facilitates reduced cueing. In this scenario, the sensors are enabled and PEAT infers which activity the user is attempting to perform. This step is non-trivial, since different activities may share objects and locations. However activity classification is reliable with adequate sensor data from a restricted set of plausible activities. The $Ambiguous$ activity class is used to avoid premature commitment to an activity when insufficient evidence has been gathered. With a competent user who does not need reminders to start or stop activities, this approach generates no cues at all. For situations where the user stalls and needs a reminder, PEAT still provides cues appropriately.

Therapy compliance. Using the activity preconditions, PEAT can also remind a user to use a mobility aid (such as a cane) when appropriate. We tested this with an activity (GetNewspaper) which has touching a cane as a precondition. When PEAT observes that the user is attempting the activity and has not recently touched the cane, PEAT generates a cue to the user to get the cane. The user at this stage can choose to either return and grab the cane, or continue the activity without the cane. In either case, PEAT logs the attempted intervention and the user's response. This style of intervention can help the user remember instructions from their therapist. Moreover the user could review the logs with a therapist to monitor progress.

Related Work

This project builds upon several pieces of related work. The PEAT system developed from research in unified planning and execution systems (Levinson 1995b). Previous research has shown that RFID technology can be used to reliably recognize ADLs (Patterson *et al.* 2005). Previous work also showed that significant locations can be inferred by clustering GPS signals (Liao, Fox, and Kautz 2004).

This project is most similar to previous work in the Autominder project (Pollack *et al.* 2003). Autominder also reasons over temporal constraints and performs intelligent cueing, either on a PDA or on a mobile robot. However, this work has not demonstrated how sensors can be integrated into the scheduling process to provide more effective cue generation.

Other research has also studied integrated sensing and cueing. Mihailidis and colleagues (Mihailidis, Carmichael, and Boger 2004; Boger *et al.* 2005) are studying how to provide effective automatic cue generation for bathroom ADLs, in particular for hand-washing. Their system senses the user with a vision system built into the bathroom. The system provides audible cues and monitors the user's behavior to infer both the user's progress in the task and the user's current cognitive state. This work focuses on building a very detailed model for one specific ADL, in comparison to the approach in this paper of providing more limited assistance over many daily activities in many locations.

Work on the House_n project (Tapia, Intille, and Larson 2004) studies how sensing technology can accurately detect which activity the user is performing. This work has evaluated the effectiveness of several sensors in the home. They have discovered issues including limitations of RFID and the effectiveness of location sensors for activity classification. PEAT is capable of using multiple type of sensors and should work with whichever sensors are available and most effective.

Discussion

Our exploration of the integrated sensing and cueing is ongoing, and there are several possible extensions for this work. One is to use automatic activity classification to facilitate a fine-grained activity specification without requiring additional interaction on the part of the user. Individual steps in an activity (such as adding milk to the tea) could be automatically learned and activities can be interleaved without requiring additional cues. Another extension is to modify cue generation based on a user's location and current activity (in a movie theater or driving). Another valuable function is to remember the last locations where objects were sensed, which could be used to help a user find their cane or their car. Finally, by extending the activity constraints with postconditions, PEAT can form plans by chaining together activities with matching preconditions and postconditions.

Although adding sensors to PEAT provides several benefits, there are potential pitfalls. One is that the increased nondeterminacy of sensor-contingent cueing could lower the user's trust in PEAT's reliability. Lowered trust could also occur if sensor errors lead to incorrect cues to the user. We believe these issues can be managed by conservative cueing and reliable sensors. Although we are generating activity models for the activity classifier manually, learning the models from training data has been effective in the past (Patterson *et al.* 2005). There could be challenges in personalizing the activity models for individual users which may require gathering some training data for each user. There are still many open issues in generating effective cues over a range of activities. The most appropriate cue could vary depending on the user's current state, which could be subject to both intra- and inter-day variations. There is also the potential for the deterioration of an individual's capabilities from illness progression or increased prosthetic reliance. Finally, there are privacy issues that can arise if the information about a user's activities is not managed effectively. Currently all this information is stored locally on the cell phone and is under the user's control.

Conclusions

We have described how a reminder system can incorporate sensors to improve performance. The system infrastructure is flexible, adjusting to incorporate the available sensors but functioning even when sensors are not available. The addition of sensors and activity classification can improve the cues generated by the reminder system and extend the system's functionality through the introduction of cues for therapy compliance. The integration of the component technologies demonstrates the feasibility of this approach. Although further work is required to demonstrate effectiveness in large deployments, we believe this approach can substantially improve the quality of life for many users.

Acknowledgments

We would like to thank Intel Research Seattle for providing the RFID bracelet.

References

Bai, T.; Modayil, J.; and Kautz, H. 2008. Improving the recognition of interleaved activities. In *submission*.

Boger, J.; Poupart, P.; Hoey, J.; Boutilier, C.; Fernie, G.; and Mihailidis, A. 2005. A decision-theoretic approach to task assistance for persons with dementia. In *Proceedings of International Joint Conference on Artificial Intelligence (IJCAI)*, 1293–1299.

Fisch, J.; Dowds, M.; Levinson, R.; and Bushnik, T. 2007. Current evidence-base for the efficacy of electronic aids for memory and organization. In *American Congress on Rehabilitation Medicine*.

Levinson, R. 1995a. A computer model of prefrontal cortex function. *Annals of the New York Academy of Sciences: The Structure and Function of Prefrontal Cortex* 769:381–388.

Levinson, R. 1995b. A general programming language for unified planning and control. *Artificial Intelligence* 76:319–375.

Levinson, R. 1997. The planning and execution assistant and trainer. *Journal of Head Trauma Rehabilitation* 85–91.

Liao, L.; Fox, D.; and Kautz, H. 2004. Learning and inferring transportation routines. In *Proc. 19th National Conf. on Artificial Intelligence (AAAI-2004)*, 348–353.

Mihailidis, A.; Carmichael, B.; and Boger, J. 2004. The use of computer vision in an intelligent environment to support aging-in-place, safety, and independence in the home. *IEEE Transactions on Information Technology in BioMedicine* 8(3):238–247.

Patterson, D.; Fox, D.; Kautz, H.; and Philipose, M. 2005. Fine-grained activity recognition by aggregating abstract object usage. In *Proceedings of the IEEE International Symposium on Wearable Computers*.

Pollack, M.; Brown, L.; Colbry, D.; McCarthy, C.; Orosz, C.; Peintner, B.; Ramakrishnan, S.; and

Tsamardinos, I. 2003. Autominder: An intelligent cognitive orthotic system for people with memory impairment. *Robotics and Autonomous Systems* 44:273–282.

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

Tapia, E.; Intille, S.; and Larson, K. 2004. Activity recognition in the home using simple and ubiquitous sensors. In *Pervasive*, 158–175.