chapter four

Pervasive computing in the home and community

Donald J. Patterson
University of California, Irvine, California
Henry A. Kautz
University of Washington, Seattle, Washington
Dieter Fox
University of Washington, Seattle, Washington
Lin Liao
Google Inc., Kirkland, Washington

Contents

4.1 Introduction

One of the major risks to independence for the elderly is the decline in the accomplishment of typical activities of daily living (ADLs) due to mild cognitive impairment (MCI), a precursor to Alzheimer's disease. (In this chapter, we use "ADL" to refer to basic ADLs [such as eating or dressing] as well as to instrumental ADLs [such as cooking or housework].) Lawton[1] measured the impact of impairment on instrumental ADLs and showed that the time spent on activities such as housework, shopping, and recreation declined 27 percent to 44 percent for impaired individuals. Furthermore, regardless of the cause of cognitive disabilities, research suggests that one of the best ways to prolong independence is to encourage the successful completion of ADLs.[2] A side benefit of such an improvement is an increase in the quality of life of family caregivers.

The term assisted cognition (AC)[3] has been coined to describe systems that use sensor data to determine the activities that a person is trying to perform and optionally provide prompts, warnings, or other kinds of interventions to help the person perform the activities safely and independently. Research in AC combines ideas from sensor networks and ubiquitous computing, artificial intelligence (AI), and human–computer interaction (HCI).

The number of sensors that can be used to track human activity, both in the marketplace and deployed in the environment, is quickly growing. They include familiar consumer-grade technologies like global positioning systems (GPS), Wi-Fi, and radio frequency identification (RFID) tags, as well as more specialized technologies such as MICA motes (small low-power CPUs optimized for sensing). Research in AI, some of which is described in this chapter, has developed methods for interpreting noisy sensor data in terms of hierarchical models of subjects' physical actions, activities, goals, and cognitive states. Work on HCI has shown that people with cognitive disabilities can effectively use interfaces that present simple choices and instructions and that employ simple text and meaningful photographic images.[4,5,6]

AC has the potential to be useful for many types of assistance:

- Logging: Logs of ADL performance provide valuable data for healthcare professionals. Creating logs automatically would be more comprehensive and less expensive than relying on manual observations.
- Rating and trending: A system that provides direct assistance to a subject must be able to rate how well an activity is performed. Evaluating how performance changes over time, even in subtle ways, provides information about cognitive decline.[7]
- Guidance: Providing immediate feedback about the results of evaluation could help an individual successfully complete difficult ADLs. Guidance strategies must be carefully designed to avoid increasing the subject's cognitive load and should take into account the probability and associated costs of providing incorrect guidance.
- Actuation: In some situations, an AC system should simply perform an activity for the user, in order to improve immediate safety and security; for example, it might lock the doors at night after the user has gone to sleep.

This chapter will review academic and commercial research on assisted cognition systems and then consider two projects from the University of Washington in detail: an outdoor navigation system called Opportunity Knocks and an indoor ADL tracking system called Barista.

4.2 Overview of research on assisted cognition

There are a number of research groups that have progressed in developing assistive technology with an element of cognitive reasoning.

4.2.1 Location, navigation, and wayfinding

Location-sensing technologies, which are surveyed and evaluated in reference 8, are the foundation for wayfinding systems. More recently, the Place Lab initiative[9] is a project designed to make outdoor Wi-Fi localization ubiquitous through mass collaboration. A single sensor can be augmented with a user model, possibly learned, to improve its accuracy.[10,11,12,13,14,15] Complementary to this work, Liao et al.[16] presented a discriminative model to automatically classify significant places and activities based on the framework of relational probabilistic models.[17]

A popular class of location-aware applications is the tour guide,[18] such as Campus Aware[19] and GUIDE.[20] Such tour guides are precursors to the kind of wayfinding system with which we are concerned.

4.2.1.1 Nursebot

Nursebot is a project at Carnegie Mellon University that provides a robotic platform for delivering navigation assistance to the elderly. This robot is envisioned to operate under the auspices of a community living home and helps users make it to appointments on time and provides directions to get there as well.

Nursebot requires caregiver support to update its understanding of the world: for example, the remote location of people and the state of their schedules. Because it is a robotic platform it only makes use of sensors embedded on the robot, although in principle extending the robot's knowledge to a network of distributed sensors would be possible.

Nursebot's navigation is based on robotic mapping technology and laser-range finders. Potential destinations are identified on a known map. When an elderly individual indicates a potential destination, Nursebot plans a route to the location and executes the plan. The plan is updated according to real-time laser-range finder inputs, which help to regularly update the position of the robot and the robot's knowledge of the people in the environment.

4.2.1.2 IMP

Closely related to Nursebot is IMP,[21] a walker that is augmented with a laser-range finder and navigational reasoning. When a user wants to go somewhere in the mapped facility, he or she can indicate the destination on an attached computer and a path-planning algorithm will guide the user to the destination using an arrow. Onboard sensors monitor progress and assist the user in getting to the destination.

One of the design decisions that this system made was to navigate a person directly through the use of a displayed arrow. This puts a high burden on the navigational system and sensor suite to avoid leading users into dangerous environments that cannot be sensed by the walker. This is probably not a large concern for controlled environments such as nursing homes but would be a problem for outdoor transportation assistance.

4.2.2 Wandering alert systems

A number of companies have attempted, with varying success, to create wandering alert systems. Some have been stand-alone systems, such as Digital Angel, and others have been integrated into smart assisted living environments, such as Elite Care assisted living homes. The company Independent Living has an installable system that promises to monitor both ADLs and wandering and alert a caregiver when programmed parameters are exceeded.

4.2.3 Home ADL tracking and support

There are many research groups that apply ubiquitous and wearable computing to the goal of aging in place.

4.2.3.1 Smart homes

One of the first and best-known smart home projects is Georgia Tech's Aware Home.[22] This project experimented with many technologies for capturing low-level sensor data about human activities but did not focus on automated activity recognition.

The MIT House_n project developed an instrumented condominium for studying activity recognition in a naturalistic environment.[23] This space is designed as a living laboratory from which experiments in activity recognition can be conducted in a naturalistic manner. From this line of research a variety of activity recognition research has been published that explores how simple sensors can recognize activities as they are being performed.[24,25]

The University of Florida's Mobile and Pervasive Computing Laboratory has several projects that are directed at cognitive assistance for the elderly, including meal preparation assistance and preliminary research into using cell phones for cognitive assistance.[26]

The Bath Institute for Medical Engineering developed a number of prototype commercial applications of cognitive devices and collaborates with Dementia Voice, a dementia services center for the southwest of England, and Housing 21, a U.K. housing association. Some of their projects include a cooker monitor (an instrumented stove that monitors for dangerous situations such as gas leaks, smoke, or burning pans); a misplaced-object finder; and a tap monitor (an instrumented faucet that controls temperature and prevents flooding).

4.2.3.2 COACH

The COACH (Cognitive Orthosis for Assisting aCtivities in the Home) system aims to address all aspects of ADL performance, from recognition to guidance, for the specific tasks of bathroom activities.[27] The current system has been tested for assisting people with advanced dementia with handwashing.

COACH is an adaptive device that learns by using Markov decision processes for how best to guide a user through the process of washing hands. The sensor input for this task is an overhead camera located over the sink, which is processed primarily for the location of relevant objects and then becomes the source of information for the decision process. This work is noteworthy for its use of verbal prompts that increase in specificity as the user becomes less and less likely to achieve the goal of handwashing. Of all the systems mentioned here, it targets individuals with the most severe forms of dementia.

4.2.3.3 Autominder

Autominder[28] is a planning assistance system that is designed to help a user meet scheduling goals for day-to-day activities. Its main reasoning component is a temporal constraint satisfaction engine that can determine when activities should be performed in order to avoid conflicts with other activities. Autominder's prompting module tries to minimize the number of times it interrupts the user. For example, if the user is scheduled to perform two activities at around the same time (for example, taking meds and brushing teeth), Autominder will combine the prompts.

4.2.3.4 PDA-based reminding systems

PEAT[29] is a commercial product that has many of the same goals as Autominder. It is built on a PDA platform and its goal is to help individuals who experience cognitive difficulty when formulating and following plans. It allows a caretaker to enter information about the time and steps of each task to be performed and then uses sounds and graphical alerts to guide the user through the tasks. The user manually clicks the PDA after each step; no automated activity recognition is attempted.

The AbleLink company does research and production on practical computer-based systems for supporting people with cognitive disabilities. Like PEAT, AbleLink has developed and tested PDA-based task-prompting systems, mainly for users with mental retardation.[5]

4.2.3.5 Wearable activity recognition systems

There has been much recent work in using wearable sound, video, and acceleration sensors and computers for activity recognition, both individually and combined. Later we describe work on activity recognition using RFID-tagged household objects and a wearable tag reader that led to the Barista system.[30] A theme of much of this recent work is that "heavyweight" sensors such as machine vision can be replaced by large numbers of tiny, robust, easily worn sensors.[31]

4.3 Opportunity Knocks: Assisting outdoor navigation

We now turn to detailed case studies of two prototype assisted cognition systems, beginning with the navigation system Opportunity Knocks (hereafter OK). Many ADLs, such as working, shopping, going to a doctor's office, or attending social events, require a person to move throughout his or her community. Individuals with cognitive disabilities must generally rely on rides from caregivers, use point-to-point taxi or shuttle services, use public transportation, or restrict their movements to places reachable by foot. The first two are often unavailable or affordable, and few cities are so compact that they can be easily traversed by foot, so we will concentrate on public transportation.

Public transportation provides a variety of cognitive challenges, such as remembering transit schedules; getting on the correct bus or train; determining when to get off; making changes between vehicles; and recovering from errors. These challenges are so great that many cognitively disabled individuals become housebound. However, if impaired individuals had effective compensatory cognitive aids to help them use public transportation, their independence and safety would improve, they would have new opportunities for socialization and employment, and stress on their families and caregivers would be reduced.

This idea of a personal navigation aid is substantially different from current commercial GPS navigation systems. Today's personal GPS devices are typically optimized for a particular mode of transportation, such as driving in a car or hiking cross-country. The personal navigation aid we propose would help a user perform complex transportation plans that involve moving between modes of transportation—for example, walking to a bus stop, riding on a bus, and then walking again. The system not only tracks the person's location, but also the mode of transportation and the status of the transportation plan.

Additionally, because our target audience is cognitively disabled, this solution should not require a user to explicitly program a device or to always take the initiative in using it. We will describe a system that learns a user's pattern of public transportation use, predicts the user's current transportation goals, infers user errors, and provides proactive assistance.

4.3.1 A usage scenario

In order to ground our system, we present a running example that will help illustrate the most important features of OK. The steps of the scenario are illustrated in Figure 4.1. John works at a cafeteria at the university. One day he leaves work to go home and is momentarily confused about where to go. He consults his OK system, which is running on his GPS-enabled cell phone. OK offers images of four destinations that he typically travels to after work: home, his doctor's office, and the homes of two of his friends (A). OK has learned two different plans for John to get home from work: he can walk to a bus stop and catch a bus home, or he can walk to the parking lot where he gets a ride with a work colleague. Because OK is uncertain about which plan is correct for the day, it asks John to choose between the two (B). He selects the bus icon, and OK provides walking directions to the bus stop (C) and instructs him to get on the #17 bus. Unfortunately, John erroneously gets on the #19 bus, which initially travels along the same route as the #17, but which ultimately goes to John's friend's house, rather than to his own home. However, once the #19 departs from the common portion of the route, OK recognizes the error (D). The system alerts John and immediately constructs a repair plan to get him home, using its general knowledge of the transit system (E). OK guides him to exit the bus, walk to a nearby stop where he can board the #17, and resume his journey home (F).

4.3.2 System architecture

Figure 4.2 diagrams OK's overall system architecture. The data flow of the system starts at a sensor beacon that is carried by a user. The sensor samples the environmental context of the user and forwards this information over a secure Bluetooth connection to the cell phone. The cell phone initially acts as a network access point and again forwards the context information to a remote server over the high-speed General Packet Radio System (GPRS) data network. The remote server, which is running the OK software, uses the sensor information in conjunction with Geographic Information Systems'

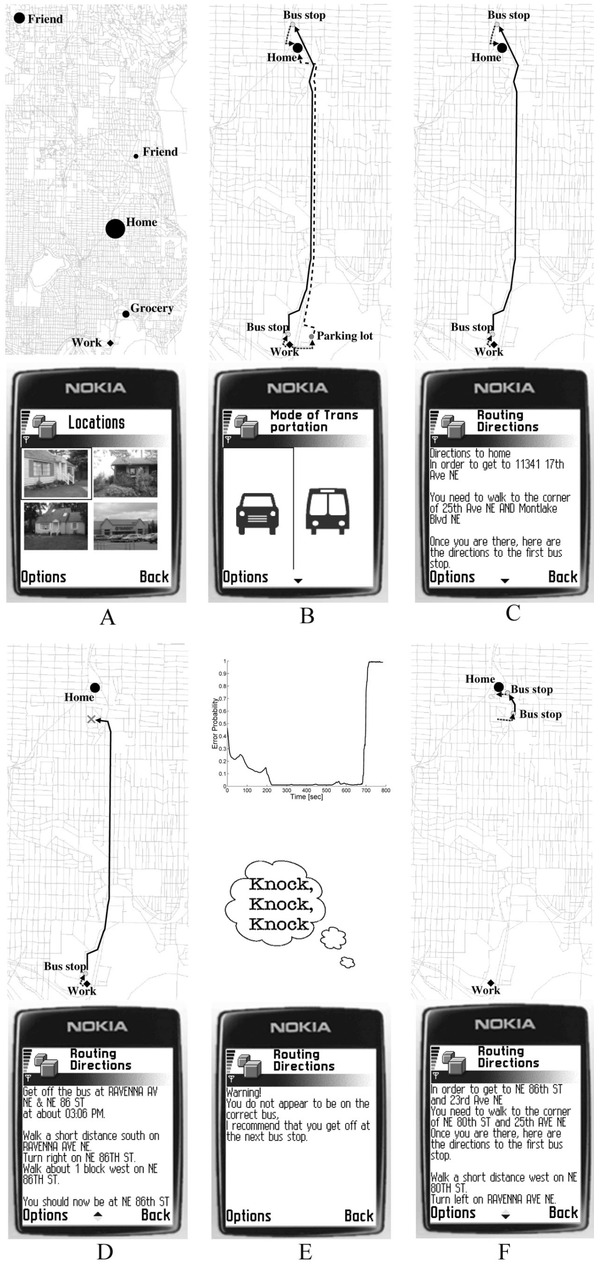

Figure 4.1 Scenario of Opportunity Knocks in action (see text for description).

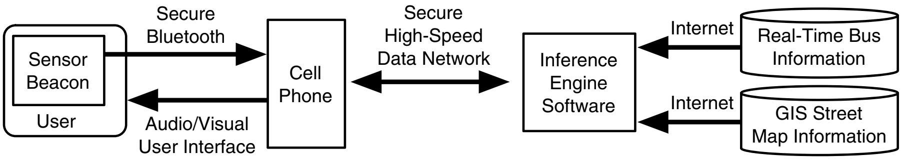

Figure 4.2 Architectural diagram of Opportunity Knocks.

(GIS) databases to localize the user. When the software has sufficient confidence in the position of the user, it is then able to suggest opportunities about which the user may want to know. These opportunities are sent back to the cell phone for display through the user interface. If an urgent opportunity, such as a plan for recovering from boarding the wrong bus, is recognized, the phone proactively alerts by making a door-knock sound; otherwise the phone remains passive with information available for reference by the user. If the user selects an opportunity, such as a route to a frequent destination, the cell phone requests supporting information from the server, which may require referencing real-time information about bus schedules.

We chose a cell phone as the client hardware because of its role as a de facto standard for a portable computing device. It has inherent value that is related to its primary function as a phone and for many people it is as common to carry as a wallet or a purse. As a result, it is likely to be a familiar, nonstigmatizing method of delivering assistive services. In the cell phone market products span from a traditional phone to a personal digital assistant (PDA). We opted for devices that were more like traditional phones rather than smartphones because of their ubiquity, simple interface, and limited maintenance requirements.

The system currently uses a Nokia 6600 cell phone. The Nokia 6600 phone is a GSM phone that has a wide range of features required by OK. First it supports the J2ME mobile information device profile (MIDP) 2.0 that provides support for secure networking, serial port connection support, and the application management system—a push registry that enables authorized applications to be launched remotely. Some model-specific features of the phone that we utilize include a high-resolution (176 × 208 pixels), high-color (16-bit) screen, a digital camera, Bluetooth support, and high-speed data network capabilities (GPRS).

When the user desires transportation assistance, he refers to the phone and observes up to four images of predicted destinations (later we describe how this selection is made). If he would like to go to one, he selects it. If the system has observed the user going to this destination in different ways (for example, by foot or by bus) it will prompt him for the method he would like to take. The previously observed route is then provided in text form. The system will not present destinations to which the user hasn't previously traveled, but it will allow the user to select a familiar destination even if it has never observed the user getting there from the current location. In this case OK presents a route that is based on a real-time bus route–planning service provided by the local transit authority. In the course of this interaction the user did not have to provide any information about where he was, and only a very small amount of information about where he wanted to go, yet the system was still able to route him effectively.

There are two occasions when the phone might become proactive and make a knocking sound. The first is when the system has high confidence that a novel or erroneous event has occurred. The second is when the system identifies that the user is at a new significant location and may wish to photograph it using the phone's camera. In the future, whenever the system wants to refer to that location, it simply uses the photo to identify the spot. Then the user does not need to input or recognize GPS coordinates or street addresses. A further advantage of this approach is that the user can decide what is visually meaningful about the location.

The current prototype system does not make use of voice prompts. Using voice is an obvious extension and is being incorporated in the next version of the system.

4.3.3 The inference engine

The inference engine that drives OK must learn and reason about the user's movements. As outlined earlier, the system must learn about its user's transportation routines in an unsupervised and unobtrusive manner, be able to predict likely destinations the user may want to go to at any given moment in time, and be able to recognize anomalous behavior. Because of the inherent uncertainties about human behavior as well as the possible errors from maps and GPS measurements, OK must reason probabilistically.

Ashbrook and Starner[11] have proposed using a second-order Markov model as a predictive tool for reasoning about likely destinations toward which a user may next travel. In contrast to our desired behavior, this model is not able to refine estimates of the current goal using GPS information observed when moving from one significant location to another. Because significant locations might be long distances away, this causes an unacceptable lag in noticing unusual behavior and significant amounts of GPS information are disregarded.

Patterson et al.[13] proposed using a dynamic Bayesian network (DBN) for inferring a user's transportation mode and location on a street grid from GPS data. Although this approach provided accurate tracking, it had poor predictive power, because the model had no representation of a user's destinations or transportation plans; information about the user's patterns of movement was only recorded as the probability of the user turning in a particular direction at each street corner.

In order to overcome these limitations, OK employs the new hierarchical DBN model representing transportation routines introduced by Liao et al.[32] The new model subsumes the capabilities of the previous models and bridges the gap between the raw sensor measurements and the abstract goal intentions of a user. A brief discussion of this model follows; refer to reference 32 for full technical details of the model structure, inference, and training.

Figure 4.3 shows the graphical structure of the new model. At the very highest level of this model, goals g_k (subscript k indicates the discrete time step) are explicitly represented as significant locations. Transitions between goals have specific probability distributions independent of the routes by which they are reached. Each goal destination influences the choice of which trip segment the user takes. Trip segments are sequences of motion in which the transportation mode is constant. Each trip segment t includes its start location t_{sk}, end location t_{ek}, and the mode of transportation t_{mk} the person uses during the segment. Each trip segment biases the expectation over the mode of transportation and the changes in location. The mode of transportation m, in turn, determines the location and velocity distribution of the user. At the bottom level, we denote by $x_k = <l_k, v_k>$ the location and motion velocity of the person. Edge transition τ_k indicates the next street when passing an intersection and data association θ_k "snaps" a GPS measurement onto some streets around it. The switching nodes f_{kg}, f_{kt}, and f_{km} indicate when changes in a variable's value can happen.

An efficient algorithm based on Rao-Blackwellised particle filters[33,34] has been developed to perform online inference for this model. At the lowest level, location tracking on the street map is done using graph-based Kalman filtering. At the highest level, the joint distribution of goals and trip segments is updated analytically using exact inference techniques. As a result, this

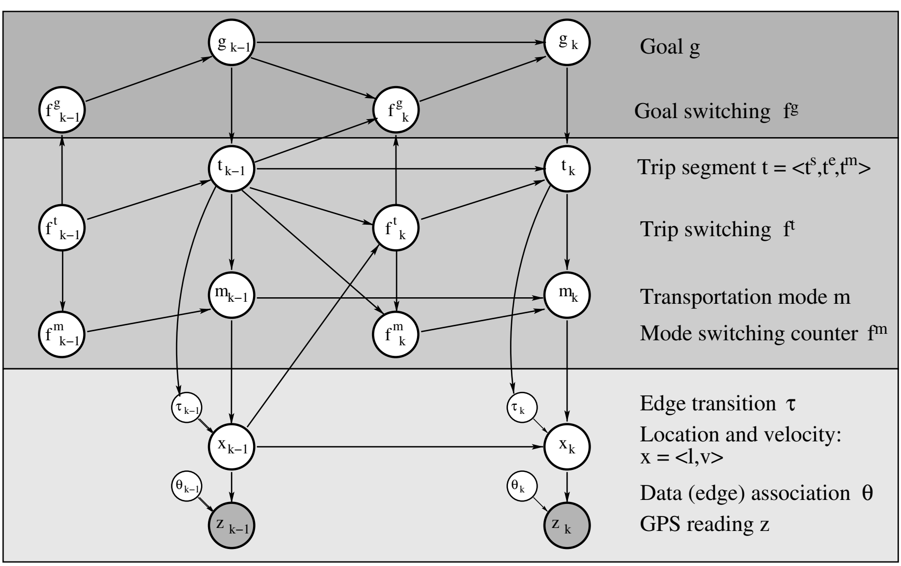

Figure 4.3 Hierarchical activity model representing a person's outdoor activities. The top level estimates the current goal, the middle layer represents segments of a trip and mode of transportations, and the lowest layer estimates the person's location on the street map. Figure reproduced from reference 32.

model makes it possible to reason about high-level goals (or significant locations) explicitly. The contribution of this model is that it considers not only previous significant locations visited but also the current location and the path taken so far to reason about likely destinations.

The parameters in the model are estimated in an unsupervised manner. This is a three-step process. In a first pass through the data, the possible goals for a user are discovered by observing when the user stays at a location for a long time. Then in a second pass, the usual parking spots and bus stops are inferred using an expectation-maximization algorithm.[35] Finally, the transition matrices at all levels are reestimated simultaneously using a second expectation-maximization procedure with the full model. The learning process does not require any labeled data and therefore requires no intervention from the user.

To detect abnormal events, the approach uses two models with different transition parameters. The first tracker assumes the user is behaving according to his personal historical trends and uses the learned hierarchical model for tracking. The second tracker assumes a background model of activities and uses an untrained prior model that accounts for general physical constraints but is not adjusted to the user's past routines. The trackers are run in parallel, and the probability of each model given the observations is calculated. When the user follows his ordinary routine the learned hierarchical model should have a higher probability, but when the user does something unexpected the second model should become more likely. To compute the probability of each model, we use the concept of Bayes factors, which are standard tools for comparing the quality of dynamic models based on measurements.

The above approach can detect unexpected events but cannot distinguish errors from deliberate novel behavior. An important contribution of OK, however, is the ability to differentiate these cases using knowledge of the user's destination. This is possible because there are times when the system knows where the user is going: for example, if the user asks for directions to a destination, if a caregiver or job coach indicates the "correct" destination, or if the system has access to a location-enabled datebook. In those situations we can clamp the value of the goal node in our model and reinterpret the low-level observations. When the observations diverge significantly from the clamped high-level predictions, the system is able to signal a possible error.

This model can spot anomalous behavior even if the user follows a well-trodden path, provided that path does not lead to the specified destination. For example, in the scenario described above, OK determines that John is on the wrong bus to get home, even though he sometimes does take that bus to go to his friend's house. A small graph illustrating this example of error detection when there is a clamped goal appears in step E of Figure 4.1.

4.3.4 Status

OK was built as a proof-of-concept prototype of an assisted cognition navigation system. It was successfully tested on real data gathered by students enacting various scenarios using models learned by using the system for approximately two hours a day for three weeks. However, the prototype was not robust enough for clinical trials. A particular problem of the architecture is that it required a continuous (wireless) Internet connection to a central computer server where inference was performed.

OK is currently being reengineered to be more robust and to perform inference on a Windows CE PDA carried by the user. Development leading to clinical trials for subjects with navigation difficulties due to brain injuries is being supported by the National Institute for Disability and Rehabilitation Research.

4.4 Understanding home activities

We now turn to our second case study with the assisted cognition system Barista. Indoor activity recognition has many potential benefits. Tracking the performance of ADLs is the first step in creating systems for ADL prompting and guidance. Furthermore, ADL monitoring is an ongoing, important activity in healthcare. For example, in the United States, any nursing home that receives Medicare funds has to record and report ADLs. Trained caregivers spend a great deal of time measuring and tracking ADL accomplishment for persons under their care. However, manual monitoring is time-consuming, error prone, and invasive. Automated aids that can address these issues and reduce the record-keeping burden on caregivers are of great interest.

Most systems that have been built to recognize home activities have been limited in the variety of activities they recognize, their robustness to noise, and their ease of use. In particular, most previous work on activity recognition has used sensors that provide only a very coarse idea of what is going on—for example, by detecting movement in a room, one might infer that an activity associated with that room is happening.[36] Also previous work required deployment of an extensive custom-sensing apparatus to monitor each task[27,37,38,39,40] or relied on solutions to deep technical problems such as machine vision.[41,42]

In this section we describe an approach to activity recognition that addresses these problems by combining the use of wearable RFID tag sensors to determine when a user is manipulating physical objects, with a simple and flexible probabilistic framework for modeling activities in terms of object touches.

4.4.1 Sensing using RFID

RFID tags are the size of postage stamps (including adhesive backing), have no batteries, and can withstand day-to-day use for years. A tag reader sends a radio frequency pulse to the tag, which responds with a unique identifier. Depending on the power of the reader, a tag can be sensed from a few inches to several yards away.

RFID deployment involves tagging tens to hundreds of objects in the environment, then entering each tag identifier into a database. This can be done incrementally; the more tags there are, the broader and deeper the potential coverage of ADLs. Furthermore, market forces are pushing toward the near-universal use of RFID tags on essentially all products. Such preexisting tags could then be used for applications such as ours by using a database to map tag IDs to types of objects.

Recently Intel Research and other companies have begun to develop small wearable tag readers. Such readers can determine when a wearer touches a tagged object. The work described in this chapter used a short-range RFID reader built into the palm of a glove, with a Crossbow Mica Mote radio, a USB-based power supply, and a rechargeable battery. Intel Research has more recently developed a reader in the form of a small bracelet.

ADLs that would be difficult or impossible to detect using either coarse location sensors or state-of-the-art machine vision can often be recognized on the basis of contact with a tagged object. For instance, consider trying to determine if a person is reading. Location alone is clearly inadequate, while reliably recognizing the act of "reading" from a video stream under a wide range of orientations, positions, and lighting conditions is far beyond the capabilities of machine vision for the foreseeable future. On the other hand, if all the books, magazines, and newspapers in the home were tagged, determining when a person was reading could be done quite reliably.

Although data from RFID tags are less noisy than many other kinds of sensor data, any real-world data streams still contain extraneous readings (for example, when the user's hand happens to brush by a tagged but unused object) and missing readings. Therefore we propose to interpret the data using probabilistic models that are robust in the face of noisy data.

4.4.2 Modeling activities

Representing ADLs in terms of the gross manipulation of physical objects requires us to face the problem of developing a formal model that satisfies a number of constraints: first, the model should easily express significant properties of and distinctions between activities, while remaining robust to unimportant variations in activity performance; second, the parameters of the model should be easily estimated; and third, the model should be implementable in a manner that supports efficient and scalable inference.

Descriptions of activities from a wide variety of sources, including healthcare literature, instructional manuals, and recipes typically break an activity down in a set of steps, where each step involves manipulating one or more objects over some period of time. Although textual descriptions usually present the steps in a total order, the underlying logical dependencies between steps often form only a partial order and include alternative and optional steps. The kinds of objects used in a step are usually flexible, and it is not difficult to form a coarse estimate on the probability of object use on the basis of the description. For example, while making a cup of tea (used as a running example in this section), we might estimate that the probability of using a spoon to stir the tea is 75 percent, allowing for cases where one uses a different utensil or none at all.

Figure 4.4 gives an example of modeling the activity "making tea" in four stages: getting out the supplies, heating the water, steeping the tea, and flavoring (i.e., adding sugar or lemon to the tea). The first stage consists of two steps that must both occur but in any order: this is indicated by a conjunctive arc across the first pair of outgoing arrows and the following pair of incoming arrows. This is an example of a partial ordering constraint. The disjunctive choice of which of two ways to heat the water (using the microwave or using the stove) is indicated by a set of plain arcs. The fact that the flavoring step is optional is represented by a disjunctive arc that bypasses the step.

Each step also has a Gaussian duration. Duration information can provide important constraints for distinguishing activities that use similar objects. For example, washing your hands at the kitchen sink takes about a minute, while washing dishes at the kitchen sink takes about ten minutes. Finally, each step includes a set of objects that is expected to be used. (In Figure 4.4 the duration and object information are only shown for the "boil water" step.) The value associated with each object is termed an object use probability and is the estimate of the probability that the object is manipulated at least once before the step completes. Also included in the model but not shown in the illustration are prior probabilities on each activity as a whole and on choice transitions within a model (such as the probability of including the optional "flavor tea" step); by default these are uniform across choices.

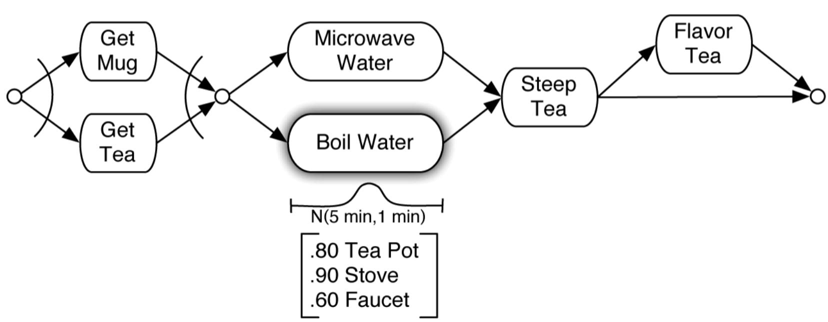

Figure 4.4 Making tea represented as an activity graph.

4.4.3 Coarse-grained ADL recognition

Matthai et al.[30,43] demonstrated that RFID sensors and probabilistic models of the form described earlier could reliably distinguish the performance of a large number of common ADLs. They chose fourteen ADLs to monitor, such as using the toilet, completing housework, using an appliance, taking medication, etc. (It is notable that the fourteen ADLs tested were eleven more than any other system has attempted.) One of the experimenter's homes was instrumented with 108 tags. Next, an activity model was created by hand to relate ADLs to tagged objects, using rough prior estimates of the timing and probability of object use, and a uniform prior estimate over all transitions from the end of one activity to the start of another.

Fourteen test subjects were each asked to perform the activities in any manner they liked and in any order, while wearing an RFID tag-reader glove. Activities took from twenty to sixty minutes, depending on the subject. A particle-filter inference engine computed the most likely sequence of activities performed by each subject, which was compared to a manual log of the actual sequence. The system did well on average, with a precision of 88 percent (correct labeling of recognized activities) and recall of 73 percent (percent of actual activities correctly recognized). To place these numbers in further context, we note that this was the first time any system had been created that could handle nine of the fourteen ADLs.

4.4.4 Barista: Fine-grained ADL recognition

Although some of the ADLs in the work described above used some objects in common, most could be uniquely distinguished by the manipulation of particular objects. The purpose of the Barista system was to determine if the same general approach could be used for fine-grained tracking of interleaved activities that shared many common objects.[44] In particular, we wished to find the simplest and most robust modeling methodology.

Barista focused on a morning routine in a small home. The following eleven activities were considered:

1. Using the bathroom
2. Making oatmeal
3. Making soft-boiled eggs
4. Preparing orange juice
5. Making coffee
6. Making tea
7. Making or answering a phone call
8. Taking out the trash
9. Setting the table
10. Eating breakfast
11. Clearing the table

To create the data set, one of the authors performed each activity twelve times in two contexts: Each activity was performed by itself twice, and then on ten mornings all of the activities were performed together in a variety of patterns. In order to capture the identities of the objects being manipulated, the kitchen was outfitted with sixty RFID tags placed on every object touched by the user during a practice trial. For example, in the bathroom the door knob, toilet handle, and faucet handle were tagged. In the kitchen, tags were placed on appliances, cookware, dishes, and food packages. Deploying the tags required less than two hours.

In this experiment the user simultaneously wore two RFID gloves (unlike in the first experiment in which one glove was used). The time and ID of every object touched was sent wirelessly by the glove to a database for analysis. The mean length of the ten interleaved runs was 27.1 minutes (σ = 1.7) and object touches could be captured at approximately ten per second. The mean length of each uninterrupted portion of the interleaved tasks was seventy-four seconds. Most tasks were interleaved with or interrupted by others during the ten full data-collection sessions.

The activities were not performed sequentially or in isolation from each other. Whenever there was a pause in an activity, progress was attempted in other activities (such as when waiting for water to boil) and some activities interrupted others at uncontrolled times (such as answering the phone).

4.4.5 Modeling choices

In order to justify the inference model that we ultimately developed we proceeded systematically by first focusing on accuracy and then on robustness. We developed the simplest possible probabilistic model, evaluated its performance, and then augmented it with features that were sufficient to disambiguate errors. In this section we present the techniques we used to improve accuracy by describing the two baseline models and another model that incorporated reasoning with aggregate features. The models increase in complexity by adding representational power. In subsequent sections we present abstraction techniques that we used to improve robustness.

The first baseline model consists of independent, single-state hidden Markov models (HMMs) for each activity. Used in a generative context, each state emits an object-X-touched event or a no-object-touched event at each tick of the clock. Each state's emission probability was trained on the twelve examples of a user performing the corresponding activity. After training, the probability of emitting a no-object-touched event was equalized across all HMMs so that the timing characteristic of the model was completely captured by the self-transition probability. To infer the activity being performed at each second, each HMM was presented with a seventy-four-second window of data (the average activity duration) ending at the query second. This produced a log-likelihood value for each model at each tick of the clock. The activity model with the highest log likelihood was used as the system's estimate of the current activity. This model was trained and tested on data in which object types were equalized so that there was no distinction made between spoon #1 and spoon #2, for example, but both appeared identically as a "spoon."

The second baseline model connected the states from the eleven independent HMMs of baseline A in order to be able to learn about and subsequently smooth the transitions between activities. We retrained this HMM using the ten examples of the user performing the eleven interleaved activities. The no-object-touched emission probability was again equalized across all states. This HMM was evaluated over the entire data window, and the Viterbi algorithm[35] was used to recover the activity at every time point given by the maximum likelihood path through the state space. Again, this model was trained and tested on data in which object types were equalized to eliminate distinctions between instantiations of objects.

For our third model, we chose to examine the effect of reasoning about aggregate information. The specific feature that we wanted to model was how many objects of a given type were touched during the course of the current activity. This aggregate can only be computed if globally unique object instances can be identified. This choice was motivated by the desire to differentiate activities that use the same object repeatedly from those that use many different objects of the same type. For example, consider the activities of setting the table and eating breakfast. The first involves single touches of several different plates, spoons, and cups, while the latter involves touching the same plate, spoon, and cup repeatedly.

Aggregate features can be handled in a DBN model by introducing variables that keep track of how many different instances of each object type are touched during the performance of an activity, in addition to a global variable whose value is the current activity. At the end of an activity (i.e., the value of the global variable changes), the final counts are treated as pseudo-observation. See Patterson et al.[44] for details.

The various features of these models are summarized in the following table:

	Exponential Timing Distribution	Interactivity Transitions	Aggregate Information
Independent HMMs			
Connected HMMs	✓	✓	
Aggregate DBN	✓	✓	✓

In this table, exponential timing distributions refer to the fact that the model expects the length of an uninterrupted portion of an activity to occur with a duration that is distributed according to an exponential distribution. The parameters of the distribution are learned from the data. This timing distribution is a result of the structure of the HMMs and DBNs used. Interactivity transitions refers to the ability of the model to represent the tendency of certain activities to follow or interrupt other activities more or less often.

Finally, aggregate information refers to the ability of the model to represent aggregations over individual objects.

4.4.6 Accuracy experiments

Our accuracy experiments were conducted with leave-one-out cross-validation across the ten interleaved runs. We calculated two accuracy metrics. The first was what percentage of the time the model correctly inferred the true activity. This metric is biased against slight inaccuracies in the start and end times and will vary based on the time granularity with which the experiments were conducted. We also evaluated our models using a string edit distance measure. In this case we treated the output of the inference as a string over an eleven-character alphabet, one character per activity, with all repeating characters merged. We calculated the minimum string edit distance between the inference and the ground truth. A string edit distance of one means that the inferred activity sequence either added a segment of an activity that didn't occur (insertion), it missed a segment that did occur (deletion), or it inserted an activity that didn't occur into the middle of an activity (reverse splicing). A perfect inference will have a string edit distance of zero. The string edit distance is biased against rapid changes in the activity estimate and is tolerant of inaccuracies in the start and end times of activities. The following table summarizes the results of the experiments:

	Time-Slice Accuracy (σ)	Edit Distance (σ)
Independent HMMs	68% (5.9)	12 (2.9)
Connected HMMs	88% (4.2)	9 (6.2)
Aggregate DBN	87% (3.1)	7 (2.2)

The independent HMM model performed badly because it rapidly and inaccurately switched between activities. The smoothing provided by the connected HMM model gave much better accuracy. However, Figure 4.5 shows that the connected HMM model confused the activity "eat breakfast" with the activity "clear the table." The aggregate DBN distinguished these properly and overall had a slightly improved accuracy and much better edit distance error measure.

4.4.7 Improving robustness

One of the concerns with the previous models is how well they will respond if someone used an object of a type that did not appear in the training data but was functionally similar to objects that did appear. For example, in our model we cooked oatmeal using a cooking spoon. Our inference should not fail if the user performed the same task using a tablespoon. Likewise, if the user makes tea in a cup rather than a mug, that should be a less likely but still plausible alternative. To solve this problem we introduce the concept of abstraction smoothing.

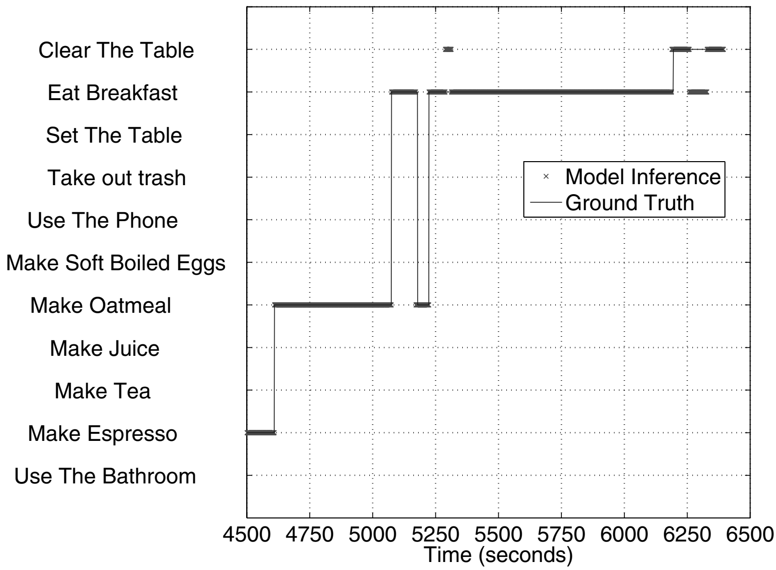

Figure 4.5 Sample connected HMM results. Ground truth is indicated by the thin line. Inference is indicated by the dots.

In order to perform smoothing over objects we created a relational model inspired by Anderson and Domingos.[45] Unlike the full power of that work, however, we used a single hierarchical object relation rather than a lattice. The hierarchy that we used was mined with supervision from the Internet shopping site Froogle (see Figure 4.6). The name of each object was entered into the shopping search engine and the hierarchy that was returned for that object was inserted into the global object tree. In the case of objects with multiple hierarchies, one was manually selected.

The semantics that we applied to the resulting tree were that objects that were close to each other in the graph were functionally similar. To specify the notion of "close," we weighted all edges on the graph equally and created an all-pairs functional equivalence metric according to the following formula,

$$P(O_i \Rightarrow O_j) = \frac{\exp\left(-\frac{Dist(O_i,O_j)}{2}\right)}{\sum_j \exp\left(-\frac{Dist(O_i,O_j)}{2}\right)}$$

where $Dist(O_i,O_j)$ is the shortest-path distance between O_i and O_j on the graph. This says that when object O_i is expected in the model, it will be substituted

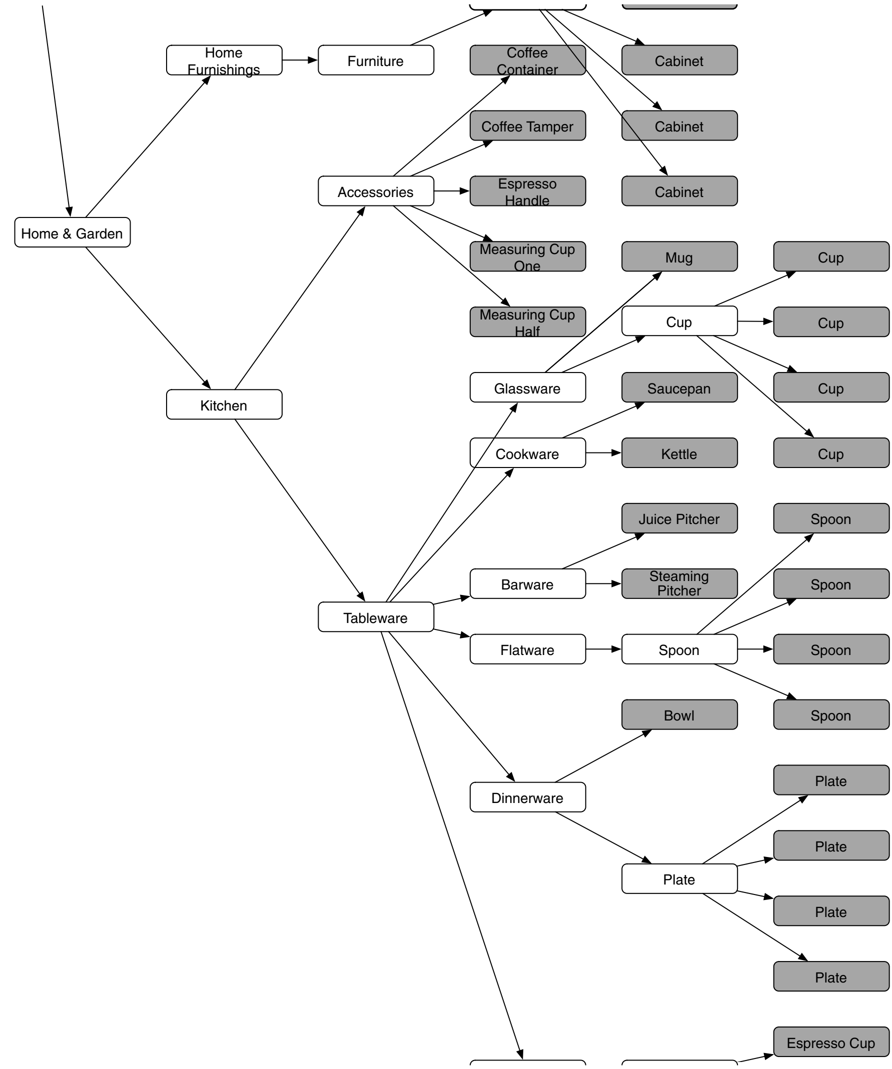

Figure 4.6 A portion of the object abstraction hierarchy mined from an Internet shopping site. Objects in our training data are shaded. Abstractions are not shaded.

by object O_j with probability $P(O_i \Rightarrow O_j)$. The likelihood of substituting one object for another falls off exponentially with distance in the hierarchy.

To validate how well this technique worked when objects were substituted, we reran our experiments with abstraction smoothing added to the aggregate DBN model. This resulted in an insignificant decrease in accuracy of 0.1 percent and –0.1 in edit distance.

Next, we reran our experiments with the same data streams, but with all instances of a particular object replaced by other instances of functionally similar objects of a distinct type. The following table shows our results for one scenario:

	Individual HMMs	Single HMM	Aggregates with Abstraction
Mean Accuracy	52.5%	77.4%	81.2%
Net Change	–15.1%	–10.9%	–6.4%
Mean Edit Dist.	24.7	35.6	8.8
Net Change	+12.7	+26.6	+1.1

Whereas abstraction smoothing doesn't greatly harm normal activity recognition, it greatly increases robustness to object substitution. The metrics from this table were generated by replacing a mug in all of the testing sequences with a cup. In two of the baseline models accuracy is dramatically lowered, but the abstraction model suffers a relatively modest decrease in accuracy, especially according to the string edit metric.

4.5 Summary

In this chapter we explored research on assistive technology for cognitive disabilities that combines advances in sensors and artificial intelligence to promote independence in the face of cognitive decline. By developing information systems that augment cognition in the same way that physical devices compensate for physical disabilities, we may be able to maintain higher quality of life for patients suffering from cognitive decline due to aging, trauma, or disease. At the same time, we can hope to reduce family caregivers' emotional and financial burden.

We looked at two possible avenues for assistance. The first is using an outdoor activity recognition system based on GPS to help people who make occasional cognitive errors recover safely. The second is an indoor activity recognition system based on a wearable computing platform and RFID tags that is designed to monitor which activities occur in a home.

In the outdoor case, we demonstrated that such a system could successfully be built now, that the reasoning that such a system can perform is both accurate and valuable, and that user–interface innovations make such a system usable without extensive user programming. In the indoor case, we saw that a single technology can subsume many previous activity recognition techniques in a way that is robust, easily deployable, and accurate at a fine level of detail.

An important challenge for work on assisted cognition systems is to develop and test effective user interfaces that decrease, rather than increase, a user's cognitive load. Opportunity Knocks began to explore the space of user interfaces, but the reliance on textual prompts would clearly be inappropriate for many users. The activity recognition system we described could be used as one part of a larger system that not only tracked activities but also provided prompts when necessary to help a user complete activities.

A significant complication in building interfaces for assisted cognition systems is that there is always the possibility that the system might misinterpret the data, so that what appears to be a user error might actually be a system (modeling) error. The problem of weighing the probabilities of system versus user errors, and the costs to the user of providing bad advice versus not providing good advice, is naturally formulated as one of decision making under uncertainty. Therefore we expect methods for solving such decision-theoretic problems to play important roles in future research in this area.

Acknowledgments

Preparation of this chapter was supported by grants from the National Institute on Disability and Rehabilitation Research (NIDRR), National Science Foundation (NSF), Defense Advanced Research Projects Agency (DARPA), and Intel. We thank Matthai Philipose for inspiration and technical leadership.

References

1. Lawton, M.P. Aging and Performance of Home Tasks. *Human Factors*, 32(5) 1990: 527.
2. Banaszak-Holl, J. et al. Predicting Nursing Home Admission: Estimates from a 7-Year Follow-Up of a Nationally Representative Sample of Older Americans. *Alzheimer Disease Association Disorders*, (2) 2004: 83.
3. Kautz, H. et al. An Overview of the Assisted Cognition Project, 2002. In *AAAI-2002 Workshop on Automation as Caregiver: The Role of Intelligent Technology in Elder Care*, 2002, 65.
4. Hopper, T., Bayles, K., and Kim, E. Retained Neuropsychological Abilities of Individuals with Alzheimer's Disease. *Seminars in Speech and Language*, 22(4) 2001: 261.
5. Davies, D.K., Stock, S.E., and Wehmeyer, M.L. Enhancing Independent Task Performance for Individuals with Mental Retardation through Use of a Handheld Self-Directed Visual and Audio Prompting System. *Education and Training in Mental Retardation and Developmental Disabilities*, 37, 2002: 209.
6. Inglis, E. et al. Issues Surrounding the User-Centred Development of a New Interactive Memory Aid. *Universal Access in the Information Society*, 2(3) 2003: 226.
7. Katz, S. Assessing Self-Maintenance: Activities of Daily Living, Mobility and Instrumental Activities of Daily Living. *Journal of the American Geriatrics Society*, 31(12) 1983: 721.
8. Hightower, J. and Borriello, G. Location Systems for Ubiquitous Computing. *Computer*, 34(8), 2001: 57.
9. Schilit, B., et al. Challenge: Ubiquitous Location-Aware Computing and the Place Lab Initiative. In *Proc. of WMASH 2003: The First ACM Intl. Workshop on Wireless Mobile Applications and Services on WLAN*, 2003, 29.
10. Krumm, J., Williams, L., and Smith, G. SmartMoveX on a Graph—An Inexpensive Active Badge Tracker. In *Proc. of the 4th International Conference on Ubiquitous Computing*, Springer-Verlag, 2002, 229.

11. Ashbrook, D. and Starner, T. Learning Significant Locations and Predicting User Movement with GPS. In *Proc. of the Sixth IEEE International Symposium on Wearable Computers*, Seattle, WA, October 2002, 101.
12. Cielniak, M., Bennewitz, G., and Burgard, W. Where is . . . ? Learning and Utilizing Motion Patterns of Persons with Mobile Robots. In *Proc. of the Seventeenth Intl. Joint Conf. on A.I.*, 2003.
13. Patterson, D.J. et al. Inferring High-Level Behavior from Low-Level Sensors. In *Proc. of UBICOMP 2003: The Fifth International Conference on Ubiquitous Computing*, A. Dey, A. Schmidt, and J.F. McCarthy, eds., Springer-Verlag, 2003, 73.
14. Schulz, D., Fox, D., and Hightower, J. People Tracking with Anonymous and ID-Sensors Using Rao-Blackwellised Particle Filters. In *Proc. of the Eighteenth Intl. Joint Conference on Artificial Intelligence (IJCAI)*, 2003, 921.
15. Hariharan, R. and Toyma, K. Project Lachesis: Parsing and Modeling Location Histories. In *Third International Conference on GIScience*, Aldephia, MD, 2004.
16. Liao, L., Fox, D., and Kautz, H. Location-Based Activity Recognition Using Relational Markov Networks. In *Proc. of the Nineteenth Intl. Joint Conf. on A.I.*, 2005, 773.
17. Getoor, L. et al. Learning Probabilistic Relational Models. In *Relational Data Mining*, S. Dzeroski and N. Lavrac, eds., Springer-Verlag, 2001, 307.
18. Abowd, G.B. et al. Cyberguide: A Mobile Context-Aware Tour Guide. *Wireless Networking*, 3(5) 1997: 421.
19. Burrell, J. et al. Context-Aware Computing: A Test Case. In *Proc. of the 4th International Conference on Ubiquitous Computing*, Springer-Verlag, 2002, 1.
20. Cheverst, K. et al. Developing a Context-Aware Electronic Tourist Guide: Some Issues and Experiences. In *Proc. of CHI 2000*, 2000, 17.
21. Morris, A.C. et al. A Robotic Walker That Provides Guidance. In *Proc. of the 2003 IEEE Conference on Robotics and Automation (ICRA '03)*, 2003.
22. Kidd, C.D. et al. The Aware Home: A Living Laboratory for Ubiquitous Computing Research. *Cooperative Buildings*, 1999, 191.
23. Intille, S.S. et al. The PlaceLab: A Live-In Laboratory for Pervasive Computing Research. In *Pervasive Computing, Third International Conference, PERVASIVE 2005, Video Program, Lecture Notes in Computer Science*, Springer-Verlag, 2005, 183.
24. Bao, L. and Intille, S.S. Activity Recognition from User-Annotated Acceleration Data. In *Proc. of Pervasive Computing, Second International Conference, PERVASIVE 2004*, A. Ferscha and F. Mattern, eds., Vienna, Springer-Verlag, 2004, 1.
25. Tapia, E.M., Intille, S.S., and Larson, K. Activity Recognition in the Home Using Simple and Ubiquitous Sensors. In *Proc. of Pervasive Computing, Second International Conference, PERVASIVE 2004*, A. Ferscha and F. Mattern, eds., Vienna, Springer-Verlag, 2004, 158.
26. Mann, W., Giraldo, C., and Helal, S. mPCA—A Mobile Patient Care-Giving Assistant for Alzheimer Patients. In *Proc. of UbiCog '02: First International Workshop on Ubiquitous Computing for Cognitive Aids*, Gothenberg, Sweden, 2002.
27. Mihailidis, A., Barbenel, J.C., and Fernie, G. The Efficacy of an Intelligent Cognitive Orthosis to Facilitate Handwashing by Persons with Moderate-to-Severe Dementia. *Neuropsychological Rehabilitation*, 14, 2004: 135.
28. McCarthy, C.E. and Pollack, M.E. A Plan-Based Personalized Cognitive Orthotic. In *Proc. of the 6th International Conference on AI Planning and Scheduling*, 2002.

29. Levinson, R. The Planning and Execution Assistant and Trainer (PEAT). *The Journal of Head Trauma Rehabilitation*, 1997.
30. Philipose, M. et al. Inferring Activities from Interactions with Objects. *IEEE Pervasive Computing: Mobile and Ubiquitous Systems*, 3(4) 2004: 50.
31. Hill, J.L. and Culler, D.E. Mica: A Wireless Platform for Deeply Embedded Networks. *IEEE Micro*, 22(6) 2002: 12.
32. Liao, L., Fox, D., and Kautz, H. Learning and Inferring Transportation Routines. In *Proc. of the 19th National Conf. on Artificial Intelligence (AAAI)*, 2004.
33. Bui, H. A General Model for Online Probabilistic Plan Recognition. In *Proc. of the Intl. Joint Conf. on Artificial Intelligence*, 2003, 136.
34. Doucet, A. et al. Rao-Blackwellised Particle Filtering for Dynamic Bayesian Networks. In *Proc. of the Conf. on Uncertainty in Artificial Intelligence*, 2000, 176.
35. Rabiner, L.R. A Tutorial on Hidden Markov Models and Selected Applications in Speech Recognition. In *Readings in Speech Recognition*, A. Waibel and K.-F. Lee, eds., Morgan Kaufmann, 1990, 267.
36. Glascock, A. and Kutzik, D. Behavioral Telemedicine: A New Approach to the Continuous Nonintrusive Monitoring of Activities of Daily Living, *Telemedicine Journal*, 6(1) 2000: 33–44.
37. Barger, T. et al. Objective Remote Assessment of Activities of Daily Living: Analysis of Meal Preparation Patterns. Technical Report, Medical Automation Research Center, University of Virginia Health System, 2002.
38. Mihailidis, A., Carmichael, B., and Boger, J. 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) 2004: 238.
39. Shi, Y., et al. Propagation Networks for Recognition of Partially Ordered Sequential Action. In *Proc. of the 2004 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, CVPR 2004*, 2004, 862.
40. Tran, Q. and Mynatt, E. What Was I Cooking? Towards Deja Vu Displays of Everyday Memory. Technical Report GIT-GVU-TR-03-33, Georgia Institute of Technology, 2003.
41. Jebara, T. and Pentland, A. Action Reaction Learning: Automatic Visual Analysis and Synthesis of Interactive Behaviour. In *ICVS '99: Proc. of the First Intl. Conf. on Computer Vision Systems*, Springer-Verlag, 1999, 273.
42. Moore, D., Essa, I., and Hayes, M. Exploiting Human Actions and Object Context for Recognition Tasks. In *Proc. of IEEE International Conference on Computer Vision 1999 (ICCV.99)*, 1999, 80.
43. Philipose, M. et al. Guide: Towards Understanding Daily Life via Auto-Identification and Statistical Analysis. In *Proc. of UbiHealth 2003: The 2nd International Workshop on Ubiquitous Computing for Pervasive Healthcare Applications*, Seattle, WA, 2003.
44. Patterson, D. et al. Fine-Grained Activity Recognition by Aggregating Abstract Object Usage. In *Proc. of the IEEE International Symposium on Wearable Computers*, 2005, 18.
45. Anderson, C., Domingos, P., and Weld, D. Relational Markov Models and Their Application to Adaptive Web Navigation. In *Proc. of the Eighth Intl. Conf. on Knowledge Discovery and Data Mining*, ACM Press, 2002, 143.