Designing Smart Environments:
A Paradigm Based on Learning and Prediction

Sajal K. Das and Diane J. Cook

Department of Computer Science and Engineering
University of Texas at Arlington

Abstract

This chapter proposes a learning and prediction based paradigm for designing smart home environments. The foundation of this paradigm lies in information theory as it manages uncertainties in inhabitants' contexts (e.g., location or mobility, and activities) in daily lives. The underlying idea is to intelligently build compressed dictionaries of context profiles collected from sensor data, efficiently learn from this information, and then predict inhabitant's future contexts. Successful prediction helps automate device control operations and tasks within the environment as well as to identify anomalies. Thus, the learning and prediction based paradigm optimizes goal functions of smart home environments such as minimizing maintenance cost, manual interactions and energy utilization. After describing some important features of smart environments, this chapter presents the architecture details of our MavHome project. The proposed paradigm is then applied to the inhabitant's location and activity tracking and prediction, and automated decision making capability. MavHome implementation issues and some practical issues are also discussed.

Keywords: Smart environments, context-aware computing, machine learning, prediction, user modeling

1. Introduction

We live in an increasingly connected and automated society. Smart environments embody this trend by linking computers and other devices to everyday settings and commonplace tasks. Although the desire to create smart environments has existed for decades, research on this multidisciplinary topic has become increasingly intense in the last ten years or so. Indeed, tremendous advances in such areas as smart (portable) devices and appliances, wireless mobile communications, pervasive computing, wireless sensor networking, machine learning and decision making, robotics, middleware and agent technologies, and human computer interfaces have made the dream of smart environments become a reality. As depicted in Figure 1, a smart environment is a small world where sensor-enabled and networked devices work continuously and collaboratively to make lives of inhabitants more comfortable. A definition of "smart" or "intelligent" is "the ability to autonomously acquire and apply knowledge", while an "environment" refers to our surroundings. We therefore define a "smart environment" as *one that is able to acquire and apply knowledge about an environment and to adapt to its inhabitants in order to improve their experience in that environment* [8].

The type of experience that individuals wish from their environment varies with the individual and the type of environment considered. For example, they may wish the environment to ensure the safety of its inhabitants, they may want to reduce the cost or overhead of maintaining the environment, they may wish to optimize the resource (e.g. utility/energy bills or communication bandwidth) usage, or they may want to automate tasks they typically perform in the environment. The expectations of such environments have evolved with the history of the field. In [8], we introduced the necessary technologies, architectures, algorithms, and protocols to build a smart environment along with a variety of existing applications. In this chapter, we will demonstrate that wireless mobile and sensor networks play a significant role in this domain.

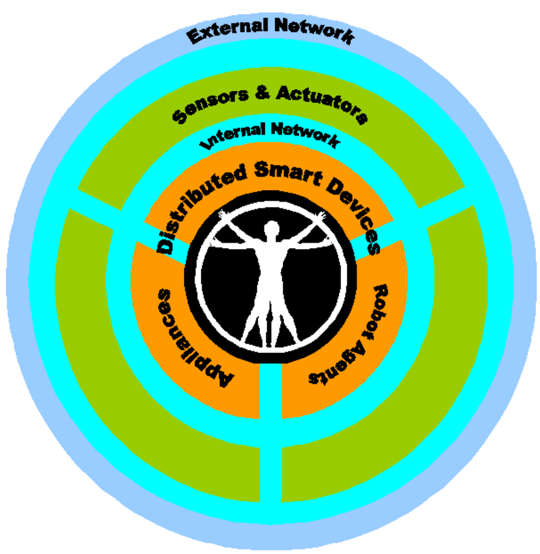

Figure 1. A Schematic View of a Smart Environment.

Reflecting the increased interest in smart environments, research labs in academia and industry are picking up the theme and creating environments with their own individual spin and market appeal. For example, the Georgia Tech Aware Home [1, 22], the Adaptive House at the University of Colorado at Boulder [26], and the MavHome smart home at the University of Texas at Arlington [10] use sensors to learn models of the inhabitants and automate activities accordingly. Other types of smart environments, including smart offices, classrooms, kindergartens, tables, and cars have been designed by MIT [4, 33], Stanford University [14], the University of California at Los Angeles [31, 32], INRIA in France [23], and Ambiente, Nissan, and Intel. Connected homes with device communications capability have become the focus of companies such as Philips, Cisco [6], GTE, Sun, Ericsson, and Microsoft [5]. Still other groups have focused on smart environments to assist individuals with health challenges. These projects include the Gloucester Smart Home [15], the Edinvar Assisted Interactive Dwelling House [13], the Intel Proactive Health project [21], agent based smart health monitoring in MavHome [11], and MALITDA smart house for individual with special needs [18]. It is easy to see that such environments are results of phenomenal advancements in wireless mobile communications infrastructures and sensor networking technologies, among others.

This chapter presents our research experience in developing smart environments through a project called MavHome [10], funded by the US National Science Foundation. In particular, we propose "learning and prediction" as an overarching framework or paradigm for designing efficient algorithms and smart protocols in such environments. The foundation of this paradigm lies in information theory as it manages inhabitants' uncertainties in mobility and activities in daily lives. The underlying idea is to build intelligent (compressed) dictionaries of mobility and activity profiles (or histories) of inhabitants, collected from sensor data, learn from this information, and then predict future mobility and action., The prediction in turn helps automate device operations and manage resources efficiently, thus optimizing the goals of the smart environment.

The chapter is organized as follows. Section 2 describes important features of smart environments. Section 3 presents the architectural details of our MavHome smart home project. Section 4 deals with the proposed paradigm for inhabitant's (indoor) location and activity prediction and automated decision making capability. Section 5 discusses MavHome implementation issues, while Section 6 highlights practical considerations. Finally, Section 7 concludes the chapter.

2. Features of Smart Environments

Important features of smart environments are that they possess a degree of autonomy, adapt themselves to changing environments, and communicate with humans in a natural way. Intelligent automation can reduce the amount of interaction required by the inhabitants, as well as reducing utility consumption and other potential wastages. These capabilities can also provide important features such as detection of unusual or anomalous behaviors for health monitoring and home security, for example.

The benefits of automation can influence every environment we interact with in daily lives. As an example, consider operations in a smart home and illustrate with the help of the following scenario. To minimize energy consumption, the home keeps the temperature cool throughout the night. At 6:45am, the home turns up the heat because it has learned that it needs 15 minutes to warm to the inhabitant's favorite waking temperature. The alarm sounds at 7:00, which signals the bedroom light to go on as well as the coffee maker in the kitchen. The inhabitant, Bob, steps into the bathroom and turns on the light. The home records this manual interaction, displays the morning news on the bathroom video screen, and turns on the shower. While Bob is shaving, the home senses that Bob is four pounds over his ideal weight and adjusts his suggested daily menu and displays in the kitchen. When Bob finishes grooming, the bathroom light turns off while the kitchen light and display turn on. During breakfast, Bob requests the janitor robot to clean the house. When Bob leaves for work, the home secures all doors behind him and starts the lawn sprinklers despite knowing the 30% predicted chance of rain. To reduce energy costs, the house turns down the heat until 15 minutes before Bob is due home. Because the refrigerator is low on milk and cheese, the home places a grocery order. When Bob arrives home, his grocery order has arrived, the house is back at Bob's desired temperature, and the hot tub is waiting for him.

This scenario highlights a number of desired features in a smart environment such as a home. In the following, let us look at some of these features in more detail [8].

Figure 2. Device Control in Smart Environments.

2.1 Remote Control of Devices

The most basic feature of smart environments is the ability to control devices remotely or automatically. Powerline control systems have been available for decades and basic controls offered by X10 can be easily installed. By plugging devices into such a controller, inhabitants of an environment can turn lights, coffee makers, and other appliances on or off in much the same way as couch potatoes switch television stations with a remote control (Figure 2). Computer software can additionally be employed to program sequences of device activities and to capture device events executed by the powerline controllers.

With this capability, inhabitants are free from the requirement of physical access to devices. Individuals with disabilities can control devices from a distance, as can the person who realized when he got to work that he left the sprinklers on. Automated lighting sequences can give the impression that an environment is occupied while the inhabitants are gone, thus handling basic routine procedures by the environment with almost no human intervention.

2.2 Device Communications

With the maturity of wireless mobile communications and middleware technology, smart environment designers and inhabitants have been able to raise their standards and expectations. In particular, devices use these technologies to communicate with each other, share data to build a more informed model of the state of the environment and/or inhabitants, and retrieve information from outside sources over the Internet or wireless communication infrastructure. This allows better response to the current state and needs.

As mentioned earlier, such “connected environments” have become the focus of many industry-developed smart homes and offices. With these capabilities, for example, the environment can access the weather page to determine the forecast and query the moisture sensor in the lawn to determine how long the sprinklers should run. Devices can access information from the Internet such as menus, operational manuals, or software upgrades, and can post information such as a grocery store list generated from monitoring inventory with an intelligent refrigerator or trash bin.

Activation of one device can also trigger other sequences, such as turning on the bedroom radio, kitchen coffee maker, and bathroom towel warmer when the alarm goes off. Inhabitants can benefit from the interaction between devices by muting the television sound when the telephone or doorbell rings; temperature as well as motion sensors can interact with other devices to ensure that the temperature is kept at a desired level wherever the inhabitants are located within the environment. Moreover, a smart environment will provide a neat service forwarding capability with the help of individual smart devices that communicate with each other without human intervention. For example, in a smart environment, calls on a mobile phone can be automatically forwarded to a nearby landline phone while emails to the mobile phone instead of outdoor cellular network.

2.3 Sensory Information Acquisition/Dissemination

The recent past has observed tremendous advancements in sensor technology and in the ability of sensors to share information and make low-level decisions. As a result, environments can provide constant adjustments based on sensor readings and can better customize behaviors to the nuances of the inhabitants' surroundings. Motion detectors or force sensors can detect the presence of individuals in the environment and accordingly adjust lights, music, or climate control. Water and gas sensors can monitor potential leaks and force the valves, thus closing them when a danger arises. Low-level control of devices offers fine-tuning in response to changing conditions, such as adjusting window blinds as the amount of daylight coming into a room changes. Networks composed of these sensors can share data and offer information to the environment at speeds and complexity not experienced in the earlier versions of smart environments. For example, the Smart Sofa [30] developed at Trinity College in Dublin, Ireland can identify an individual based on the weight and can theoretically use this information to customize the settings of devices around the house.

2.4 Enhanced Services by Intelligent Devices

Smart environments are usually equipped with numerous smart devices/appliances that provide varied and impressive capabilities. Networked together and tied to intelligent sensors and the outside world, the impact of these devices becomes even more powerful. Such devices are becoming the focus of a number of manufacturers including Electrolux, Whirlpool, and a collection of startup companies.

As examples of such devices, Frigidaire and Whirlpool offer intelligent refrigerators with features that include web cameras to monitor inventory, bar code scanners, and Internet-ready interactive screens. Through interactive cameras, inhabitants away from home can view the location of security or fire alerts and remote caregivers can check on the status of their patients or family. Merloni's Margherita 2000 washing machine is similarly Internet controlled, and uses sensor information to determine appropriate cycle times. Other devices such as microwaves, coffee makers, and toasters are quickly joining the collection.

In addition, specialized equipments have been designed in response to the growing interest in assistive environments. AT&T's Kids Communicator resembles a hamster exercise ball and is equipped with a wireless videophone and remote maneuverability to monitor the environment from any location. A large collection of companies including Friendly Robotics, Husqvarna, Technical Solutions, and University of Florida's Lawn Nibbler have developed robotic lawn mowers to ease the burden of this time-consuming task, and indoor robot vacuum cleaners including Roomba and vacuums from Electrolux, Dyson, and Hitachi are gaining in popularity and usability. Researchers at MIT's Media Lab are investigating new specialized devices, such as an oven mitt that can tell if food has been warmed all the way through. A breakthrough development from companies such as Philips is an interactive tablecloth that provides cable-free power to all chargeable objects placed on the table's surface. An environment that can combine the features of these devices with information gathering and remote control power of previous research will realize many of the intended goals of smart environment designers.

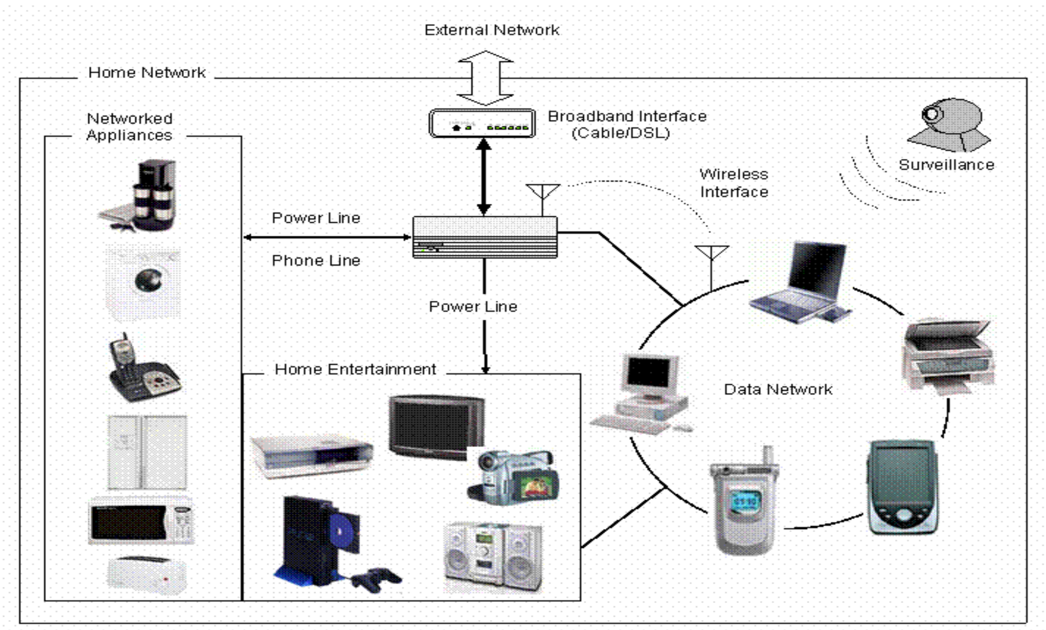

Figure 3. Networked Devices in Smart Environments.

2.5 Networking Standards

A smart environment will be able to control all of its various networked devices (see Figure 3) such as computers, sensors, cameras, and appliances, from anywhere and at anytime through the Internet. For example, when the inhabitant is away, he can still be in contact with his different environments to monitor their status and/or access his personal database. From that perspective, all the hardware and software for enabling the smart environments should be based on open standards. Moreover, they should be easy to install, configure, and operate in order to be user-friendly to the nonprofessional inhabitants or consumers. IEEE 802.11 and IEEE 802.15 based wireless LANs, and Bluetooth using spread spectrum techniques under

2.4GHz or 5GHz unlicensed ISM (Industrial, Science and Medical) wireless spectrum and Home RF (radio frequency) technology have been applied to wireless networking infrastructures for smart environments. Alongside, Ethernet (IEEE 802.3), PNA (phoneline networking alliance), and X10 powerline networking have emerged as smart-environment wired networking technologies in the market. These technologies have advantages and disadvantages. For example, X10 powerline networking has the widest availability; however, it has a much lower speed than other PNA and wireless standards. Performance comparison, co-existence capability and interoperability of these technologies have started in the academic and industry research realms while implementing prototypes of smart environments using the above standards.

2.6 Predictive Decision Making Capabilities

The features of a smart environment described up to this point provide the potential for fulfilling the goal of a smart environment; that is, improving the experience of inhabitants of the environment. However, control of these capabilities is mostly in the hands of the users. Only through explicit remote manipulation or careful programming can these devices, sensors, and controllers adjust the environment to fit the needs of the inhabitants. Full automation and adaptation rely on the software itself to learn, or acquire information that allows the software to improve its performance with experience.

Specific features of recent smart environments that meet these criteria incorporate predictive and automatic decision-making capabilities into the control paradigm. Contexts (mobility, activity, etc.) of inhabitants as well as of the environment can be predicted with good accuracy based on observed activities and known features. Models can also be built of inhabitant patterns that can be used to customize the environment for future interactions. For example, an intelligent car can collect information about the driver including typical times and routes to go to work, theatre, restaurant, and store preferences, and commonly used gas stations. Combining this information with data collected by the inhabitant's home and office as well as Internet-gathered specifics on movie times, restaurant menus and locations, and sales at various stores, the car can make recommendations based on the learned model of activity patterns and preferences.

Similarly, building a model of device performance can allow the environment to optimize its behaviors and performance. For example, a smart kitchen may learn that the coffee maker requires ten minutes to complete brewing a full pot of coffee, and will start it up ten minutes before it expects the inhabitants to want their first cup. Smart light bulbs may warn when they are about to expire, letting the factor automatically deliver replacements before the need is critical.

As a complement to predictive capabilities, a smart environment will be able to make decisions of how to automate its own behaviors to meet the specified goals. Device settings and timings of events are now in the control of the environment. Such a smart environment will also have to elect between alternate methods of achieving a goal, such as turning on lights in each room entered by an inhabitant or anticipating where the inhabitant is heading and illuminating just enough of the environment to direct the individual to their goal. In fact, this learning and prediction aspect of smart environments will be the focus of the rest of this chapter.

3 The MavHome Smart Home

The MavHome at the University of Texas at Arlington represents an environment that acts as an intelligent agent, perceiving the state of the home through sensors and acting upon the environment through device controllers. The goal is to maximize inhabitants' comfort and minimize the home's operating cost. To achieve this goal, the house must be able to reason about, learn, predict, and adapt to its inhabitants.

In MavHome, the desired smart home capabilities are organized into an agent based software architecture that seamlessly connects the components while allowing improvements to be made to any of the supporting technologies. Figure 4 describes the architecture of a MavHome agent that separates the technologies and functions into four cooperating layers. The *Decision* layer selects actions for the agent to execute. The *Information* layer collects information and generates inferences useful for making decisions. The *Communication* layer is responsible for routing and sharing information between agents. The *Physical* layer contains the environment hardware including devices, transducers, and network equipment. The MavHome software components are connected using a distributed inter-process communication interface.

Because controlling an entire house is a large-scale complex learning and reasoning problem, it is decomposed into reconfigurable tasks. Thus, the Physical layer for one agent may represent another agent somewhere in the hierarchy, which is capable of executing the task selected by the requesting agent.

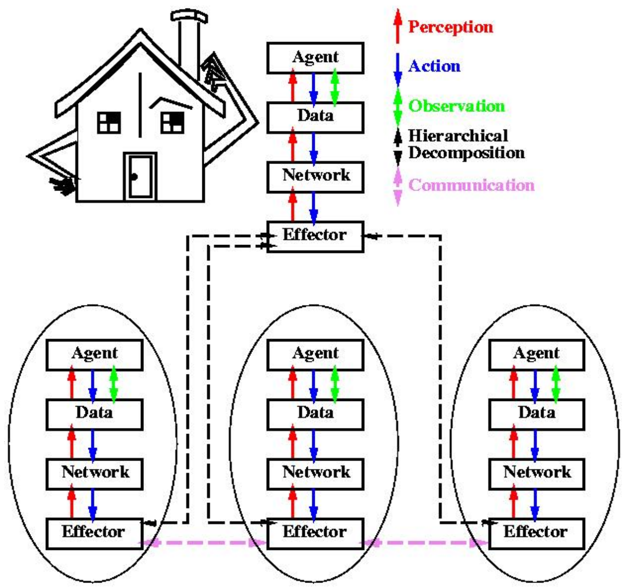

Figure 4. MavHome Agent Architecture.

Perception is a bottom-up process. Sensors monitor the environment (e.g., lawn moisture level) and, if necessary, transmit the information to another agent through the Communication layer. The database records the information in the Information layer, updates its learned concepts and predictions accordingly, and alerts the Decision layer of the presence of new data. During action execution, information flows top down. The Decision layer selects an action (e.g., run the sprinklers) and relates the decision to the Information layer. After updating the database, the Communication layer routes the action to the appropriate effector to execute. If the effector is actually another agent, the agent receives the command through its effector as perceived information and must decide upon the best method of executing the desired action. Specialized interface agents allow interaction with users, robots, and external resources such as the Internet. Agents can communicate with each other using the hierarchical flow shown in Figure 4. In the remaining discussions, a smart home will generically represent a smart environment.

4. Automation through Learning and Prediction

In order to maximize comfort, minimize cost, and adapt to the inhabitants, a smart home must rely upon sophisticated tools for intelligence building such as learning, prediction, and making automated decisions. We will demonstrate that learning and prediction indeed play an important role in determining the inhabitant's next action and anticipating mobility patterns within the home. MavHome uses these predictions in order to automate selected repetitive tasks for the inhabitant. The home will need to make this prediction based solely on past mobility patterns and previously seen inhabitant interaction with various devices (e.g., motion detectors, sensors, device controllers, video monitors), as well as the current state of the inhabitant and/or the house. The captured information can be used to build sophisticated models that aid in efficient prediction algorithms. The number of prediction errors must be minimal, and the algorithms must be able to deliver predictions with minimal delays for computation. Prediction is then handed over to a decision-making algorithm that selects actions for the house to meet its desired goals. The underlying concepts of MavHome prediction schemes lie in the text compression, online parsing and information theory. Well-investigated text compression methods [9, 35] have established that good compression algorithms are also good learner and hence good predictors. According to information theory [9], a predictor with an order (size of history used) that grows at a rate approximating the entropy rate of the source is an optimal predictor. In the following, we summarize our novel paradigm for inhabitant's mobility and activity predictions.

4.1 Inhabitant Location Prediction

By definition, a smart environment is *context-aware* in the sense that by combining inputs from multiple sensing devices, it should be able to deduce the inhabitant's intent or attributes without explicit manual input. Location is perhaps the most common example of context. Hence, it is crucial for a smart environment to track inhabitant's mobility accurately by determining and predicting his location. The prediction also helps in optimal allocation of resources and activation of effectors in location-aware applications [12, 25]. In [2], we first proposed a model-independent algorithm for location prediction in wireless cellular networks, which we later adopted for indoor location tracking and predicting inhabitant's future locations in smart homes [16, 29]. This approach is based on symbolic representation of location information that is specified not in absolute terms, but relative to the topology of the corresponding access infrastructure (e.g., sensor ids or zones through which the inhabitant passes), thus making our approach universal or technology/model independent. At a conceptual level, prediction involves some form of statistical inference, where some sample of the inhabitant's past movement history (profile) is used to provide intelligent estimates of his future location, thereby reducing the location uncertainty associated with this prediction [12, 28].

Hypothesizing that inhabitant's mobility has repetitive patterns that can be learned, and assuming the inhabitant's mobility process as stochastically random, we proved the following result [2, 3]. It is impossible to optimally track mobility with less information exchange between the system (in this case smart environment) and the device (detecting inhabitant's mobility) than the entropy rate (bits/second) of the stochastic mobility process. Specifically, given all past observations of inhabitant's position and the best possible predictors of future position, some uncertainty in the position will *always* exist unless the device and the system exchange location information. The actual *method* by which this exchange takes place is irrelevant to this bound. All that matters is that the exchange exceeds the entropy rate of the mobility process. Therefore, a key issue in establishing bounds is to characterize the mobility process (and therefore its entropy rate) in an adaptive manner. To this end, based on the information theoretic framework, an optimal on-line adaptive location management algorithm, called *LeZi-update*, was proposed [2, 3] for cellular communication networks. Rather than assuming a standard mobility model of the node, LeZi-update learns node movement history stored in a Lempel-Ziv (LZ) type of compressed dictionary [35], builds a *universal* mobility model by minimizing entropy, and predicts future locations with a high degree of accuracy. In other words, LeZi-update offers a model-independent solution to manage uncertainty related to node mobility. This framework is quite general and applicable to other contexts such as activity prediction [17], resource provisioning [12, 28], anomaly detection, and so on.

Figure 5(a) depicts a typical floor plan layout of MavHome together with the placement of motion (in-building) sensors along the inhabitant's routes, by partitioning MavHome's coverage area into sensor *zones* or sectors. When the system (environment) needs to contact the inhabitant, it will initiate a location prediction scheme. In order to control the location uncertainty, the system also relies on the location information as sampled by the sensors, which in turn helps reduce the search space for subsequent prediction. As shown in Figure 5(b), the floor plan can be represented as a connected graph $\boldsymbol{G} = (\boldsymbol{V}, \boldsymbol{E})$ where the node-set $\boldsymbol{V} = \{a, b, c, \ldots\}$ denotes the zones (sensor-ids) and the edge-set $\boldsymbol{E}$ denotes the neighborhood adjacency between a pair of zones. While moving from one zone to another, the inhabitant crosses an array of sensors along a route. For example, the movement from corridor (R) to the dining room (D) in the floor plan can be expressed by the collection of sensors $\{j, l\}$ or $\{j, k\}$.

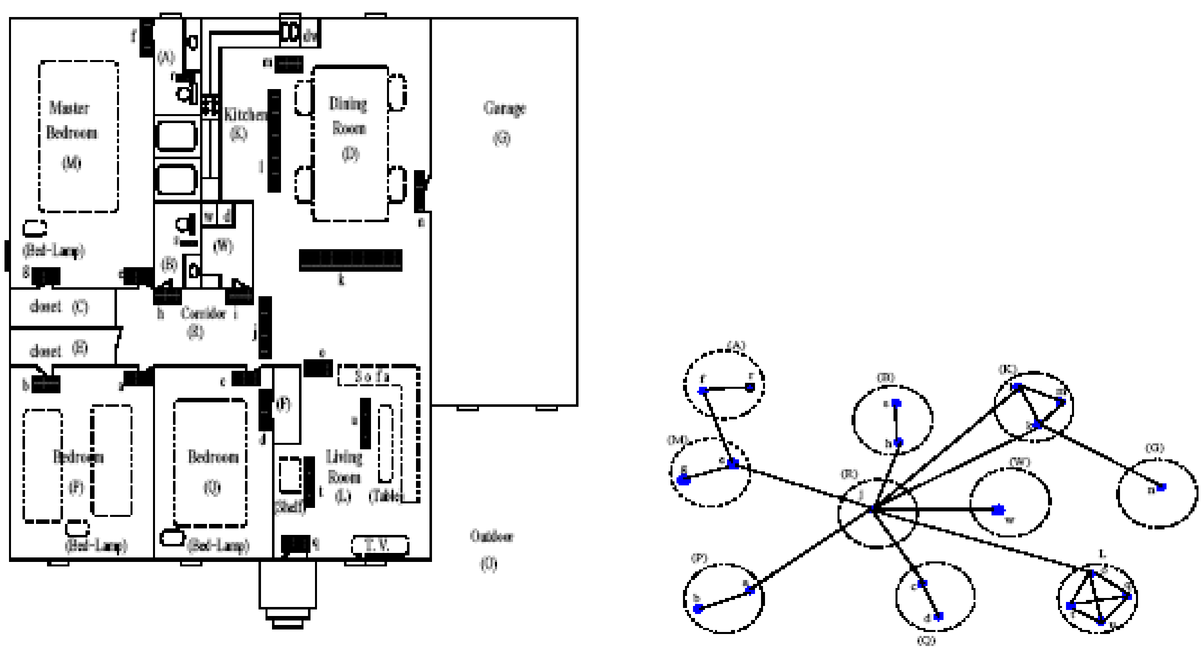

Figure 5. (a) Typical floor plan of MavHome architecture, (b) Graph representing connectivity of sensor zones

The LeZi-update framework uses a *symbolic space* to represent sensing zone of the smart environment as an alphabetic symbol and thus captures inhabitant's movement history as a string of symbols. That is, while the geographic location data are often useful in obtaining precise location coordinates, the symbolic information removes the burden of frequent coordinate translation and is capable of achieving universality across different networks [25, 28]. (The blessing of symbolic representation also helps us hierarchically abstract the indoor connectivity infrastructure into different levels of granularity.) Tacit in this formulation is that every node has some movement patterns that can be learned in an on-line fashion. Essentially, we assume that node itineraries are inherently *compressible* and this allows application of *universal data compression* algorithms [35], which make very basic and broad assumptions, and yet minimize the source entropy for stationary Ergodic stochastic processes [27].

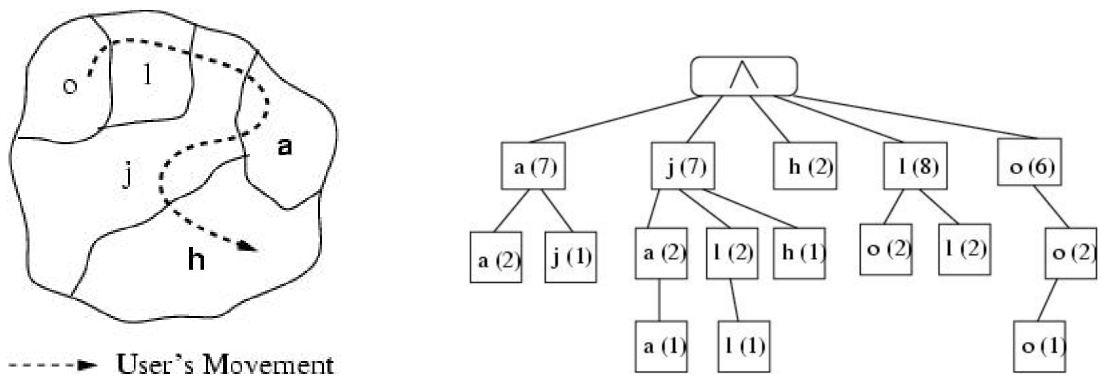

Figure 6. (a) Symbolic representation of mobility, (b) Trie holding zones and their frequencies

In LeZi-update, the symbols (sensor-ids) are processed in chunks and the entire sequence of symbols withheld until the last update is reported in a compressed (encoded) form. For example, referring to the abstract representation of mobility route in Figure 6(a), let *ajlloojhhaajlloojaajlloojaajll…* be the inhabitant's movement history at any instant. This string of symbols can be parsed as distinct substrings (or phrases) "*a, j, l, lo, o, jh, h, aa, jl, loo, ja, aj, ll, oo, jaa, jll, …*". As shown in Figure 6(b), such a symbol-wise context model, based on variable to fixed-length coding, can be efficiently stored in a dictionary implemented by a *trie*. Essentially the mobile acts as an *encoder* while the system acts as a *decoder* and the frequency of every symbol is incremented for *every prefix of every suffix* of each phrase. By accumulating larger and larger contexts, one can affect a paradigm shift from traditional position update to *route update*. For stationary Ergodic sources with n symbols, this framework achieves asymptotic optimality, with improved update cost bounded by $\Omega(\lg n - \lg \lg n)$ where lg n denotes logarithm base 2.

Table 1. Phrases and their frequencies at context "jl", "j" and Λ

Jl	*J*	Λ		
l|*jl*(1)	*a*|*j*(1)	*a*(4)	*aa*(2)	*aj*(1)
Λ|*jl*(1)	*aa*|*j*(1)	*j*(2)	*ja*(1)	*jaa*(1)
	l|*j*(1)	*jl*(1)	*jh*(1)	*l*(4)
	ll|*j*(1)	*lo*(1)	*loo*(1)	*ll*(2)
	h|*j*(1)	*o*(4)	*oo*(2)	*h*(2)
	Λ|*j*(2)	Λ(1)		

One major objective of LeZi-*update* scheme is to endow the prediction process, by which the system finds nodes whose position is uncertain, with sufficient information regarding the node mobility profile. Each node in the trie preserves the relevant frequencies provided by the update mechanism in the current context. Thus, considering "*jll*" as the latest update message, the usable contexts are its prefixes, namely: "*jl*", "*j*" and Λ (null symbol). A list of all predictable routes (parsed phrases) with frequencies in this context is shown in Table 1. Following the blending technique of *prediction by partial match* (PPM) [7], the probability computation starts from the leaf nodes (highest level) of the trie and *escapes* to the lower levels until the root is reached. Based on the principle of *insufficient reasoning* [27], every phrase probability is distributed among individual symbols (zones) according to their relative occurrence in a particular phrase. The total residence probability of every zone (symbol) is computed by adding all the probabilities it has accumulated from all possible phrases at this context. The optimal prediction order is now determined by polling the zones in decreasing order of these residence probabilities.

So overall, the application of information theoretic methods to location prediction allowed quantification of minimum information exchanges to maintain accurate location information, provided an on-line method by which to characterize mobility, and in addition, endowed an optimal prediction sequence [12]. Through learning this approach allows us to build a higher order mobility model rather than assuming a finite model, and thus minimizes entropy and leads to optimal performance.

While the basic LeZi-Update algorithm was used to predict only the current location from past movement patterns, this approach has also been extended in [29] to predict the likely future routes (or trajectories) of inhabitants in smart homes and also for heterogeneous environments [24]. The route prediction exploits the Asymptotic Equipartition Property in information theory [9] which states that for a random process X with entropy $H(X)$, the number of observed unique paths of length n is $2^{H(X)}$ with probability 1. In other words, for reasonably large n, most of the probability mass is concentrated in only a small subset (called the *typical set*) of routes, which encompasses the inhabitant's most likely routes and captures the *average nature of long-length sequences*. Accordingly, the algorithm simply predicts a relatively small set of likely paths (one of which the user will almost surely take next). A smart home environment can then act on this information by activating resources (for example, by turning on the lights on corridors that constitute one or more of these routes) in a minimal and efficient manner rather than turning on all lights in the house. Experiments demonstrate that our predictive framework can save up to 70% (electrical) energy in a typical smart home environment [29]. The accuracy of prediction is up to 86% and only 11% of routes constitute the typical set.

4.2 Inhabitant Action Prediction

A smart home inhabitant typically interacts with various devices as part of his routine activities. These interactions may be considered as a sequence of events, with some inherent pattern of recurrence. Again, this repeatability leads us to the conclusion that the sequence can be modeled as a stationary stochastic process as for mobility. Inhabitant action prediction consists of first mining the data to identify sequences of actions that are regular and repeatable enough to generate predictions, and then using a sequence matching approach to predict the next action in one of these sequences.

To mine the data, a window can be moved in a single pass through the history of inhabitant actions, looking for sequences within the window that merit attention. Each sequence is evaluated using the Minimum Description Length principle [27], which favors sequences that minimize the description length of the sequence once it is compressed by replacing each instance of the discovered pattern with a pointer to the pattern definition. A regularity factor (daily, weekly, monthly) helps compress the data and thus increases the value of a pattern. Action sequences are first filtered by the mined sequences. If a sequence is considered significant by the mining algorithm, then predictions can be made for events within the sequence window. Using this algorithm as a filter for two alternative prediction algorithms, the resulting accuracy increases on average by 50%. This filter ensures that MavHome will not erroneously seek to automate anomalous and highly variable activities [19, 20].

As above, the action prediction algorithm parses the input string (history of interactions) into substrings representing phrases. Because of the prefix property used by the algorithm, parsed substrings can be efficiently maintained in a trie along with frequency information. To perform prediction, the algorithm calculates the probability of each symbol (action) occurring in the parsed sequence, and predicts the action with the highest probability. To achieve optimal predictability, the predictor must use a mixture of all possible order models (phrase sizes) when determining the probability estimate. To accomplish this, techniques from the PPM family of predictors are incorporated, that generate weighted Markov models of different orders. This blending strategy assigns greater weight to higher-order models, in keeping with the advisability of making the most informed decision.

In experiments run on sample smart home data, predictive accuracy of this approach converged on 100% for perfectly-repeatable data with no variation, and converged on 86% accuracy for data containing variations and anomalies [17].

4.3 Automated Decision Making

As mentioned earlier, the goal of MavHome is to enable the home to automate basic functions in order to maximize the inhabitants' comfort and minimize the operating cost of the home. We assume comfort is a function of the number of manual interactions with the home, and the operating cost of energy usage.

Because the goal is a combination of these two factors, blind automation of all inhabitant actions is frequently not the desired solution. For example, an inhabitant might turn on the hallway light in the morning before opening the blinds in the living room. MavHome could, on the other hand, open the blinds in the living room before the inhabitant leaves the bedroom, thus alleviating the need for the hallway lights. Similarly, turning down the air conditioning after leaving the house and turning it back up before returning would be more energy efficient than turning the air conditioning to maximum after arriving home in order to cool it as quickly as possible [29].

To achieve its goal, MavHome uses reinforcement learning to acquire an optimal decision policy. In this framework, the agent learns autonomously from potentially-delayed rewards rather than from a teacher, reducing the requirement for the home's inhabitant to supervise or program the system. To learn a strategy, the agent explores the effects of its actions over time and uses this experience to form control policies that optimize the expected future reward.

MavHome learns a policy based on a state space, $S = \{s_i\}$, consisting of the states of the devices in the home, the predictions of the next event, and expected energy utilization over the next time unit. A reward function, r, takes into account the amount of required user interaction, the energy consumption of the house, and other parameters that quantify the performance of the home. This reward function can be tuned to the particular preferences of the inhabitants, thus providing a simple means to customize the home's performance. Q-learning is used [34] to approximate an optimal action strategy by estimating the predicted value, $Q(s_t, a_t)$, of executing action a_t in state s_t at time t. After each action, the utility is updated as: $Q(s_t, a_t) \leftarrow \alpha[r_{t+1} + \gamma \max_{a \in A} Q(s_{t+1}, a) - Q(s_t, a_t)]$. After learning, the optimal action, a_t, can be determined as: $a_t = \arg\max_{a \in A} Q(s_t, a)$.

5. MavHome Implementation

In the MavHome smart home project at the University of Texas at Arlington, student activity data are collected continuously based on their interactions with devices in the environment. Off-the-shelf X10 controllers automate most devices and thus inhabitant's actions. Arrays of sensors track their mobility.

Using the ResiSim 3D simulator, a graphical model has been constructed of the intelligent environment. The model allows a visitor at a remote location to monitor or change the status of devices in MavHome, as shown in Figures 7 and 8. Images in the left column of Figure 7 show web cameras placed throughout the environment, and the simulator visualization is shown on the right. The "Information" window in the lower right indicates that devices have recently been manipulated, either manually or by

MavHome. Figure 8 shows that the light in the entryway (upper left) is illuminated once Darin enters the environment and the lamp on Ryan's desk (lower left) turns on to assist him with work. The updated status of the lamp is shown by the yellow circle in the ResiSim model (right). The model will indicate the status of sensors as well – the orbs in Figure 9 indicate that there are two areas of activity captured by motion sensors.

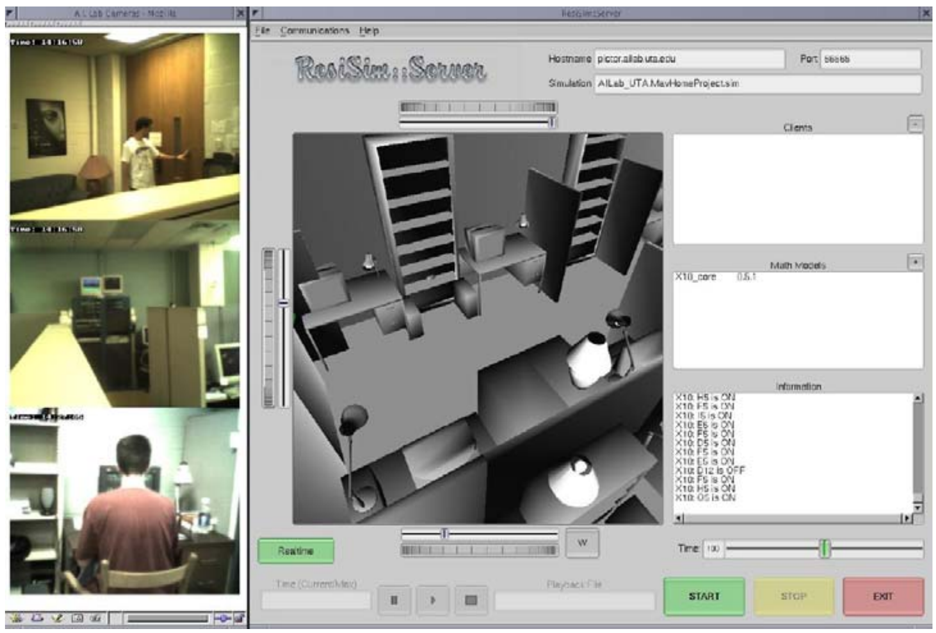

Figure 7. Web camera views of MavHome environment (left) and ResiSim visualization (right).

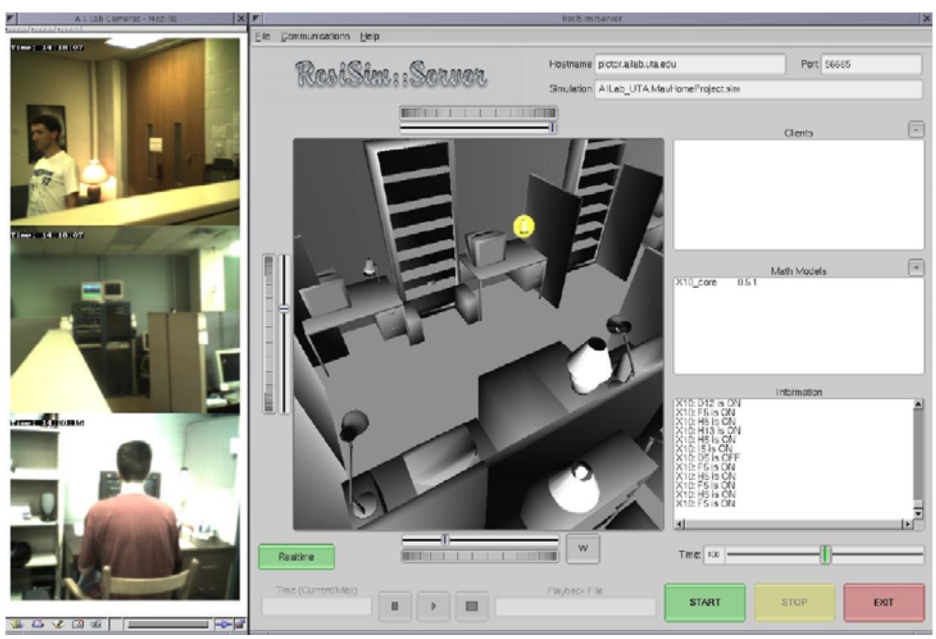

Figure 8. ResiSim update after desk lamp (lower left) is turned on.

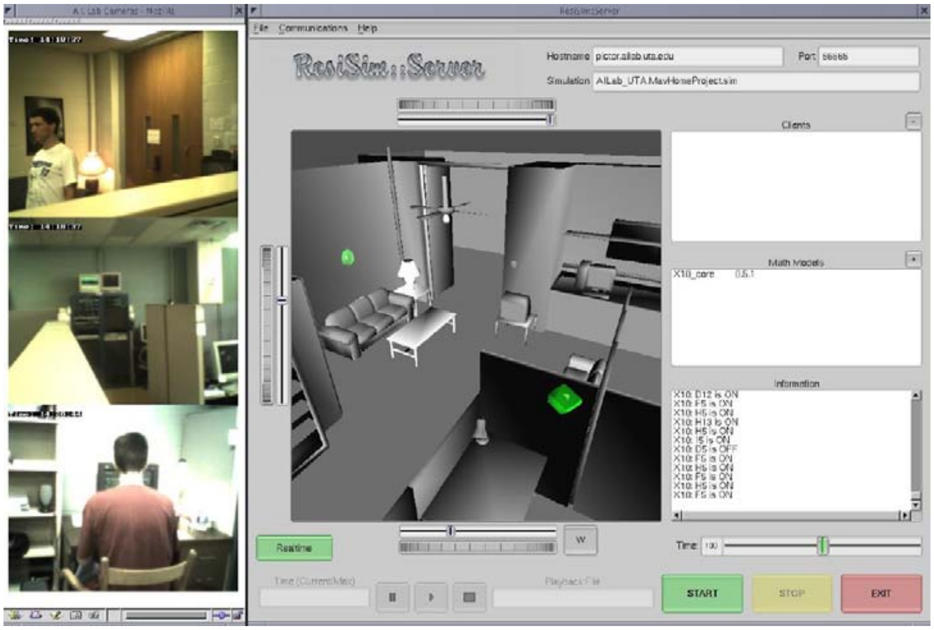

Figure 9. ResiSim indicates activated motion sensors with green orbs.

A live demonstration of MavHome was conducted in the fall of 2004. During the previous weeks, activity data was collected for one of the project participants ("MavHome Bob"). Actions included turning on lights en route to his desk in the morning, watching a live news feed on the computer, taking a coffee and TV break, and turning off devices on the way out at the end of the day. Despite the presence of approximately fifty people during the live demonstration (who were setting off motion sensors throughout the environment), MavHome correctly predicted and automated each activity. Figure 10 reflects the movements of MavHome Bob as he moves through the environment and lights are illuminated reflecting his typical activities.

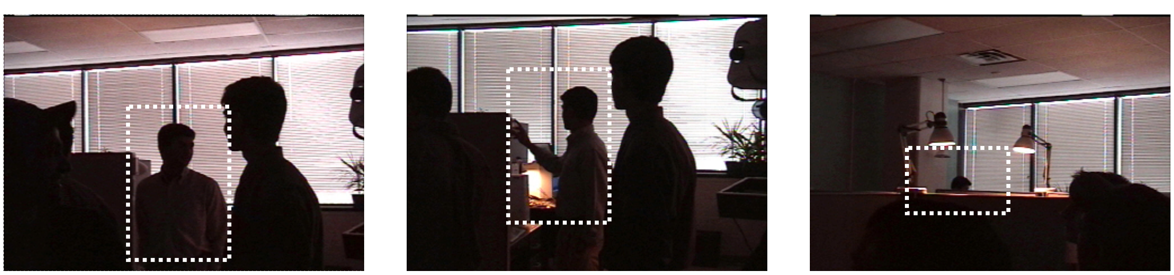

Figure 10. Bob's movements in MavHome. Bob's position is indicated by a dashed box.

6. Practical Considerations

So how easily can the features of a smart home be integrated into new or existing homes? The software described in the MavHome implementation consists of commercial X10 controllers, a computer, a variety of sensors, and a wireless network. A simple implementation can be integrated into an existing house for under a thousand dollars, in many cases. If robots or customized devices are introduced, the cost increases.

A computer interface to a smart home must be very simple. Manual control of devices can override home decisions, and alternative interfaces including voice control are offered. Other than starting or resetting the software, no interaction with the computer is required. In our experiments, the software adapted to user activities in a couple of weeks, but the training time will vary according to the complexity of inhabitant's actions and the number of people in the home. Although minimal expertise is required, various types of interaction are possible depending on the needs of the inhabitant. The user can vary certainly the threshold at which activities are automated, although this is not necessary because manual resetting of actions selected by the house constitute negative reward and will eventually cause the house to not automate those particular commands. The inhabitant can also request that the home simply make suggestions for automation; selection of rules for automation will be made by the inhabitant on a case-by-case basis.

Introducing intelligent control into a house can result in a number of privacy and safety issues. Safety constraints must be placed on each device to ensure that the house will not select an action that endangers inhabitants. The house may not be allowed, for example, to select a temperature setting below 50 degrees or above 90 degrees. The entire automation can be quickly disabled with one mouse click or voice command – each device can operate with or without computer control. The inhabitant also needs to specify the type of data that can be collected, and which data, if any, can be disseminated for learning across multiple households or cities.

Similarly, smart homes typically benefit from collecting information about the health, typical patterns, and other features of their inhabitants. This leads to a number of privacy and security issues. Data should only be collected on features allowed by the inhabitants, and shared with other sites only as volunteered.

New smart homes in neighboring locations could, for example, benefit from patterns learned in an older home, but care must be taken to share information without violating the privacy of home inhabitants.

7. Conclusions

This chapter demonstrated the effectiveness of learning and prediction based paradigm in a smart home environment. Efficient prediction algorithms provide information useful for future locations and activities, automating activities, optimizing design and control methods for devices and tasks within the environment, and identifying anomalies. These technologies reduce the work to maintain a home, lessen energy utilization, and provide special benefits for elderly and people with disabilities. In the future, these abilities will be generalized to conglomeration of environments, including smart offices, smart roads, smart hospitals, smart automobiles, and smart airports, through which a user may pass through in daily life. Another research challenge is how to characterize mobility and activity profiles of multiple inhabitants (e.g., living in the same home) in the same dictionary and predict or trigger events to meet the common goals of the house under conflicting requirements of individual inhabitants.

Acknowledgements: This research was supported by the US National Science Foundation grants under award numbers IIS-0121297 and IIS-0326505.

References

[1] G. D. Abowd, "Classroom 2000: An Experiment with the Instrumentation of a Living educational Environment," *IBM Systems Journal*, special issue on human-computer interaction: a focus on pervasive computing, 38(4), pp, 508-530, 1999.

[2] A. Bhattacharya and S. K. Das, "LeZi-Update : An Information-Theoretic Approach to Track Mobile Users in PCS Networks," *Proc. ACM International Conference on Mobile Computing and Networking (MobiCom)*, pp. 1-12, Aug 1999.

[3] A. Bhattacharya and S. K. Das, "LeZi-Update: An Information-Theoretic Approach for Personal Mobility Tracking in PCS Networks," *Wireless Networks*, 8(2-3), pp. 121-135, Mar-May 2002.

[4] A. Bobick, S. Intille, J. Davis, F. Baird, C. Pinhanez, L. Campbell, Y. Ivanov, A. Schutte, and A. Wilson, "The KidsRoom: A Perceptually-Based Interactive and Immersive Story Environment," *Presence*, 8(4), pp. 369-393, Aug 1999.

[5] B. Brumitt, J. Kumm, B. Meyers, and S. Shafer, "Ubiquitous Computing and the Role of Geometry," *IEEE Personal Communications*, 7(5), pp. 41-43, Aug 2000.

[6] Cisco, http://www.cisco.com/warp/public//3/uk/ihome

[7] J. G. Cleary and I. H. Witten, "Data Compression Using Adaptive Coding and Partial String Matching," *IEEE Transactions on Communications*, vol. 32, no. 4, pp. 396-402, April 1984.

[8] D. J. Cook and S. K. Das, *Smart Environments: Technology, Protocols, and Applications*, Chapter 1, John Wiley, 2005.

[9] T. M. Cover and J. A. Thomas, *Elements of Information Theory*, Wiley, 1991.

[10] S. K. Das, D. J. Cook, A. Bhattacharya, E. O. Heierman, and T.-Y. Lin, "The Role of Prediction Algorithms in the MavHome Smart Home Architecture," *IEEE Wireless Communications*, 9(6), pp. 77-84, Dec 2002.

[11] S. K. Das and D. J. Cook, "Agent Based Health Monitoring in Smart Homes," *Proceedings of the International Conference on Smart Homes and Health Telematics (ICOST)*, Singapore, Sept 2004 (Keynote Talk).

[12] S. K. Das and C. Rose, "Coping with Uncertainty in Wireless Mobile Networks," *Proceedings of IEEE Personal, Indoor and Mobile Radio Communications*, Barcelona, Spain, Sept 2004. (Invited Paper)

[13] Edinvar, http://www.stakes.fi/tidecong/732bonne.html

[14] A. Fox, B. Johanson, P. Hanrahan, T. Winograd, "Integrating Information Appliances into an Interactive Space," *IEEE Computer Graphics and Applications*, 20(3), pp. 54-65, 2000.

[15] Gloucester, http://www.dementua-voice.org.uk/Projects_GloucesterProject.html

[16] K. Gopalratnam and D. J. Cook, "Online Sequential Prediction via Incremental Parsing: The Active LeZi Algorithm", *to appear in IEEE Intelligent Systems*, 2005.

[17] K. Gopalratnam and D. J. Cook, "Active LeZi: An Incremental Parsing Algorithm for Sequential Prediction", *International Journal of Artificial Intelligence Tools*, 14(1-2), 2004.

[18] S. Helal, B. Winkler, C. Lee, Y. Kaddoura, L. Ran, C. Giralo, S. Kuchibholta and W. Mann, "Enabling Location-Aware Pervasive Computing Applications for the Elderly," *Proceedings of IEEE International Conference on Pervasive Computing and Communications* (PerCom'03), pp. 531-538, Mar 2003.

[19] E. Heierman, M. Youngblood, and D. J. Cook, "Mining Temporal Sequences to Discover Interesting Patterns", *KDD Workshop on Mining Temporal and Sequential Data,* 2004.

[20] E. Heierman and D. J. Cook, "Improving Home Automation by Discovering Regularly Occurring Device Usage Patterns", *Proceedings of the International Conference on Data Mining*, 2003.

[21] Intel, http://www.intel.com/research/prohealth

[22] C. Kidd, R. J. Orr, G. D. Abowd, D. Atkeson, I. Essa, B. MacIntyre, E. D. Mynatt, T. E. Starner, and W. Newstetters, "The Aware Home: A Living Laboratory for Ubiquitous Computing," *Proceedings of the Second International Workshop on Cooperative Buildings*, 1999.

[23] C. Le Gal, J. Martin, A. Lux and J. L. Crowley, "Smart Office: Design of an Intelligent Environment," *IEEE Intelligent Systems*, 16(4), July-Aug 2001.

[24] A. Misra, A. Roy and S. K. Das, "An Information Theoretic Framework for Optimal Location Tracking in Multi-System 4G Wireless Networks," *IEEE INFOCOM*, Mar 2004.

[25] A. Misra and S. K. Das, "Location Estimation (determination and Prediction) Techniques in Smart Environments," In: *Smart Environments* (Eds. D. J. Cook and S. K. Das), Chapter 8, pp. 193-228, John Wiley, 2005.

[26] M. Mozer, "The Neural Network House: An Environment that Adapts to its Inhabitants," *Proceedings of the AAAI Spring Symposium on Intelligent Environments*, 1998.

[27] J. Rissanen, *Stochastic Complexity in Statistical Inquiry*, World Scientific, 1989.

[28] A. Roy, S. K. Das and A. Misra, "Exploiting Information Theory for Adaptive Mobility and Resource Management in Future Wireless Cellular Networks," *IEEE Wireless Communications*, 11(4), pp. 59-64, Aug 2004.

[29] A. Roy, S. K. Das Bhaumik, A. Bhattacharya, K. Basu, D. J. Cook and S. K. Das, "Location Aware Resource Management in Smart Homes," *Proceedings of IEEE International Conference on Pervasive Computing and Communications* (PerCom'03), pp. 481-488, Mar 2003.

[30] Smart sofa, http://www.dsg.cs.tcd.ie/?category_id=350

[31] M. B. Srivastava, R. Muntz nd M. Potkonjak, "Smart Kindergarten: Sensor-Based Wireless Networks for Smart Developmental Problem-Solving Environments," *Proceedings of 7th ACM International Conference on Mobile Computing and Networking* (MobiCom'01), 2001.

[32] P. Steurer and M. B. Srivastava, "System Design of Smart Table," *Proceedings of the 1st IEEE International Conference on Pervasive Computing and Communications* (PerCom'03), Mar 2003.

[33] M. C. Torrance, "Advances in Human-Computer Interaction: The Intelligent Room," *Working Notes of the CHI 95 Research Symposium*, 1995.

[34] C. J. Watkins, *Learning from Delayed Rewards*, PhD Thesis, Cambridge Uivesity, 1989.

[35] J. Ziv and A. Lempel, "Compression of Individual Sequences via Variable Rate Coding," *IEEE Transcations on Information Theory*, 24(5), pp. 530-536, Sept 1978.