Published in final edited form as:
J Ambient Intell Smart Environ. 2010 ; 2(2): 145–152. doi:10.3233/AIS-2010-0061.

Predicting Air Quality in Smart Environments

Seun Deleawe[a], **Jim Kusznir**[b], **Brian Lamb**[b], and **Diane J. Cook**[b,*]

[a] Department of Computer Science and Engineering, University of North Texas, Denton, TX, USA

[b] School of Electrical Engineering and Computer Science, Washington State University, Pullman, WA, USA

Abstract

The pervasive sensing technologies found in smart environments offer unprecedented opportunities for monitoring and assisting the individuals who live and work in these spaces. As aspect of daily life that is often overlooked in maintaining a healthy lifestyle is the air quality of the environment. In this paper we investigate the use of machine learning technologies to predict CO_2 levels as an indicator of air quality in smart environments. We introduce techniques for collecting and analyzing sensor information in smart environments and analyze the correlation between resident activities and air quality levels. The effectiveness of our techniques is evaluated using three physical smart environment testbeds.

Keywords

data mining; machine learning; smart environments; activity recognition; air quality

1. Introduction

With the aging of the world's population, researchers have started to focus on creating technologies that can assist with monitoring and ensuring the health and safety of older adults living alone [1]. Much of the emphasis has been placed on detecting falls and ensuring that residents are performing daily activities [10]. On the other hand, there are other factors of an environment which can dramatically impact health. One of these is the air quality of the environment. The World Health Organization [31] reports that 2.4 million individuals die annually from causes directly attributable to air pollution, 1.5 million of these from indoor air pollution. Worldwide there are more deaths from poor air quality than from automobile accidents.

While researchers do study the effect of air quality on human health, outdoor air quality has historically received a significantly greater amount of attention than indoor air quality [14]. Nevertheless, a recent study [25] showed that US residents, on average, spend 88% of their day inside buildings, 7% in a vehicle, and only 5% outside. Because individuals continue to spend a majority of their lives indoors, indoor air quality continues to have a significant effect on health. Indoor air quality is often described by the presence or absence of various pollutants. These pollutants include but are not limited to combustion products, volatile organic compounds, and biological particles. Carbon dioxide (CO_2), a colorless, odorless gas formed in the body during metabolic processes can also serve as an indicator of indoor air quality [32]. Typical indoor CO_2 concentrations range between 720 and 2000 ppm but can exceed 3000 ppm [14]. Moderate levels of CO_2 can cause feelings of stuffiness and

[*] Corresponding author. cook@eecs.wsu.edu.

discomfort, respiration can be slightly affected by levels greater than 15,000 ppm and exposure to levels over 30,000 can lead to headaches, dizziness, and nausea.

Because indoor air quality is strongly related to health, society would benefit from automated methods of predicting the indoor air quality that is anticipated for a variety of conditions. *We hypothesize that machine learning techniques can predict indoor air quality given sensor data that is collected in a smart environment.* The ability to predict indoor air quality in dynamic situations would be extremely beneficial. The information can be used to evaluate alternative methods for providing clean air. In addition, injecting fresh air can now be based on detection of resident state and activities and not just on static factors such as maximum room capacity, as is traditionally employed.

To validate our hypothesis, we collect sensor data in several physical smart environment testbeds. We calculate total motion and use machine learning techniques to automatically recognize activities from the raw sensor data. We also collect CO_2 levels in these settings. We use this collected data to evaluate the ability of machine learning techniques to predict air quality from smart environment data.

2. Smart Environments

A recent convergence of technologies in machine learning and pervasive computing has caused interest in the development of *smart environments* to emerge. In addition to providing an interesting platform for developing adaptive and functional software applications, smart environments can also be employed for valuable functions such as at-home health monitoring and automation assistance. The long-term goal of our CASAS smart environment project [23] is to perform automated health monitoring and to provide automated assistance that will allow individuals to remain independent in their own homes. Given the aging of the population, the cost of formal health care, and the importance that individuals place on remaining independent in their own homes [1],[11], these technologies will become an increasingly important component of our everyday lives.

The emphasis of smart home assistance for individuals with special needs has been to monitor completion of ADL (Activities of Daily Living) activities [3],[18],[21]. We are taking the research to the next step by determining if automatically-recognized activities and automatically-calculated motion levels can be mapped onto predict indoor air quality levels We treat a smart environment as an intelligent agent that perceives the state of the resident and the physical surroundings using sensors and acts on the environment using controllers in such a way that the specified performance measured is optimized [6]. Researchers have generated ideas for smart environment software algorithms that track the location of single residents, that generate reminders, and that react to hazardous situations [34]. Some projects with physical testbeds have begun to emerge including the MavHome [35], the Gator Tech Smart House [12], the iDorm [9], and the Georgia Tech Aware Home [2]. Resulting from these advances, researchers are now beginning to recognize the importance of applying smart environment technology to health assistance [3],[15],[16],[19],[22] and companies are recognizing the potential of this technology for a quickly-growing consumer base [13].

The role of smart environments in this research is to provide non-obtrusive monitoring that not only detects motion and recognizes activities but also uses this information to provide real-time predictions and estimates of air quality. In order to predict air quality in smart environments, we collect sensor data in our smart environment testbeds. We calculate total motion that occurs within a time window in the space and also identify the current activity. We next use this information to predict air quality. All of our data and experimental validation is performed in the context of our three physical smart environment testbeds: two smart apartments and a smart workplace.

3. Testbeds

Our smart environment testbeds are located on the Washington State University campus and are maintained as part of our ongoing CASAS smart home project [23]. We performed our testing in three separate physical smart environment testbeds: two apartments and one workplace environment. The physical layout and sensor placement for these three environments are shown in Figs. 1 through 3. As shown in Fig. 1, the first smart apartment testbed (which we label “Kyoto”) contains three bedrooms, one bathroom, a kitchen, a living room, and a dining area. The apartment is equipped with motion sensors distributed approximately 1 meter apart throughout the space. In addition, we have also installed sensors to provide ambient temperature readings and custom-built analog sensors to provide readings for hot water, cold water, and stove burner use. Contact switches allow us to monitor usage of key items including a cooking pot, a medicine container, and the phone book. In addition, Insteon™ power controls monitor usage and control lighting throughout the space. Sensor data is captured using an in-house sensor network and is stored in a SQL database. Our middleware uses a XMPP-based publish-subscribe protocol as a lightweight platform and language-independent method to push data to client tools (i.e., our data analysis and application programs).

Our second smart apartment (which we label “Tulum”) is shown in Fig. 2. This is a smaller environment in which we equipped only the downstairs areas with motion sensors, positioned approximately 4 feet apart. Finally, we have equipped an on-campus smart workplace environment (which we label “Tokyo”), shown in Fig. 3. This is a laboratory that is organized into four cubicles with desks and computers, an open server area, a postdoc office, a meeting area, a lounge, and a kitchen. Like the apartment, the lab is equipped with motion sensors placed approximately 1 meter apart throughout the space and magnetic sensors record door openings and closing. In addition, powerline controllers operate all of the lights in the room. Each sensor event is represented by the event’s date, time, sensor ID, and sensor value.

The sensors that we use in these environments allow our algorithms to recognize and track daily activities. While this feature alone has benefits for monitoring the functional and physical well-begin of residents, we hypothesize that the data can be used additionally to detect types of social interactions. We do not employ cameras or microphones in these testbeds. While they may offer valuable insights for social interaction detection, they are typically not well-accepted by the community that we want to serve with this technology [8] and therefore are not used as part of our smart environment testbeds.

Using four Fluke 975 AirMeters (shown in Fig. 4) logging one data point each minute, we collected CO_2 at four different locations. The first three are our smart environments: Kyoto, Tulum, and Tokyo. The fourth location is an outdoor spot directly outside the Tokyo smart workplace environment (which we label “TokyoOut”). Each air quality data reading consists of a reading number, temperature, wet bulb, dew point, percent relative humidity, carbon monoxide level, carbon dioxide level, and time stamp. For the purpose of this study, we focused on the temperature, the carbon dioxide level, and the time stamp. We collected a total of 10,080 readings in these environments over a one week time span. During this time each apartment (Kyoto and Tulum) housed two volunteer participants who performed their normal daily routines. Approximately ten undergraduate and graduate students performed their research activities in the smart workplace (Tokyo) during this time.

Fig. 5 shows the CO_2 readings that were measured at each of the sites during our data collection process. As the graph shows, all of the air quality levels are fairly good and are well within the recommended ranges for indoor air quality. The outdoor readings are better overall than the indoor readings, which may be due to the increased amount of human activity in the somewhat confined indoor spaces. We can see that there are fluctuations in the readings throughout the day. However, the fluctuations do not cover a wide range of values (with the possible exception of readings in the Kyoto environment), which make this a more difficult concept to learn than in environments where there is a tremendous amount of fluctuation in indoor air quality levels.

4. Processing Smart Environment Data

The goal of this study is to determine if smart environment sensor data can be used to predict air quality readings. In order to achieve this goal we collect smart environment sensor readings for a window of time leading up to the corresponding CO_2 data reading. There are a number of decisions that need to be made to generate the most accurate model possible, and we address these decisions here.

4.1. Determining the time window size

The first decision to make is how much smart environment data to collect that corresponds to one air quality reading. Since we are collecting readings for a fixed window of time before the CO_2 reading, this translates to a question of what the time window size should be. A big window size will generate a large amount of data to train the model, which can improve the accuracy of the model. On the other hand, the most current data most dramatically impacts the air quality level and collecting data that is old may actually degrade the accuracy of the model. In order to analyze the effect of different window sizes on different environment layouts and activities we test a variety of window sizes. Our results report the accuracy of models generated with windows of size 2, 5, 10, 15, 20, and 30 minutes.

4.2. Generating motion level features

One of the most valuable pieces of information a smart environment provides for air quality is the *activity level* or *motion level* in the environment. This can be measured by calculating the number of motion sensors that are activated during the data collection time window. Our motion sensors are activated by movement in a 4′ sq radius and stay on until no new movement is detected for 5 seconds. In our first approach the motion count is a total of all motion sensors that activated during the time window.

In our second approach, we calculate a weighted mobility count. We do this because the activity closest to the Fluke meter will most greatly affect the readings at the meter itself. Each motion sensor in this approach is assigned a value between 1 and 8 based on its proximity to the meter (8 being the closest to the meter). We then compute a sum of these weighted values to generate input data for our model.

4.3. Automatically-recognized activities

While collecting sequences of sensor readings in a smart environment is valuable, determining what activities these sequences represent provides even more valuable insights on the residents' functional health. This information can in turn be used to assess the residents' well-being and to provide context-aware, customized interventions and services for the residents. On the other hand, recognizing activities from raw sensor data is challenging. Researchers have conducted studies that assess the ability of machine learning technologies to recognize activities using wearable sensors [18], by monitoring interactions with objects in the environment [20],[21], by videotaping activities [4], and by analyzing motion sensor data [7]. A variety of models including naïve Bayes classifiers [4],17,[28], decision trees [18], and probabilistic model such as Markov models, dynamic Bayes networks, and conditional random fields [7],[17],[21],[26] have been tested.

While these studies have indicated the power of algorithmic methods for activity recognition, they have been tested in single-resident settings where the activities are uninterrupted. In contrast, we have designed an approach that handles cases where some of the activities are interrupted [27], some activities occur in parallel, and some activities involve multiple residents [28]. In order to recognize these activities in a smart environment, we use a portion of the data as sample data for creating a model of the activities. Specifically, we use a hidden Markov model (HMM) to model the dynamic system. A hidden Markov model (HMM) is a statistical model in which the underlying model is a stochastic process that is *not* observable (i.e. hidden) and is assumed to be a Markov process which can be observed through another set of stochastic processes that produce the sequence of observed symbols. A HMM assigns probability values over a potentially infinite number of sequences. Because the probabilities values must sum to one, the distribution described by the HMM is constrained. This means that the increase in probability values of one sequence is directly related to the decrease in probability values for another sequence.

In the case of a Markov chain, all states are observable states and are directly visible to the observer. Thus, the only other parameter in addition to the prior probabilities of the states and the distribution of feature values for each state is the state transition probabilities. In the case of a hidden Markov model, there are hidden states which are not directly visible, and the observable states (or the variables) influence the hidden states. Each state is associated with a probability distribution over the possible output tokens. Transitions from any one state to another are governed by transition probabilities as in the Markov chain. Thus, in a particular state an outcome can be generated according to the associated probability distribution.

HMMs are known to perform very well in cases where temporal patterns need to be recognized which aligns with our requirement in recognizing interleaved activities. The conditional probability distribution of any hidden state depends only on the value of the preceding hidden state. The value of an observable state depends only on the value of the current hidden state. The observable variable at time t, namely x_t, depends only on the hidden variable y_t at that time. We can specify an HMM using three probability distributions: the distribution over initial states $\Pi = \{\pi_k\}$, the state transition probability distribution $A = \{a_{kl}\}$, with $a_{kl} = p(y_t{=}l|y_{t-1}{=}k)$ representing the probability of transitioning from state k to state l; and the observation distribution $B = \{b_{il}\}$, with $b_{il} = p(x_t{=}i|y_t{=}l)$ indicating the probability that the state l would generation observation $x_t{=}i$. These distributions are estimated based on the relative frequencies of visited states and state transitions observed in the training data.

Given a set of training data our algorithm uses the sensor values as parameters of the hidden Markov model. Given an input sequence of sensor event observations $x_1..x_t$, our goal is to find the most likely sequence of hidden states or activities, $y_1..y_t$, which could have generated the observed event sequence. We use the Viterbi algorithm [30] to identify this sequence of hidden states following the calculation in Equation 1.

$$\arg\max_{x_1..x_t} P(y_1, .., y_t, y_{t+1}|x_{1:t+1}) \tag{1}$$

In our implementation of a hidden Markov model, we treat every activity as a hidden state. Next, every sensor is treated as an observable state in the model due to the fact that every sensor which is used is observable in our dataset. Based on the collected data we estimate the prior probability (i.e., the start probability) of every state which represents the belief about which hidden state the HMM is in when the first sensor event is seen. The training data is also used to estimate the transition probability between any two states *a* and *b*, (as the ratio of occurrences of a transition from *a* to *b* to the total number of transitions out of state a) and the emission probability that represents the likelihood of observing a particular sensor event for a given activity. We have tested these approaches on data in our smart apartment and have generated results above 90% for most activities.

For this study, we add activity information for one of the environments, the Kyoto smart apartment. We capture and annotate 16 possible activities for this environment and abstract them into 5 categories for this task, which include:

1. Sleeping
2. Grooming/cleaning
3. Eating
4. Shower
5. Studying
6. Other

4.4. Additional features

In addition to motion counts and activities, we collect additional features that may have an effect on air quality. These include time of day and temperature readings. Each of these features was discretized using equal-size binning. Time of day was discretized into morning, afternoon, evening, and night values, while temperature readings were discretized into high, mid, and low readings.

5. Machine Learning Model

We employed three different machine learning methods to model the air quality data. The goal of the model is to learn a mapping from smart environment features to an air quality range. The first technique we evaluate is a naïve Bayes classifier which selects the air quality label which fits the feature descriptions with the greatest probability. A naïve Bayes classifier uses the relative frequencies of feature values and the class values found in training data to learn a mapping from a data point description to a classification label. Naïve Bayes classifiers have been shown to yield promising results for activity recognition, which is one reason we consider this model for our study. For our second model we use a multilayer perceptron, which traditionally handles continuous-valued attributes (such as temperature and motion sensor counts in our case) well. Our network employs one hidden layer with five hidden nodes, a learning rate of 0.3, and a momentum of 0.2. Finally, we learn the mapping from features to classification with a decision tree model which uses entropy to select an ordering of feature values to consider in the concept rule description. Because a decision tree generates decision rules as its model we can understand the attributes that were most influential in predicting the air quality class. The decision tree we employ has a confidence factor of 0.25. For all three approaches we use the Weka [33] implementation of the learning algorithms.

6. Results

In our experiments we want to determine if smart environment information can be used to learn a model of air quality. We are also interested to see how alternative feature choices and alternative environments affect the predictive accuracy of the models. In our first experiment we compare data collection window sizes, number of class labels, and learning algorithms for the Tulum smart apartment. Each of the results is generated using 3-fold cross validation. The results are listed as percentages of the test data-set that are correctly classified with the true air quality level. For each row in the table we indicate with an asterisk (*) each case in which the accuracy of the leading classifier is significantly higher ($p < 0.05$) based on a paired t-test calculation. As the results in Table 1 indicate, the decision tree algorithm consistently outperformed the other learning methods on this problem. While the predictive accuracy is lower when a greater number of classes are being learned, both models were learned with a fairly high degree of success. The larger window sizes tend to perform best. We notice that since the Fluke meter averages past readings together, it does take a while for the CO_2 readings to drop back down to normal after a flurry of CO_2-generating activity is done. This may explain why larger time windows are effective at predicting air quality for these studies.

In our first testbed, the decision tree significantly ($p < 0.05$) outperforms the other methods in each case. While this is usually the case in the other test-beds as well (summarized in Tables 2 through 4), we should point out that all of the models perform consistently well, which indicates that air quality can in fact be learned from smart environment information. This is the main goal of our effort and the evidence validates in these studies.

In the second experiment we analyze data collected in the Tokyo smart workplace environment. As before, we are interested in seeing if the machine learning algorithm can successfully predict air quality levels from smart environment data. As in the first experiment we compare three different learning algorithms for different time window sizes and number of class labels. In this experiment, we also include an attribute that represents outdoor air quality levels. This information is provided by the TokyoOut location immediately outside the Tokyo lab.

Table 2 summarizes the results from the second experiment. Once again the decision tree algorithm performed the best and the larger time windows yielded better predictive results. The smart environment data combined with the learning algorithms did successfully predict air quality levels in this second smart environment testbed. An interesting observation is that the inclusion of outdoor air quality levels did improve the performance of the weaker models (the naïve Bayes classifier and the neural network) but did not significantly improve the performance of the strongest model. While outdoor air quality may have an effect on nearby indoor environments, this will in general be dependent upon the number of windows, the quality of filters, and the activities in the indoor environment. More testing on a number of different types of environments will provide greater insight on the effect of outdoor air quality on the indoor values for particular environments.

Our last experiments focused on the Kyoto smart apartment testbed. Here we once again test the learning algorithms on varying window sizes and number of class labels. In addition, in this study we compare the result of using an unweighted count of motion sensor events with a weighted count. We also consider the effect of discretize motion counts into ranges instead of using raw values. These results are summarized in Table 3. The decision tree algorithm did perform best again. The numbers did not improve when we used weighted motion sensor counts. This was contrary to our expectations. These are fairly small spaces and the results indicate that activity that occurs even away from the sensor can affect the overall indoor air quality readings to a measurable degree. As a result, all activity that occurs within a smart environment should be included in any model that is used to predict air quality in the environment. In our final experiment we include activity information as an attribute that is fed in to the machine learning algorithms. This represents the output of one machine learning algorithm that is provided as input to another algorithm in order to improve the performance of the overall system. We tested our theory that activity information would provide useful insights on air quality prediction in the context of the Kyoto smart apartment testbed. The results of this experiment are summarized in Table 4.

The inclusion of activity information does improve the accuracy of each of the models. The average accuracy of the models with no activity information is 82.74% for two class labels and 73.88% for three labels. In contrast, the average accuracy of the models including activity information is 86.69% for two class labels and 79.33% for three class labels. The addition of the activity information thus improves the prediction accuracy by 4.70%, on average. This result indicates that not only smart environment sensor data is useful for predicting air quality, but smart environment algorithms that intelligent process sensor data to recognize resident activities are valuable for predicting the quality of air in an indoor smart environment.

7. Conclusions

The goal of this work is to determine if smart environment sensor data can be used to predict air quality levels. The results we obtained from our study indicate that CO_2 levels can be learned with a reasonable amount of accuracy and therefore machine learning models build from sensor data can be used as a partial indicator of dynamic air quality conditions in smart environments. In an ideal scenario CO_2 levels would be used with other partial indicators of air quality such as volatile organic compounds for a more holistic quantification of the quality of air in the environment.

This study allows us to investigate the ability of machine learning algorithms to predict air quality levels based upon smart environment sensor information. We note that this study used specific implementations and customizations of three machine learning models. In order to perform a thorough analysis of the most effect model for this problem additional models and parameterizations should be considered and analyzed.

There are additional influencing factors that could be analyzed in order to provide a more comprehensive model. For example, all of the data in this study was collected during summer months. The nature of air quality influences will likely be different in the colder winter months. In addition, features of the residents themselves should be taken into consideration, such as monitoring environments where one or more residents are smokers. Future work can sample data under a greater number of varying conditions to determine the ability of these models to predict and generalize over such variations. In our current approach we sample several different time windows to collect input data for the learned model. In actuality the resident’s time spent within a room will affect the optimal window size for the analysis. We plan to estimate resident times from available CO_2 and smart environment sensor data and use this information to automate window size selection.

When the air quality is predicted it can then be used to automate tasks such as automated ventilation and purification of air. The intelligent regulation of such tasks in a smart environment will eventually prove to be economical as well as efficient. Future improvements upon the results we obtained may require more accurate estimates of the proximity of the sensors to the meters, alternative feature selection methods, and more accurate measurement of CO_2 levels.

Acknowledgments

This work is supported in part by National Science Foundation grant IIS-0647705.

References

1. AARP. These four walls… Americans 45+ talk about home and community. 2003.
2. Abowd, G.; Mynatt, E. Designing for the human experience in smart environments. In: Cook, D.; Das, S., editors. Smart Environments: Technology, Protocols, and Applications. Wiley; 2004. p. 153-174.
3. Barger T, Brown D, Alwan M. Health status monitoring through analysis of behavioral patterns. EEE Transactions on Systems, Man, and Cybernetics. 2005; Part A(35):22–27.
4. Brdiczka, O.; Reignier, P.; Crowley, J. Detecting individual activities from video in a smart home. Proceedings of the International Conference on Knowledge-Based and Intelligent Information and Engineering Systems; 2007. p. 363-370.
5. Centers for Disease Control and Prevention. Health aging for older adults. www.cdc.gov/Aging
6. Cook, D.; Das, D., editors. Smart Environments: Technology, Protocols, and Applications. Wiley; 2004.
7. Cook D, Schmitter-Edgecombe M. Assessing the quality of activities in a smart environment. Methods of Information in Medicine. 2009; 48(5):480–485. [PubMed: 19448886]
8. Demiris G, Oliver DP, Dickey G, Skubic M, Rantz M. Findings from a participatory evaluation of a smart home application for elder adults. Technology and Health Care. 2008; 16:111–118. [PubMed: 18487857]
9. Doctor F, Hagras H, Callaghan V. A fuzzy embedded agent-based approach for realizing ambient intelligence in intelligent inhabited environments. IEEE Transactions on Systems, Man, and Cybernetics, Part A. 2005; 35:55–56.
10. Griswold W, Shanahan P, Brown S, Boyer R, Ratto M, Shapiro R, Truong T. ActiveCampus – Experiments in community-oriented ubiquitous computing. IEEE Computer. 2006; 37:73–81.
11. Gross, J. A grass-roots effort to grow old at home. The New York Times; August 14. 2007
12. Helal A, Mann W, El-Zabadani H, King J, Kaddoura Y, Jansen E. The Gator Tech smart house: A programmable pervasive space. IEEE Computer. 2005; 38:50–60.
13. Intel Proactive Health. www.intel.com/research/prohealth/cs-aging_in_place.htm
14. Jones AP. Indoor air quality and health. Atmospheric Environment. 1999; 33:4535–4564.
15. Kautz, H.; Arnstein, L.; Borriello, G.; Etzioni, O.; Fox, D. An overview of the assisted cognition project. Proceedings of the AAAI Workshop on Automation as Caregiver: The Role of Intelligent Technology in Elder Care; 2002. p. 60-65.
16. Larson, C. In elder care, signing on becomes a way to drop by. The New York Times; February 4. 2007
17. Liao, L.; Fox, D.; Kautz, H. Location-based activity recognition using relational Markov networks. Proceedings of the International Joint Conference on Artificial Intelligence; 2005. p. 773-778.
18. Maurer, U.; Smailagic, A.; Siewiorek, D.; Deisher, M. Activity recognition and monitoring using multiple sensors on different body positions. Proceedings of the International Workshop on Wearable and Implantable Body Sensor Networks; 2006. p. 113-116.
19. Mihailidis A, Barbenl J, Fernie G. The efficacy of an intelligent cognitive orthosis to facilitate handwashing by persons with moderate-to-severe dementia. Neuropsychological Rehabilitation. 2004; 14:135–171.
20. Munguia-Tapia, E.; Intille, SS.; Larson, K. Proceedings of PERVASIVE. 2004. Activity recognition in the home using simple and ubiquitous sensors; p. 158-186.
21. Philipose M, Fishkin K, Perkowitz M, Patterson D, Fox D, Kautz H, Hahnel D. Inferring activities from interactions with objects. IEEE Pervasive Computing. 2004; 3:50–57.
22. Pollack M. Intelligent technology for an aging population: The use of AI to assist elders with cognitive impairment. AI Magazine. 2005; 26:9–24.
23. Rashidi P, Cook D. Keeping the resident in the loop: Adapting the smart home to the user. IEEE Transactions on Systems, Man, and Cybernetics, Part A: Systems and Humans. 2009; 39(5):949–959.

24. Rialle V, Ollivet C, Guigui C, Herve C. What do family caregivers of Alzheimer's disease patients desire in smart home technologies? Methods of Information in Medicine. 2008; 47:63–69. [PubMed: 18213430]
25. Robinson, J.; Nelson, WC. National human activity pattern survey database. United States Environmental Protection Agency; Research Triangle Park, NC: 1995.
26. Sanchez D, Tentori M, Favela J. Activity recognition for the smart hospital. IEEE Intelligent Systems. 2008; 23:50–57.
27. Singla G, Cook D, Schmitter-Edgecombe M. Tracking activities in complex settings using smart environment technologies. International Journal of BioSciences, Psychiatry and Technology. 2009; 1(1):25–35.
28. Singla G, Cook D, Schmitter-Edgecombe M. Recognizing independent and joint activities among multiple residents in smart environments. Ambient Intelligence and Humanized Computing Journal. 2010; 1(1):57–63.
29. Van Kasteren, T.; Kröse, B. Bayesian activity recognition in residence for elders. Proceedings of the International Conference on Intelligent Environments; 2008. p. 1-8.
30. Viterbi A. Error bounds for convolutional codes and an asymptotically optimum decoding algorithm. IEEE Transactions on Information Theory. 1967; 13:260–269.
31. WHO Member State. Estimated deaths and DALYs attributable to selected environmental risk factors. [Accessed 27 Aug 2009]. www.who.int/quantifying_ehimpacts/publications/preventingdisease6.pdf
32. Wolkoff P, Gunnar DN. Organic compounds in indoor air-their relevance for perceived indoor air quality. Atmospheric Environment. 2001; 3:4407–4417.
33. Witten, IH.; Frank, E. Data mining: Practical machine learning tools and techniques. Elsevier; 2005.
34. Wren, C.; Munguia-Tapia, E. Toward scalable activity recognition for sensor networks. Proceedings of the Workshop on Location and Context-Awareness; 2006.
35. Youngblood GM, Cook D. Data mining for hierarchical model creation. IEEE Transactions on Systems, Man, and Cybernetics, Part C. 2007; 37:1–12.

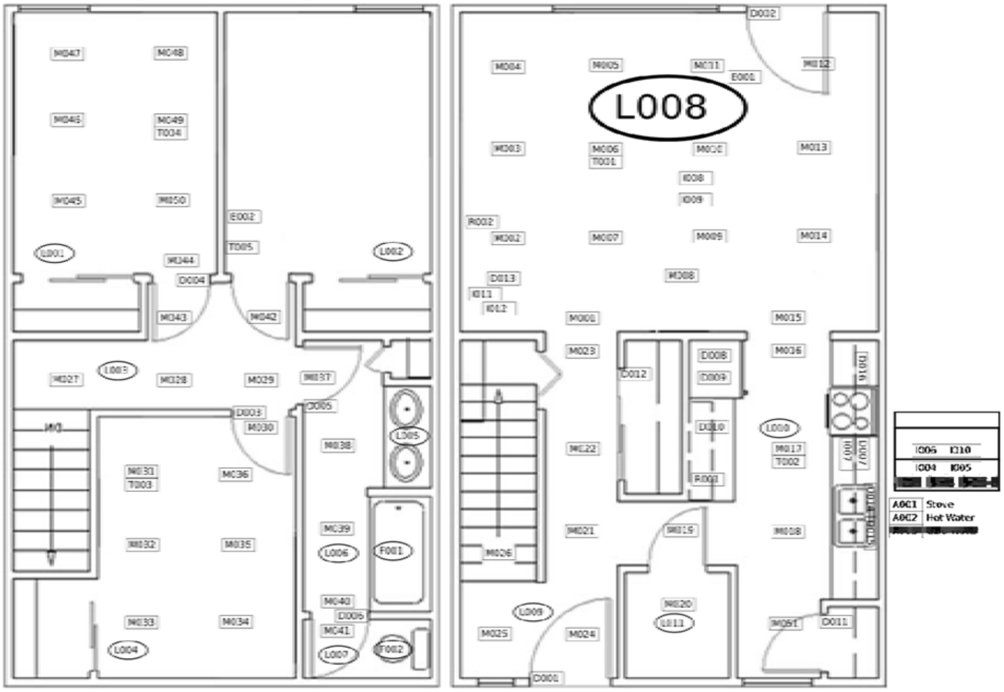

Fig. 1.

WSU Kyoto smart apartment testbed. Sensors in the apartment monitor motion, temperature, water, telephone, and item use.

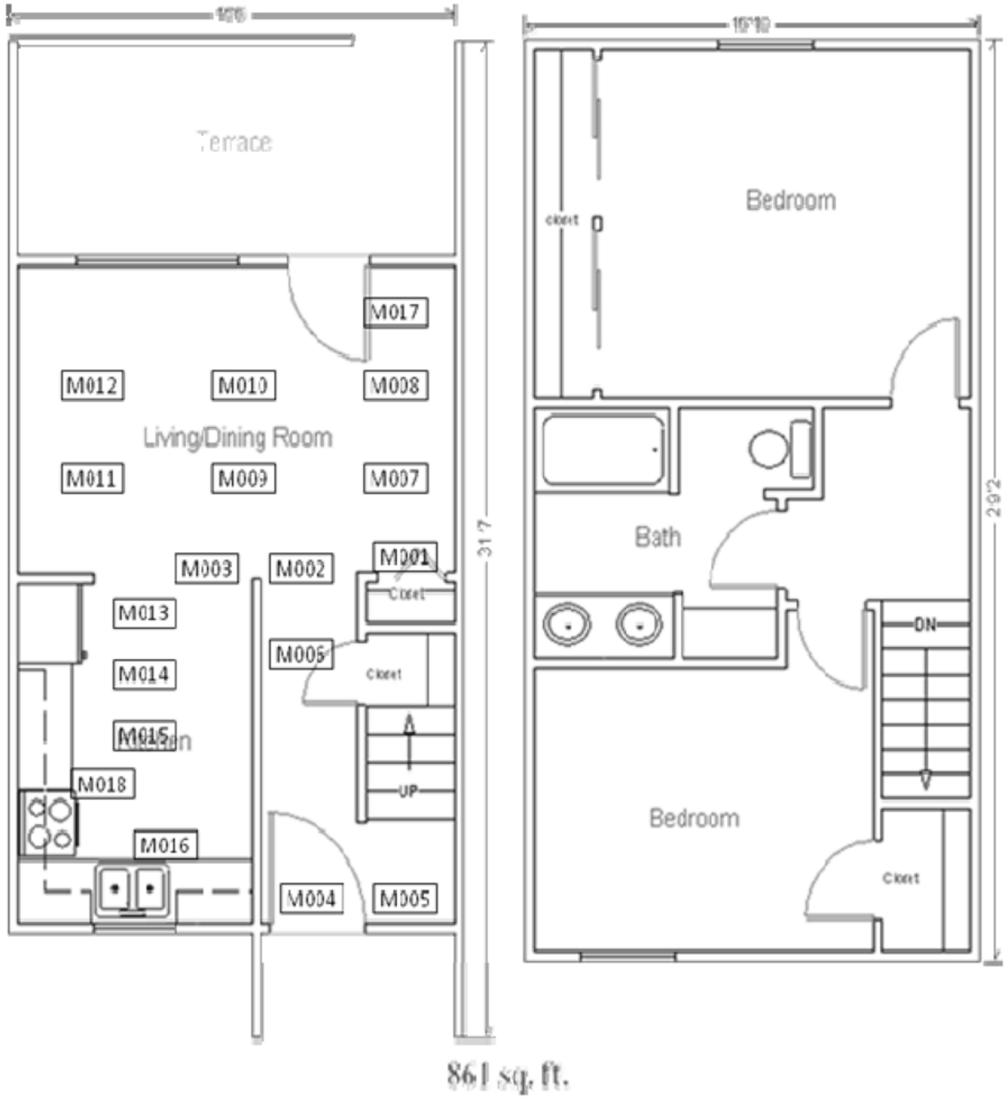

Fig. 2.
WSU Tulum smart apartment testbed and motion sensor layout.

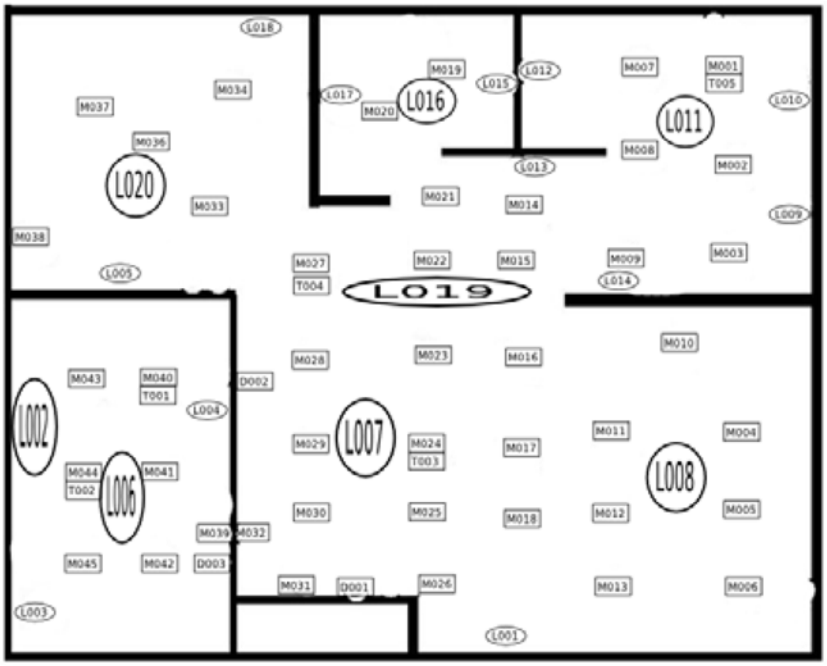

Fig. 3.
WSU Tokyo smart workplace testbed.

Fig. 4.
Fluke air quality meter.

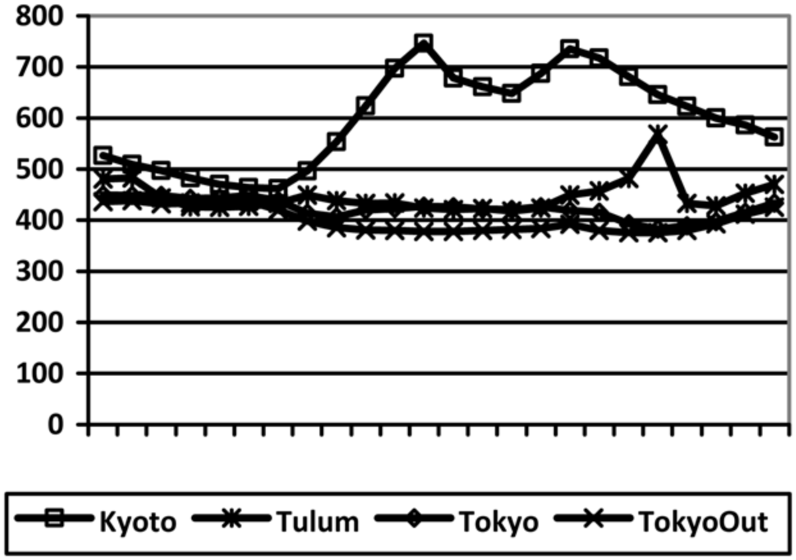

Fig. 5.

Air quality readings averaged by hour for the four monitored locations.

Table 1

Results in the Tulum smart apartment testbed for varying machine learning models, time window sizes, and numbers of classes. The data was collected and analyzed while two participants lived in the apartment and performed their normal daily routines. Statistical significance of comparative performance between the best classification accuracy and other classifiers is indicated with an asterisk (*).

Number of Air Quality Class Levels	Window size	Naïve Bayes	Neural Network	Decision Tree
Two levels (low, high)	2	65.09%*	72.95%*	86.54%
	5	66.15%*	74.33%*	88.52%
	10	66.17%*	76.48%*	88.83%
	15	66.38%*	76.43%*	89.22%
	20	67.01%*	74.54%*	**89.46%**
	30	66.13%*	75.36%*	89.23%
	Average	66.16%	75.02%	89.06%
Three levels (low, medium, high)	2	54.15%*	61.93%*	85.38%
	5	54.03%*	62.95%*	85.24%
	10	56.21%*	64.26%*	85.29%
	15	57.65%*	63.44%*	84.99%
	20	58.30%*	63.16%*	85.36%
	30	58.49%*	63.88%*	**85.61%**
	Average	56.47%	63.27%	85.31%

Table 2

Results in Tokyo for varying machine learning models, time window sizes, numbers of classes, and use of outdoor air quality values in the learning model. Statistical significance of comparative performance is indicated with an asterisk (*).

Number of Air Quality Class Levels	Use Outside Air Quality Readings	Window Size	Naïve Bayes	Neural Network	Decision Tree
Two levels	no	2	67.39%*	77.26%*	91.19%
		5	67.33%*	75.83%*	91.26%
		10	64.20%*	78.39%*	**91.31%**
		15	65.27%*	77.91%*	91.28%
		20	67.61%*	77.93%*	91.24%
		30	69.68%*	78.25%*	90.90%
	yes	2	75.88%*	84.35%*	90.95%
		5	52.53%*	84.43%*	90.58%
		10	78.02%*	84.78%*	91.11%
		15	78.22%*	85.34%*	90.99%
		20	82.28%	85.54%	90.88%
		30	78.48%*	85.11%*	90.75%
Three levels	no	2	50.24%*	62.83%*	85.02%
		5	67.31%*	72.81%*	84.81%
		10	52.03%*	67.94%*	85.16%
		15	52.53%*	68.23%*	**85.95%**
		20	52.37%*	68.44%*	84.99%
		30	52.65%*	68.81%*	84.99%
	yes	2	66.51%*	71.00%*	85.15%
		5	67.31%*	72.81%*	84.81%
		10	67.79%*	73.26%*	84.55%
		15	68.68%*	73.07%*	84.34%
		20	68.70%*	74.03%*	84.35%
		30	68.64%*	72.32%*	84.83%

Table 3

Results in Kyoto for varying machine learning models, time window sizes, numbers of classes, and weighted vs. unweighted motion counts. Statistical significance of comparative performance is indicated with an asterisk (*).

Number of Air Quality Class Levels	Weighted/Unweighted Motion	Window Size	Naive Bayes	Neural Network	Decision Tree
Two levels	Unweighted	2	69.12%*	81.05%*	96.54%
		5	69.43%*	79.40%*	**96.66%**
		10	69.89%*	81.12%*	96.36%
		15	71.33%*	81.01%*	96.47%
		20	69.48%*	80.57%*	96.54%
		30	71.31%*	86.59%*	96.49%
	Weighted	2	69.12%*	81.61%*	96.55%
		5	68.87%*	80.06%*	96.74%
		10	70.21%*	80.17%*	96.49%
		15	70.94%*	85.67%*	96.46%
		20	70.74%*	80.94%*	96.45%
		30	71.42%*	80.66%*	96.29%
Three levels	Unweighted	2	58.81%*	73.78%*	95.75%
		5	58.59%*	70.52%*	**95.86%**
		10	57.72%*	70.80%*	95.74%
		15	58.59%*	70.52%*	95.86%
		20	57.09%*	65.78%*	95.76%
		30	57.09%*	69.53%*	95.76%
	Weighted	2	58.61%*	72.80%*	95.70%
		5	57.34%*	74.55%*	95.57%
		10	57.85%*	71.24%*	95.75%
		15	58.21%*	70.36%*	95.80%
		20	58.12%*	70.63%*	95.78%
		30	58.30%*	71.43%*	95.68%

Table 4

Air quality results in the Kyoto smart apartment testbed. The results are reported for varying machine learning models, different time window sizes, and different numbers of class labels. Statistical significance of comparative performance is indicated with an asterisk (*).

Number of Air Quality Class Levels	Window size	Naïve Bayes	Neural Network	Decision Tree
Two levels (low, high)	2	77.55%*	86.11%*	96.57%
	5	76.56%*	86.42%*	96.61%
	10	76.23%*	87.97%*	96.60%
	15	76.15%*	88.27%*	96.33%
	20	76.55%*	88.05%*	96.47%
	30	77.02%*	84.50%*	96.40%
Three levels (low, medium, high)	2	61.88%*	78.93%*	95.65%
	5	62.02%*	80.50%*	95.93%
	10	62.36%*	78.43%*	95.68%
	15	64.17%*	79.36%*	95.89%
	20	64.45%*	80.01%*	95.99%
	30	63.61%*	77.38%*	95.80%