

Assessing the Quality of Activities in a Smart Environment

D.J. Cook and M. Schmitter-Edgecombe
Washington State University

Corresponding Author:

Diane J. Cook
School of Electrical Engineering and Computer Science
Box 642752
Washington State University
Pullman, WA 99163
cook@eecs.wsu.edu
Office: +1 509-335-4985
Fax: +1 509-335-3818

Summary

Objectives: Pervasive computing technology can provide valuable health monitoring and assistance technology to help individuals live independent lives in their own homes. As a critical part of this technology, our objective is to design software algorithms that recognize and assess the consistency of Activities of Daily Living that individuals perform in their own homes.

Methods: We have designed algorithms that automatically learn Markov models for each class of activity. These models are used to recognize activities that are performed in a smart home and to identify errors and inconsistencies in the performed activity.

Results: We validate our approach using data collected from 60 volunteers who performed a series of activities in our smart apartment testbed. The results indicate that the algorithms correctly label the activities and successfully assess the completeness and consistency of the performed task.

Conclusions: Our results indicate that activity recognition and assessment can be automated using machine learning algorithms and smart home technology. These algorithms will be useful for automating remote health monitoring and interventions.

Keywords: activities of daily living, smart homes, activity recognition, health monitoring, machine learning

1. Introduction

A convergence of technologies in machine learning and pervasive computing has caused interest in the development of *smart environments* to emerge and assist with valuable functions such as remote health monitoring and intervention. The need for development of such technologies is underscored by the aging of the population, the cost of formal health care, and the importance that individuals place on remaining independent in their own homes. When surveyed about assistive technologies, family caregivers of Alzheimer's patients ranked activity identification, functional assessment, medication monitoring and tracking at the top of their list of needs [1].

To function independently at home, individuals need to be able to complete both basic (e.g., eating, dressing) and more complex (e.g., food preparation, medication management, telephone use) Activities of Daily Living (ADLs) [2]. We have designed an algorithm that labels the activity that an inhabitant is performing in a smart environment based on the sensor data that is collected by the environment during the activity. In the current study, our goal is to introduce a method of assessing the quality of the performed activity by determining its level of consistency with normal execution of the activity. To test our approach, we have participants simulate difficulty with ADL completion but provide little direction about the types of errors to be committed. We then evaluate whether our algorithm will not only recognize activities, but will also assess the quality of the activity and will be robust enough to identify a larger variety of errors that can occur with ADL completion in a real world environment.

There is a growing interest in designing smart environments that reason about residents [3,4], provide health assistance [5], and perform activity recognition [6,7,8]. However, several challenges need to be addressed before smart environment technologies can be deployed for health monitoring. These include the design of activity recognition algorithms that generalize

over multiple individuals and that identify missing steps in the activity execution. Building on earlier work, we address whether learned models of activity recognition can be used to measure how completely and consistently activities are performed.

2. Methods

2.1. Data Collection

To validate our algorithms, we test them in a smart apartment testbed located on the WSU campus. The testbed is equipped with motion and temperature sensors as well as analog sensors that monitor water and stove burner use (see Figure 1). VOIP captures phone usage and we use contact switch sensors to monitor usage of the phone book, a cooking pot, and the medicine container. Sensor data is captured using a customized sensor network and stored in a SQL database.

To provide physical training data for our algorithms, we brought 20 WSU undergraduate students recruited from the psychology subject pool into the smart apartment, one at a time, and had them perform the following five activities:¹

1. *Telephone Use*: Look up a specified number in a phone book, call the number, and write down the cooking directions given on the recorded message.
2. *Hand Washing*: Wash hands in the kitchen sink.
3. *Meal Preparation*: Cook oatmeal on the stove according to the recorded directions, adding brown sugar and raisins (from the kitchen cabinet) once done.
4. *Eating and Medication Use*: Eat the oatmeal together with a glass of water and medicine (a piece of candy).
5. *Cleaning*: Clean and put away the dishes and ingredients.

¹ This data is available at ailab.eecs.wsu.edu/casas.

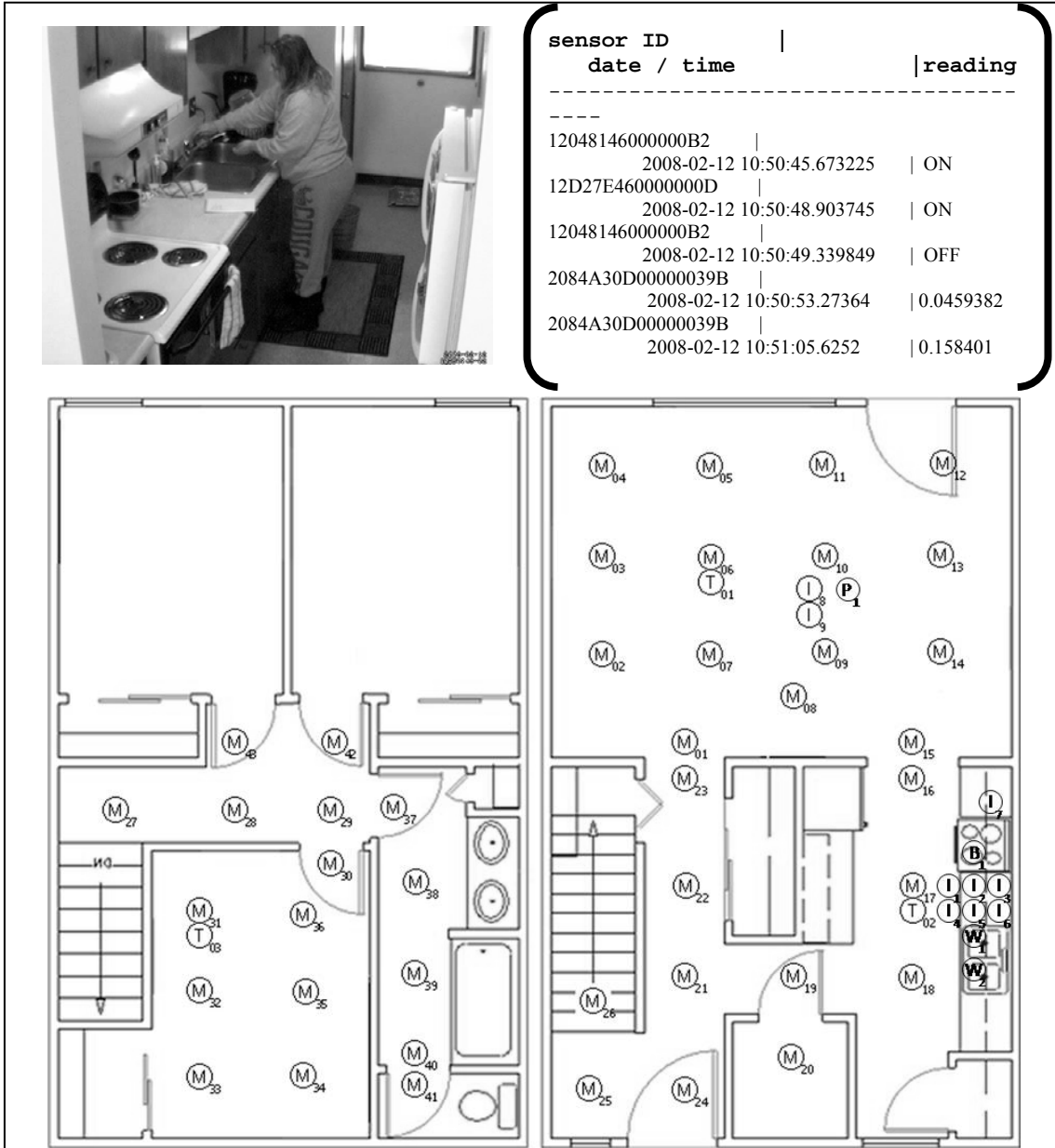


Figure 1 Resident performing “hand washing” activity (left). This activity triggers motion sensor ON/OFF events as well as water flow sensor values (right). Sensors in the apartment (bottom) monitor motion (M), temperature (T), water (W), burner (B), phone (P), and item use (I).

The selected activities include both basic and more complex ADLs that are found in clinical questionnaires [9]. Noted difficulties in these areas can help identify individuals who may be having difficulty functioning independently at home [10]. As shown in Figure 1, each sensor

reading is tagged with the date and time of the event, the ID of the sensor that generated the event, and the sensor value.

2.2. Modeling “Normal” Activities

Our first test set consists of sensor event data for 20 individuals who were asked to perform the 5 ADL activities, yielding a total of 100 activity traces containing 5,312 sensor events. This data reflects *normal* performance of the targeted activities.

The choice of representation has a tremendous impact on the accuracy and expressivity of the activity profiles. Researchers have investigated both the use of attribute/value based learners [11] and temporal models [12] for activity recognition. We implemented both a naïve Bayesian classifier and a Markov model (MM) to recognize the five activities for our normal group. Data features that we captured for both algorithms include the room location of the individual, on/off status of the water and burner, the open/shut status of the cabinet, and the absent/present status of the item sensors, as well as the number of seconds that elapsed since the previous sensor event. The individual’s location is indicated by a motion sensor ON event. To report water or burner usage, we installed analog sensors on the kitchen faucet and burner and report usage values whenever the difference in value from the previous second is above a pre-defined threshold. We use the raw values to interpret whether the device was turned up, down, on, or off. The cabinet sensors report OPEN or CLOSED based on the status of the door and the item sensors are load sensors that report PRESENT when an item of sufficient weight (above a few ounces) is placed on the sensor and report ABSENT otherwise.

A naïve Bayes classifier uses the relative frequencies of feature values, and the activity labels for the sample training data to learn a mapping from a data point description to a classification label. For our application, activities are represented by features including the number of times

during the activity that the water or burner was on/off, the phone was used, the cabinet was open/shut, items of interest were used, and the number of times the resident was at each location.

The activity label, A , is calculated as $\arg \max_{a \in A} P(a | D) = \frac{P(D | a)P(a)}{P(D)}$. In this calculation D

represents the feature values. The denominator will be the same for all values of a so we calculate only the numerator values, for which $P(a)$ is estimated by the proportion of cases for which the activity label is a (in our case each participant performed all five activities so there is a uniform probability over all activity values) and $P(D/a)$ is calculated as the probability of the feature value combination for the particular observed activity, or $\prod_i P(d_i | a)$.

A Markov Model (MM) is a statistical model of a dynamic system. A MM models the system using a finite set of states, each of which is associated with a multidimensional probability distribution over a set of parameters. The parameters for the model are the feature values described above. The system is assumed to be a Markov process, so the current state depends on a finite history of previous states (in our case, the current state depends only on the previous state). Transitions between states are governed by transition probabilities. For any given state a set of observations can be generated according to the associated probability distribution. Because our goal is to identify the activity that corresponds to a sequence of observed sensor events, we generate one Markov model for each activity that we are learning. We use the training data to learn the transition probabilities between states for the corresponding activity model and to learn probability distributions for the feature values of each state in the model.

To label a sequence of sensor event observations with the corresponding activity, we compute A as $\arg \max_{a \in A} P(a | e_{1..t}) = P(e_{1..t} | a)P(a)$. $P(a)$ is estimated as before, while $P(e_{1..t} | a)$ is the result of computing the sum, over all states, S , in model a , of the likelihood of being in each state

after processing the sequence of sensor events $e_{1..t}$. The likelihood of being in state $s \in S$ is updated after each sensor event e_j is processed using the formula found in Equation 1. The probability is updated based on the probability of transitioning from any previous state to the current state (the first term of the summation) and the probability of being in the previous state given the sensor event sequence that led up to event e_j .

$$P(S_j | e_{1..j}) = P(e_j | S_j) \sum_{s_{j-1}} P(S_j | s_{j-1}) P(s_{j-1} | e_{1..j-1}) \quad (1)$$

2.3. Measuring Activity Consistency

While recognizing activities is useful for evaluating the everyday functional activity performance of individuals, another important aspect of health monitoring is determining the successful completion of these activities. In order to provide an accurate functional assessment of individuals in their own environments, we need to monitor how consistently and completely activities are performed. We use our MM to provide this functionality in two ways. First, we provide a numeric measure of how close to normal each performance of an ADL activity is. Second, we identify specific steps which the resident skipped or performed incorrectly and that may be important to the successful completion of an activity.

To provide data that will be useful in evaluating our algorithms, we brought an additional 40 students into the apartment one at a time and asked them to perform the same sequence of 5 ADL activities as the normal group. For half of the participants, we included one specific error in task completion for each activity that we asked participants to demonstrate. In this condition, labeled *specific error*, all 20 participants made the same errors in activity completion and errors were selected to reflect common difficulties that can compromise everyday functional independence. For the remaining 20 participants, we provided general descriptions of task completion errors

and asked participants to simulate someone having difficulties with each ADL. To help guide participants with this undertaking, we first provided a scenario that included several examples of difficulties that an individual suffering from Alzheimer's disease might experience when completing everyday tasks (e.g., leaving appliances on, getting sidetracked or distracted during task completion, taking a long time to complete tasks). Participants were told to keep the scenario examples in mind as they completed the experiment. In this condition, labeled *simulation*, the types of errors that were introduced for each task were participant generated and were extremely varied.

- *Telephone Use:*
 - *Specific Error:* Dial a wrong phone number before retrying and successfully reaching the recorded message.
 - *Simulation:* Simulate someone who is having difficulty using the phone book and recording information about the recipe.
- *Hand Washing:*
 - *Specific Error:* Leave the water running after washing hands.
 - *Simulation:* Simulate someone who gets confused and becomes stuck completing the task.
- *Meal Preparation:*
 - *Specific Error:* Leave the burner on after cooking the oatmeal.
 - *Simulation:* Simulate someone having difficulty with the timing and order of steps involved in the task.
- *Eating and Medication Use:*
 - *Specific Error:* Forget to take medication with the meal.

- *Simulation*: Simulate someone completing the task steps in a slow and inefficient manner.
- *Cleaning*:
 - *Specific Error*: Wipe off the dishes without using running water to clean them.
 - *Simulation*: Simulate someone who becomes confused and moves off task and is later directed back to the task.

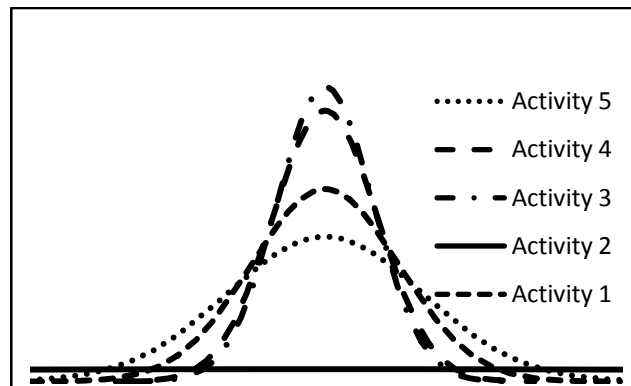


Figure 2. Plots of the normalized Gaussian distributions for each of the five activities. The peak of each curve is the likelihood mean for each model.

Detecting activity errors is important if we want to accurately monitor functional performance, to intervene if the error creates a hazardous situation and to provide activity reminders. In our approach, we use the model that was learned from the normal dataset to serve as a representation of expected behavior. First, we use the MM learned from normal data to measure the consistency (or conversely, the “anomalousness”) of an activity. The mean and standard deviations are calculated of all the model likelihood values that were calculated for the normal data set. The Gaussian distributions for the five activity models are shown in Figure 2 give an indication of how much variation is expected for the different activities.

ADL evaluation and anomaly detection start after the normal models are trained. Since the purpose is to evaluate the accuracy or consistency of the activity, the correct activity label for the

observed sequence of sensor events is known. In this second phase, the algorithm calculates the model likelihood for the observed sequence of sensor events. If the generated probability falls outside two standard deviations of the mean we flag the activity as “anomalous”, otherwise we label it as “consistent” with normal execution of the activity.

2.4. Detecting Skipped Steps

In addition to measuring the consistency of an activity, we also automatically detect specific steps of the activity that were skipped. This feature provides a basis for prompting residents to complete activities or automating steps (e.g., turning off the burner) that ensure the safety of the individual. We consider two approaches to detecting skipped steps. In the first approach, critical steps are identified by the user and our algorithm monitors execution of the ADL to make sure that each critical step is performed.

In our second approach to detecting skipped steps, we rely on our anomaly detection algorithm. When the normal model is trained, we keep track of the number of times each state is visited and the frequency with which the activity transitions from one state to the next. These values are used to generate normal distributions over state frequencies and transition frequencies. As a result, we can note each time a state is visited that is not normally visited. The algorithm can also report each state that is normally visited but was skipped during the current execution of the activity.

3. Results

We assessed the classification accuracy of our learned models using 3-fold cross validation on our 100 *normal* activity traces. The accuracy was measured as the number of activities that were correctly labeled, averaged over the three runs. On this data, the naïve Bayes algorithm achieved 91% classification accuracy while the MM achieved 98%. Significance values are gathered using

a paired student t-test and indicate that the spatial and temporal information contained in the MM is useful for activity classification although it does not significantly outperform NBC ($p \leq 0.28$) for this dataset. To further compare the classifiers, we compute the area under the ROC curve [13] for each classifier. ROC curves plot false positive vs. true positive rates. Our problem is a multi-class learning problem, but we can also view it as five separate binary class problems where each problem i attempts to learn true (positive) labels for activity i . The Bayes classifier AUC values for the five activities are 1.00, 0.94, 0.975, 0.82, and 0.85. The Markov model classifier AUC values are 1.00, 0.95, 1.00, 1.00, and 0.95. This analysis provides further evidence that the Markov model is a better-performing classifier for our activity recognition problem.

Interestingly, the two algorithms generated erroneous classifications on different activity instances. The naïve Bayes classifier labeled three cleaning activities as meal preparation activities. The number of motion sensor events that occurred in the kitchen area is very similar for these two activities. In the absence of the sequential relationships that the Markov models explicitly account for, the naïve Bayes classifier could not always adequately distinguish between these two activities. Both MM errors were due to labeling hand washing as cleaning dishes, which are in fact similar activities, distinguishable mainly by the amount of time that the water is on during the task. The fact that the models were trained on different individuals than they were tested on indicates that the models do have the capability to generalize over multiple individuals when performing well-defined types of activities.

We wanted to see how our algorithm would rate the consistency of the activities in the three datasets. After training the models on *normal* data, the models rated the consistency of the activities as shown in Table 1. While the results of the *error* dataset are expected, the number of

consistent activities in the *simulation* dataset is surprising. Looking at the *simulation* dataset, we see that many task completion errors generated by participants (e.g., difficulty locating number in phone book, trouble deciding on soap) resulted in additional time being needed to complete the tasks. This is detected by our algorithm if we compute anomalies based only on distributions created by relative event timings. In this case, an activity is considered consistent if the time taken to complete each step falls within the mean ± 2 stddev, and is considered anomalous otherwise. Using this criterion, only 75 of the 100 *simulation* activities are considered consistent.

Table 1. Number of activities from the normal, error, and simulation datasets evaluated as consistent or anomalous (outside mean ± 2 stddev).

Category	Normal Dataset	Error Dataset	Simulation Dataset	Simulation Dataset (timing alone)
# Consistent / Anomalous	95 / 5	76 / 24	96 / 4	75 / 25

Calculating the abnormality of each activity is beneficial for determining how well it was performed. These values can be used to determine an overall assessment of the functional well-being of residents in a smart home. On the other hand, not all steps of an activity have the same importance. For this reason, we next turn our attention to detecting specific steps or states that were missed while performing an activity.

To validate this algorithm, we processed each activity in the error datasets and reported the skipped step as missing if an error was detected by the algorithm. As Table 2 shows, for the specific error condition, all injected errors were detected for the washing hands and meal preparation activities. For the telephone use activity, none of the errors were detected because the software only detects when a completed call is made. When processing the eating and medicine activities, the skipped state was noted for all but one case. In that case the participant did not follow the experiment instructions and removed the medicine bottle from the cabinet. As a result,

the algorithm correctly identified the activity as having been fully completed. For the cleaning activity, the skipped state was noted in all but two cases. In the first case, the participant did not skip the state as instructed and washed the dishes with water from the sink. In the second case, a spurious sensor event was generated by the water sensor and thus falsely indicated that water was used. Such noise in the data is common for sensor-rich pervasive environments and highlights ongoing challenges that need to be addressed in this and related research projects.

Table 2. Summary of detected mistakes for the specific *error* and *simulation* datasets.

Activity	Error	#Cases (Specific Error data)	#Cases (Simulation data)
Telephone Use	Multiple calls	0	3
Hand Washing	Turn burner on	1	
Hand Washing	Open cabinet	1	
Hand Washing	Leave water on	18	2
Meal Preparation	Leave water on	18	
Meal Preparation	Went back into living room	1	
Meal Preparation	Put items back into cabinet	5	
Meal Preparation	Leave burner on	19	6
Eating & Med Use	Forget medicine	19	
Cleaning	Do not use water	18	
Cleaning	Make phone call		1
Cleaning	Turned burner on		1
Cleaning	Leave water on		1
Cleaning	Leave items out on counter		1
All	Wander around apartment	1	2

Interestingly, a number of additional errors were detected as well. For example, one participant turned on the burner during the hand washing activity in preparation for the upcoming cooking activity. Another participant opened the cabinet during the hand washing activity, and

several participants put items such as the oatmeal back in place while they were cooking instead of waiting for the cleaning activity to start before performing these steps. In addition, because the water was left on at the end of the hand washing activity, it was still on when the participant began the meal preparation activity. If we performed error detection on the entire sensor event sequence as a whole, without segmenting the data, many of these “mistakes” might be considered acceptable as they are part of a later activity. This type of activity profiling necessitates recognition and tracking of interleaved activities, which we will pursue in our ongoing research.

With regards to the *simulation* dataset, a number of the same errors were detected as in the specific *error* dataset because the participants chose to inject these errors (e.g., left burner on, left water running). In addition, our algorithm noted multiple completed phone calls (the participant “forgot” they had already made the call), roaming around the apartment during an activity, not turning the burner on during cooking, and not putting items back in the cabinet.

We note that there a number of mistakes in the *simulation* data set that our algorithm did not detect. For example, incorrectly measuring the oatmeal, not using soap when cleaning, washing hands multiple times, “forgetting” where items were located, and using too much soap. Because there were no specialized sensors for the items involved in these steps, the algorithm did not detect these kinds of errors and future research will be needed to determine what types of undetected errors pose risks for functional independence and should be monitored. Of note, some of these difficulties will manifest themselves in the amount of time taken to complete an activity and will impact the quality or consistency measure for the activity as a whole.

4. Discussion

The goal of this project was to design an algorithmic approach to identifying and assessing the consistency of an activity performed in a smart environment. Our experimental results indicate that it is possible to distinguish between activities that are performed in a smart home and to label a sensor event stream with high accuracy. Other approaches have been considered for activity recognition; however, these often depend on the participants wearing customized devices [6]. On the other hand, researchers have not investigated the issue of assessing the quality or consistency of a performed activity.

We note that when our algorithms were used to assess the activity as a whole, only 24% of the activities in the error dataset were marked as anomalous and 4-25% of the simulation-based erroneous activities were marked as erroneous. This is due primarily to the fact that the amount of error (affected by the number of steps that were incorrect) was small in relation to the amount of variance that occurred for the activity even under normal conditions. This highlights the fact that to detect errors we need to not only provide an overall assessment of the activity consistency but also to look for specific classes of errors.

Our study revealed that Markov models are effective tools for assessing activity consistency as well as providing a label for the activity. However, there are more complex monitoring situations that we need to consider before the technology can be deployed. First, we need to perform activity recognition and assessment in cases where multiple activities are interleaved. We also need to design algorithms that perform accurate profiling in environments that house multiple residents. In addition, we need to design method for distinguishing critical activity errors from inconsistencies which results from the individual performing activities in a creative manner.

In our data collection, an experimenter informed the participants of each activity to perform. In more realistic settings, such labeled training data will not be readily available and we need to research effective mechanisms for training our models without relying upon excessive input from the user. Another feature that will occur in real-world situations is the occurrence of activities that are interleaved or performed in parallel. For example, a resident may cook oatmeal at the same time that they watch television or talk on the phone. We hypothesize that ADL recognition and assessment can be performed in such situations and our future studies will evaluate the ADL recognition and assessment algorithms in the context of such interleaved activities. Finally, we encountered a limitation in our study in that one participant in the *error* study did not embed the requested error in the activity and the embedded errors in the *simulation* dataset contained a great deal of variation. Future studies will need to correct this situation to allow for more thorough analysis of the ADL assessment algorithm.

5. Conclusions

In this work we described an approach to characterizing the quality of activities performed by smart home residents. In particular, we designed an algorithm and demonstrated, using data collected in a physical environment, that the algorithm not only recognizes activities but also assesses the completeness and accuracy of the performed activity using anomaly detection.

Ultimately, we want to use our algorithm design as a component of a complete system that performs functional assessment of adults in their everyday environments. This type of automated assessment also provides a mechanism for evaluating the effectiveness of alternative health interventions. We believe these activity profiling technologies are valuable for providing automated health monitoring and assistance in an individual's everyday environments.

References

1. 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.
2. Wadley V, Okonkwo O, Crowe M, Ross-Meadows LA. Mild Cognitive Impairment and everyday function: Evidence of reduced speed in performing instrumental activities of daily living. *American Journal of Geriatric Psychiatry*. 2007;16(5):416-424.
3. Cook DJ, Das SK, editors. *Smart Environments: Technology, Protocols, and Applications*. Wiley, 2004.
4. Doctor F, Hagrais 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(1):55-56.
5. 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(1/2):135-171.
6. 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(4):50-57.
7. Sanchez D, Tentori M, Favela J. Activity recognition for the smart hospital. *IEEE Intelligent Systems*. 2008;23(2):50-57.
8. Wren C, Munguia-Tapia E. Toward scalable activity recognition for sensor networks, *Proceedings of the Workshop on Location and Context-Awareness*, 2006.

9. B. Reisberg, Finkel S, Overall J, Schmidt-Gollas N, Kanowski S, Lehfeld H, et al. The Alzheimer's disease activities of daily living international scale (ASL-IS). *International Psychogeriatrics*. 2001;13;163-181.
10. Schmitter-Edgecombe M, Woo E, Greeley D. Memory deficits, everyday functioning, and mild cognitive impairment. *Proceedings of the Annual Rehabilitation Psychology Conference*, 2008.
11. 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.
12. 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.
13. T. Fawcett. ROC graphs: Notes and practical considerations for researchers. HP Labs Technical Report HPL-2003-4, 2004.