Chapter 4
Behaviometrics for Identifying Smart Home Residents

Aaron S. Crandall and Diane J. Cook

Smart homes and ambient intelligence show great promise in the fields of medical monitoring, energy efficiency and ubiquitous computing applications. Their ability to adapt and react to the people relying on them positions these systems to be invaluable tools for our aging populations. The most privacy protecting and easy to use smart home technologies often lack any kind of unique tracking technologies for individuals. Without a built-in mechanism to identify which resident is currently triggering events, new tools need to be developed to help determine the identity of the resident(s) in situ.

This work proposes and discusses the use of *behaviometrics* as a strategy for identifying people through behavior. By using behaviometrics-based approaches, the smart home may identify residents without requiring them to carry a tracking device, nor use privacy insensitive recording systems such as cameras and microphones. With the ability to identify the residents through behavior, the smart home may better react to the multitude of inhabitants in the space.

4.1 Introduction

"Smart homes" represent a rapidly maturing field of study as well as a looming business market. Its concepts are being applied to a wide range of medical, social and ecological issues. The vague definition of "smart home" allows for numerous implementations and variations to exist. At its core, a smart home is any living space that involves sensors, controllers and some kind of computer-driven decision

A. S. Crandall (✉) · D. J. Cook
School of Electrical Engineering and Computer Science, Washington State University, Pullman, WA, USA
e-mail: aaron.crandall@email.wsu.edu
D. J. Cook
e-mail: djcook@email.wsu.edu

T. Bosse et al. (eds.), *Human Aspects in Ambient Intelligence*,
Atlantis Ambient and Pervasive Intelligence 8, DOI: 10.2991/978-94-6239-018-8_4,

making process. The addition of a proactive and intelligent decision maker to the aspects of home automation is what produces a smart home. With this loose definition in hand, the research, medical and business communities have been highly creative in leveraging this concept for their various needs.

The area with the greatest long-term feasibility for smart home commercialization is health care, though energy efficiency has a strong future in reducing our home's economic and ecological footprint. For the health care community, the ability to monitor older adults in their home to support "aging in place" [1, 2] is of significant interest.

Ideally, a smart home is subtle in its operation and conforms [3, 4] to the residents without detrimentally impacting their lifestyle. The system should take in information about the home environment and attempt to build models about the activities and interests of the residents. To make smart homes capable of supporting this goal, the research community has focused on building technologies for the detection of the Activities of Daily Living (ADLs) [5, 6], resident tracking [7, 8], resident identification [9], medical history building [2], social interaction [10], resident mental evaluation [11], and many others.

This work addresses methods of determining the residents' identities without a wireless tracking tag or biometric identifiers, such as facial recognition. With a single resident in the smart home this is a trivial problem, but multiple residents transform it into a serious issue. As soon as a second person (or other entity, such as a pet capable of causing sensor events) enters the smart home space, the multi-resident issue becomes critical. At this juncture, the smart home infrastructure must be designed to either function well in the face of several sources of data, or to differentiate between the sources by some means. If the system ignores the multi-resident problem, unaccounted for residents show up as noise in the data. In most cases, this noise in the data will lower the accuracy of algorithmic model building and interfere with operational quality. It will likely cause failure of high quality resident history building, preference generation, ADL detection, and many other computer generated models. Finding a means to address the multiple-resident problem remains a current and pressing issue for the smart home field.

To identify an anonymous resident using only low fidelity sensors, this research project leverages the concept of behaviometrics. Behaviometrics is the use of behavior to identify an actor among the group. These approaches hold more promise for smart home applications due to their position as ambient tools instead of obvious and intrusive ones that are common with biometrics. Examples of biometrics include facial recognition [12], body shape [13], fingerprints, and many more [14]. Conversely, behaviometrics use behaviors such as handwriting recognition [15], gait recognition [16], and computer interaction [17]. There have been few papers within the smart home field to date that use behaviometrics to identify individuals [18, 19]. New tools, when combined with the numerous data sources of a smart home system, allow the opportunity to determine a resident's identity via interaction with the smart home.

The hypothesis of this work is as follows: people have a variety of algorithmically differentiable behaviors that simple sensors can provide evidence of. The Center for Advanced Studies in Adaptive Systems (CASAS) at Washington State University (WSU) uses ubiquitous, passive and simple sensors to enroll individuals in the behaviometric system for future identification [9, 20]. Given a unique historical profile, a resident can then be re-identified in the future using behavior alone. This work introduces real world smart home testbed implementations and algorithms based on statistical models that leverage this concept of behaviometrics to accurately identify which person is generating events.

4.2 Project Research Testbed and Data Used

The data used for this project comes from a "real world" smart environment. This section describes the environment, as well as the form of the data collected and its processing for use in testing the algorithms.

4.2.1 Kyoto Research Testbed

The data gathered for this work comes from a smart home research testbed at the CASAS research facility. The *Kyoto* testbed is the primary research facility for the CASAS projects. This three bedroom apartment shown in Fig. 4.1 is part of the WSU University Housing system, and is ordinarily the home of two undergraduate students. *Kyoto* is designed to be a sensor-rich space designed for capturing as many ADLs and behaviors as possible. The left section in the image is the second story with bedrooms and the bathroom. The right side includes a living room, kitchen, and closets. The sensors are noted by rectangles and ovals, with the M### being motion detectors, L### lights, and D### doors.

Since its initial installation in 2007, this smart home testbed has undergone a series of improvements. These have primarily been software updates, but over time new sensors and interactive technologies have been deployed. These have focused on supporting the CASAS research objectives, such as early onset dementia evaluation and aging in place tools, although *Kyoto* is also used for studies regarding the associating of activities with energy consumption. This testbed has proven highly successful at gathering rich and well-documented data sets, many of which are available publicly [21].

The *Kyoto* testbed, also known as the "smart apartment" in many CASAS works, is representative of many American living spaces. Each resident has their own room with a bed, desk and closet. There is a shared bathroom, living room and kitchen. This resemblance to many typical homes makes the results from the research done here more generalizable than partial smart home implementations or work done with specialized facilities.

The sensor layout of *Kyoto* is dense, and fairly regular in design. The primary sensor type is the downward facing PIR Motion Sensor. These are installed on the

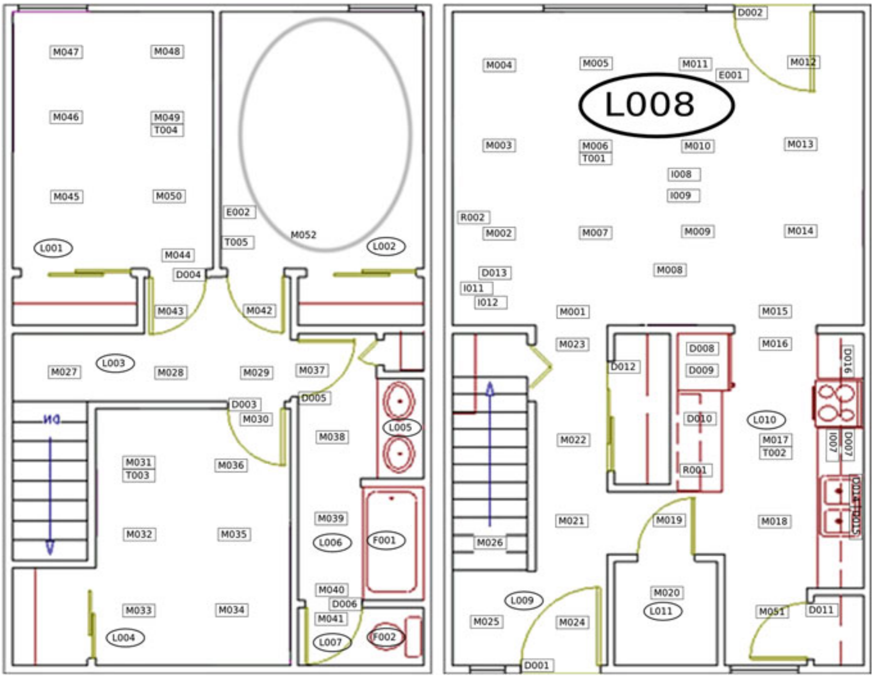

Fig. 4.1 Floorplan and sensor layout for the CASAS *Kyoto* testbed

2.4 m high ceilings with a field of view that covers roughly a 1.2 m × 1.2 m section of the floor. This sensor distribution is designed to provide enough resolution for human annotators and algorithms to localize and track the residents.

The rest of the sensors are installed on an "as needed" basis. Many of them have very specific uses to aid the artificial intelligence algorithms in their operation or to give human annotators information about the activities being performed. More detail on the *Kyoto* facility may be found in other CASAS publications [8].

4.2.2 Data Format and Annotation

The data collected from *Kyoto* was selected because it encompasses well defined times when multiple residents were occupying the space. The CASAS team did not intervene with the residents while they lived in the smart home spaces and no attempts were made to adjust their behavior over time. The residents were consulted about their behaviors to ensure an accurate final ground truth during the data annotation period.

After annotation processing, the data has five fields as shown in Table 4.1. The first four fields are generated automatically by the CASAS middleware at the time

Table 4.1 Data provided by every event, including annotation tag for person's identity

Field	Notes
Date	ISO 8601 format (yyyy-mm-dd)
Time	ISO 8601 time format (hh:mm:ss.subsec)
Serial	Unique text identifier for sensor reporting
Message	Value of sensor event
Id	Annotation tag for person causing event

Table 4.2 Example of data used for classifier training

Date	Time	Location	Message	ID
2007-12-21	16:41:41.0764	L017	ON	Res1
2007-12-21	16:44:36.8230	L017	OFF	Res1
2007-12-24	08:13:50.2819	L007	ON	Res2
2007-12-24	14:31:30.6889	L007	OFF	Res2

Table 4.3 Summary of data sets used for validation of identification algorithms

Data Set	Residents	Length	Num Events
B&B	2	5 days	20,000
TwoR	2	56 days	136,504

of the event's creation. The annotated class field is the target feature for our learning problem and contains the resident ID, to which the other fields can be mapped. An example of the data from these two data sets may be seen in Table 4.2.

The first data set, labeled *B&B*, we collected sensor data from the *Kyoto* smart apartment while two residents lived there. This data set assesses the basic ability of our algorithms to identify residents even when they occupy the space simultaneously with little training data to learn with. Both residents occupied a separate bedroom, but regularly shared the common space downstairs.

The other data set, labeled the *TwoR*, contains sensor events collected over a period of eight weeks while two residents (different than those in the *B&B* data set) lived in the *Kyoto* smart apartment. As with the *B&B* data set, this was collected to evaluate the mapping of sensor events to specific residents. However, we also used this data set to test ADL detection with other algorithms. To demonstrate the benefits of first determining the resident ID for an event on ADL detection, we performed this activity recognition first without resident identifier information and then second when the data is enhanced by adding the automatically-labeled resident identifier to each sensor event. In this manner, we determined how well residents may be recognized and the degree to which this information aids in other multi-resident tasks such as activity recognition.

4.3 Algorithms

Two algorithms were developed to test the hypothesis that behaviometrics can be used to identify individuals in a smart home space. They are based on well established machine learning algorithms and applied here to the smart home domain. Each of these tools has requirements for operation and provides unique benefits when used to identify the current residents in a smart home space. These two algorithms are:

(1) NB/ID: A naïve Bayesian-based tool
(2) HMM/ID: A Hidden Markov Model-based tool

4.3.1 Naïve Bayes: NB/ID

The first algorithm built and tested for identification was based around a Naïve Bayes classifier. In our study this tool was designated the Naïve Bayes / IDentifier (NB/ID). This classifier leverages Bayes' Rule to use the current event received to guess at the identity of the individual. Naïve Bayes classifiers have been used to good effect in other smart home contexts [22, 23]. The location, message and time features from individual events were exploited to determine the resident's identity.

A Naïve Bayes classifier uses the relative frequency of data points, their feature descriptors, and their labels to learn a mapping from a data point description to a classification label. The resident label, r, is calculated as shown in Eq. 4.1.

$$arg\ max_{r \in R}\ P(r|D) = \frac{P(D|r)P(r)}{P(D)} \tag{4.1}$$

In this calculation, D represents the feature values derived from the event to be classified. The denominator will be the same for all values of r, so we calculate only the numerator values (Table 4.3). The numerator is made of $P(r)$, which is estimated by the proportion of cases for which the resident label occurs overall and $P(D|r)$ which is calculated as the probability of the feature value combination for the particular observed resident id, or $\Pi_i\ P(d_i|r)$.

4.3.1.1 NB/ID Data Features

For a given event, the resident ID is set by the annotation process, but the feature representing that event can be derived in a variety of ways. We could attempt to use only location and message information as input to the learning problem, as shown in Table 4.4, type 1 (e.g. "Plain"), but this leaves out valuable temporal information about the resident behaviors. The remaining features, date and time, are more difficult to use. Both of these features have a very large number of possible values, so we were required to consider effective methods for abstracting date and time information.

Table 4.4 Naïve Bayes alternative time-based feature formats

Type #	Feature type	Example
1	Plain	M001#ON
2	Hour-of-Day	M001#ON#16
3	Day-of-Week	M001#ON#Friday
4	Part-of-Week	M001#ON#Weekday
5	Part-of-Day	M001#ON#Afternoon

The different feature choices that could be considered for these values, as shown in Table 4.4, divide the data in different ways and capture resident behaviors with varying degrees of fidelity.

The "Plain" feature set provides a good baseline to compare with more complex parsings. The more complex parsings, such as Part of Week (e.g. Weekday or Weekend) capture more information about the given behavior, and can furnish the classifier with more information for correct future classifications. Depending on the facets of the data set, different feature types will cause the classifier to perform better or worse.

The different feature choices available (e.g. Plain vs Hour-of-Day, etc.) divide the data up in different ways. Each method captures the behaviors or the residents with varying degrees of accuracy, depending on the feature types chosen and the behavior of the individuals in the data set.

The purely statistical nature of a Naïve Bayes classifier has the virtue of being fast for use in prediction engines, but lacks the ability to incorporate a greater context contained within the event stream that often are advantageous in discerning subtle differences in behaviors. We test the accuracy of each of these time representations when we evaluate the NB/ID algorithm.

4.3.1.2 NB/ID Summary

The statistical calculations of a Naïve Bayes classifier offer the benefit of fast learning, but lack an effective approach to reasoning about context in an event stream. In order to capture this context we also consider other approaches to learning resident IDs, as described in the next section.

4.3.2 Hidden Markov Model: HMM/ID

With this algorithm, a single model is used to encapsulate all of the residents and the sensor events they trigger. This HMM/ID tool was used to evaluate a Hidden Markov Model's ability to properly attribute events to residents.

Using the HMM, hidden nodes represent system states that are abstract and cannot be directly observed. In contrast, observable nodes represent system states that can

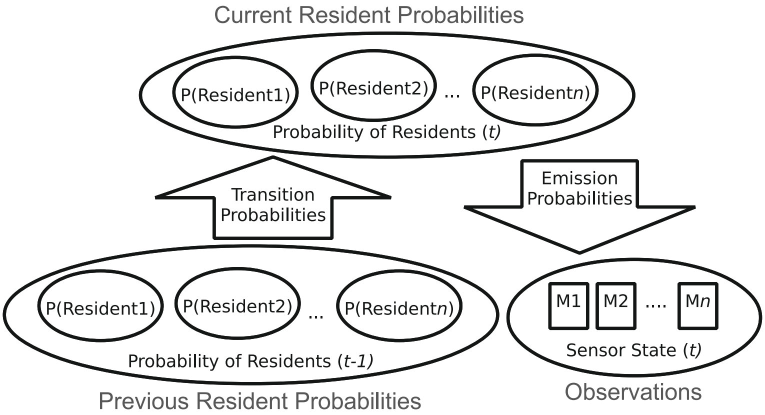

Fig. 4.2 HMM architecture of hidden states, transitions and observations

be directly observed (sensor recordings). Emission probabilities between hidden and observed nodes are learned from training data, as are transition probabilities between prior and current hidden nodes.

In our model, as shown in Fig. 4.2, each hidden node represents a single resident. The observable nodes are associated with probability distributions over feature values including the motion sensor ID and the sensor message. We can then use the Viterbi algorithm [24] to calculate the most likely sequence of hidden states that corresponds to the observed sensor sequence. This sequence of hidden states provides us with the highest-likelihood resident IDs that correspond to each sensor event in the sequence.

In the HMM/ID structure, the states Y represent the n residents from the training corpus. The start probabilities of each resident state in Y are kept in π, while the transition probabilities $a_{i,k}$ represent the likelihood of transitioning from resident Y_i to resident Y_k between events. The probability of a resident causing an event (e_t), called their emission probability, is denoted by $P(e_t|Y_i)$. If the testing corpus of sensors events is e_0, ..., e_t, then the state sequence s_0, ..., s_t is the most likely attribution of these events to the residents represented by the states. This mapping is given by the recurrence relations in Eqs. 4.2 a–c. The result of V_t (the probabilities of all residents at event t) is the probability of the most probable series of resident attributions for the first t−1 events, followed by the most likely resident at time t.

$$V_0 = \forall y \in Y : P(e_0|y) \cdot \pi_y \tag{4.2a}$$

$$V_t = \forall y \in Y : P(e_t|y) \cdot \prod_{1..n}^{k}((a_{y,Y_k}) * V_{t-1_k}) \tag{4.2b}$$

$$s_t = argmax\ V_t \tag{4.2c}$$

The events are taken one at a time without modification or manipulation, leaving the capabilities of the system entirely up to the ability of the algorithm and not choices made during pre-processing stages. The trade-off is that the tool often requires more than one event to transition between residents. It relies on some context-dependent amount of evidence for the HMM to transition from one hidden state (resident ID) to another. This sometimes leads to a delay in proper identification during operation, and is a source of error in the results. The behavior of the HMM for both "transition lag error" and "confusion error" are both discussed in Sect. 4.4.

4.3.2.1 HMM Summary

HMMs are robust in the face of noisy data and used for a number of smart home applications. The HMM/ID tool developed for classifying residents is based on a classic HMM approach and eliminates a number of shortcomings to the NB/ID tool developed earlier. This more complex algorithm reacts to the data in such a way that introduces multiple sources of error that are discussed in depth in Sect. 4.4.

4.4 Evaluation and Results

The identification algorithms introduced in Sect. 4.3 were evaluated with the data sets introduced in Sect. 4.2. The *B&B* and *TwoR* data sets are complex in nature and provide an overall evaluation of these identification tools.

Additionally, the ability for the identification results to boost ADL detection were tested with the *TwoR* data set. This test was done to demonstrate the ability for identification to provide additional features that may improve other models in the smart home context.

4.4.1 B&B Data Set Results

The *B&B* data set involves two residents simultaneously inhabiting the *Kyoto* testbed. It is a relatively short data set of five days, but does have the benefit of being occupied nearly the full 120 h of its duration. The interleaved resident tags and the cumulative evidence for an individual's identity effecting the behavior of the NB/ID and HMM/ID models are good demonstrations of how behavior-based identification might work in a smart home context.

4.4.1.1 B&B Evaluation

The NB/ID and HMM/ID classifiers were tested using 30-fold cross validation. Each classifier was trained on 29 out of 30 groups and tested on the remaining one. The results from all 30 permutations were averaged together for an overall accuracy,

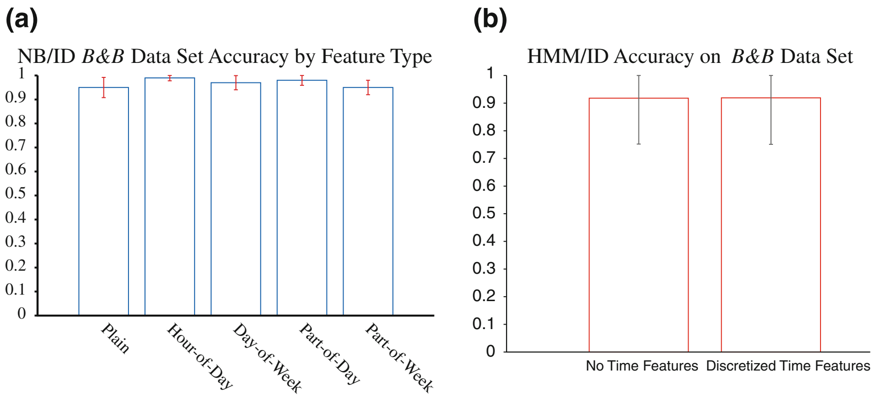

Fig. 4.3 Results for *B&B* data set. **a** NB/ID results and **b** HMM/ID results

and their variance calculated for significance values. Additional statistics showing the behavior of the classifiers and the data sets were gathered for insight into the capabilities of the tools.

The results for the tests on this data set are shown in Fig. 4.3. As can be seen, both the NB/ID and HMM/ID achieve very high classification accuracies on this two-resident, parallel-activity data. The two algorithms tested performed statistically equally on this data set. We hypothesize that having only two classes for the Naïve Bayes to choose from benefits it inordinately, at least for the behaviors exhibited by these two residents.

We found that using the Hour-of-Day gives the best results, and is a significant ($p < 0.05$) improvement over the Plain feature. Surprisingly, the inclusion of discretized time values in the HMM/ID feature vector demonstrates no benefit for the *B&B* data set. This demonstrates how both temporal and spatial information have different values for varying environments and residents. Continued efforts to discover the most valuable combination of features for identifying individuals needs to be pursued.

4.4.1.2 B&B Results Summary

The ability of our models to perform well in this unscripted, full-time, multi-resident environment is encouraging. These kinds of classifiers should be able to provide better tools for discerning an individual's activity history, even in complex multi-resident environments.

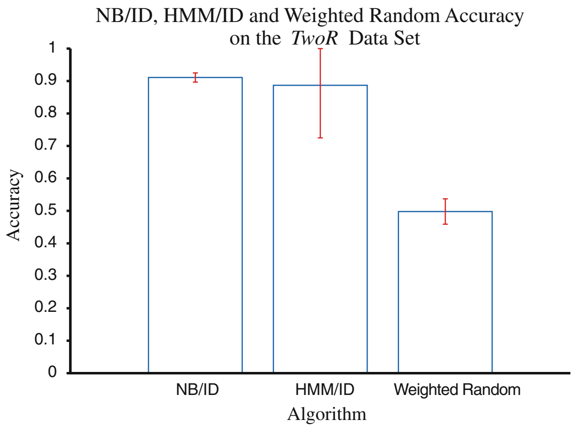

Fig. 4.4 *TwoR* data set accuracy for the NB/ID, HMM/ID and Weighted Random algorithms. The *error bars* show two standard deviations

4.4.2 TwoR Data Set Results

The *TwoR* data set provides the largest corpus of data of the three identification data sets. It has the most complex behaviors and social interactions as well. Like the *B&B* data set, the NB/ID and HMM/ID tools were evaluated for accuracy. Additionally, a more in-depth look at the behavior of the HMM/ID is discussed. Given the interleaved and social nature of the residents, the *TwoR* data exposes the various sources of error for the HMM/ID algorithm.

4.4.2.1 TwoR Evaluation

As with the evaluation of the classifiers with the *B&B* data set in Sect. 4.4.1.1, the classifiers were tested using 30-fold cross validation. Additionally, their results were compared to a Weighted Random algorithm as a base case. Each classifier was trained on 29 out of 30 groups and tested on the remaining one. The results from all 30 run permutations were averaged together for an overall accuracy, and their variance calculated for significance values. Additional statistics showing the behavior of the classifiers and the data sets were gathered for insights into the capabilities of the tools.

Both algorithms performed well on the *TwoR* and *B&B* data sets and were significantly ($p < 0.01$) better than a Weighted Random algorithm introduced as a base case for comparison. The overall accuracy of the algorithms are shown in Fig. 4.4. The HMM/ID performed slightly better than the NB/ID, though not significantly so.

Given the complexity of the data with multiple residents, and no given structure to their behavior, the highly accurate results from both algorithms attest to their robustness. Overall, the HMM/ID results are very promising. The initial hypothesis

Table 4.5 Example HMM/ID transition behavior pattern

Event number	Annotated class	Chosen class	Result
1	R1	R1	SUCCESS
2	R1	R1	SUCCESS
3	R2	R1	FAIL
4	R2	R2	SUCCESS
5	R2	R2	SUCCESS

that drawing on additional contextual information across a series of events would allow an algorithm to better differentiate between individuals seems to be supported by the overall accuracy results.

The behavior of the HMM/ID is more complex than the NB/ID when analyzing the actual pattern of classification. As the events arrive, it takes the HMM zero or more additional events to determine to whom the new events belong. For an example of this behavior, Table 4.5 shows a small snippet of events as classified by the HMM/ID. The left column is the event number, the second represents the annotated resident value for the event, the third the algorithm determined, and the final column being the success or fail results for the given event. This snippet has a transition from R1 to R2 at event #3. The HMM delays until event #4 before it has enough evidence to change states and begins attributing events correctly. This situation, where the events change from one resident to another, has been termed a resident "transition" and is an important feature of HMM/ID algorithm behavior.

By the overall accuracy metric this example has a score of 4/5, or 80 % accuracy. What is most interesting about this series is that the events arriving at the computer are initially from R1, then change to R2 at some point, but the HMM/ID algorithm takes extra events to properly transition as well. In contrast, the NB/ID algorithm takes every event in isolation, so there is no previous context to consider. With the HMM/ID algorithm, there is now a possibility of a transition window as the evidence that the new events are from a different person accumulates. The concern is that this transition window would significantly impact the effectiveness of the HMM/ID as a tool for identification.

To determine how much this transition error is effecting the HMM/ID, several statistics were gathered from the final tests. The first was the total number of occurrences in the event stream where the annotated resident value switches from one to another. This is an indication of the data complexity. If the number of transitions increases it indicates more simultaneous occupancy of the space, which can be more difficult for the HMM/ID to accurately classify.

The hypothesized inverse relationship between the rate of transitions in the data set and the final accuracy was not borne out by the results, as shown in Fig. 4.5. The transition rate line was expected to trend upward, opposite the overall accuracy across the data sets used to test the classifier. Instead it is found to trend with the accuracy, with slopes of −0.038 and −0.046 respectively. On further inspection, it is not merely the number of transitions that effects the overall accuracy, but also the location within the smart home of the residents during those transitions. If the

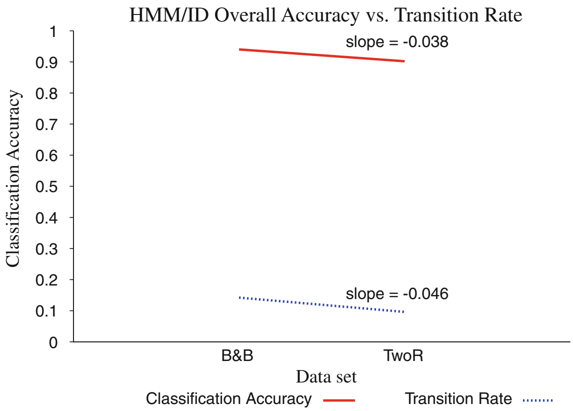

Fig. 4.5 HMM/ID's overall accuracy for each data set, with the data sets comparative transition rate. The transition rate was expected to trend opposite to the classification accuracy instead of with it

Table 4.6 HMM average transition delay length for both *B&B* and *TwoR* data sets. An average of zero would represent perfect transition accuracy

Data set	Average delay (in events)	Standard deviation (σ)
B&B	0.19	0.80
TwoR	0.38	2.17

entities' behaviors are physically close to one another, there is less evidence in the emission probabilities that the HMM should change its hidden state, and thereby transition correctly in its classifications.

If the residents are physically close together the difference in emission probabilities is lower, which causes the HMM to be less capable in detecting transitions between residents. In the *B&B* data set, the residents spent notably less time sharing communal spaces than was found in the *TwoR* data set.

As a measure of how much the delay in transition impacts the behavior of the algorithm, some additional analysis about the length of the delay was gathered. The relevant data is the average number of events after a transition before the HMM properly changes to accurately classify the resident. To find this value, the results were processed for the length of the delay in the transition on each data set.

This delay in the HMM after transitions in the data is a notable portion of the HMM's overall error. Table 4.6 shows the average length of the delay in the HMM transition for each data set. An average of zero would mean that it has no delay whatsoever on the given data set, leading to perfect classification during transitions. The lower average delay for the *B&B* data set is consistent with the overall higher accuracy as compared to the *TwoR* data set. This indicates that the HMM was able to use the evidence to accurately transition between residents based upon their behavior in the sensor space. The *TwoR* residents were notably more social than the *B&B* residents, and spent more time near one another in communal spaces during their

Table 4.7 HMM/ID non-transition error rates

Data set	Error
B&B	3.2 %
TwoR	6.1 %

stay in the testbed. Because they spent more time in close proximity, the resolution of the sensor network had more trouble providing evidence for the HMM to determine who was whom during the close interactions, causing the overall accuracy to suffer.

The other sign the *TwoR* residents were more often interacting during the time of this data gathering is the longer lengths of the HMM's transition delay. With the *B&B* data set, there were very few instances where the HMM was not able to properly transition within one or two events. This indicates that the residents were most often physically separated in the testbed space. The very long delay lengths induced by the *TwoR* were observed to be when the two residents were performing activities like cooking or homework together. In those cases, the lack of physical separation meant that the HMM was unable to differentiate between the residents for quite some time.

Another source of error in classification occurs when the HMM outright chooses the incorrect class, but there was no actual transition to another resident. In this case the algorithm is truly confused, and this error type is more akin to the type of error in the NB/ID. The total error rate for this kind of mis-identification is summed up in Table 4.7. The higher rate for the *TwoR* data set indicates that these two individuals had more behavior that was similar to one another than the two people in the *B&B* data set, which again contributes to the lower overall accuracy on the *TwoR* set.

4.4.2.2 TwoR Results Summary

Encompassing a much larger selection of behaviors over a longer time than the previous data set, the *TwoR* data set represents a valuable tool for evaluating behaviometric-based identification algorithms. The residents are closer in behavior to one another than those found in the *B&B* data set, which leads to more opportunities to inspect the behavior of the algorithms themselves. These additional hurdles provide opportunity for future identification algorithms to improve on those presented here.

4.4.3 Identification ADL Boosting

As a final demonstration of the usefulness of these identification algorithms, their ability to aid the performance of other types of smart environment tasks needed to be evaluated. Specifically, we apply the NB/ID classifier to the *TwoR* data set to map sensor events to resident IDs. Given this additional identity feature, we then use a separate Naïve Bayes classifier to identify which of 14 possible activities the residents are individually, but concurrently performing. We evaluate the performance of activity recognition with and without the learned resident identification to determine the extent to which the resident ID actually improves performance of our activity recognition algorithm.

The Naïve Bayes classifier initially achieved an accuracy of 80.8 % on this data set. This is a good result as compared to other published ADL detection tools, especially given the number of activities that we need to discriminate and the fact that residents are performing activities in an interwoven and parallel fashion.

To determine how activity recognition can benefit from learned resident information, we next enhance the *TwoR* data set by adding an extra field to each sensor event containing the resident ID that is automatically generated by the NB/ID classifier. We test our activity recognition algorithm again on this enhanced data set, and this time achieve an accuracy of 89.2 %. The results clearly demonstrate learned resident labels enhance the accuracy of other smart environment tasks such as activity recognition.

4.5 Conclusion

The algorithms introduced and explored in this chapter demonstrate the ability of behaviometrics to algorithmically identify smart home residents. They each leverage different aspects of the smart home data and react differently to various quantities and behaviors of residents. They are all demonstrably better than random guesses and provide additional insights into the workings of the smart home system.

The approach of using simple, passive, low resolution sensing environments with the algorithms introduced in this work generated results similar to those using other identification strategies. Controlled facial recognition approaches can see accuracies in the mid to high 1990s [25], height recognition in doorways may be 95 + % [26] and footstep and stride recognition has shown results around 87 % accurate [18]. Even RFID-based systems have some error in determining the identity of the RFID tag in real world implementations. This can lead to RFID accuracy rates of only 60–70 % [27], though repeated readings will likely overcome a single erroneous transmission. Depending upon the intended use and environment, these different approaches may have more or less utility for a given smart home installation. In the long run, some combination of available strategies will likely become the most successful behaviometric identification methods.

By applying these kinds of tools to the smart home data and generating a resident ID feature, ADL detection is boosted in complex, real world environments. Any modeling tools that improve the ability of smart homes to be functional and usable are important. Using algorithmic approaches to detect identity is a necessity for large scale deployments of smart home technologies that cannot have wireless devices affixed to every resident for identification purposes. The tools introduced and evaluated in this chapter initiate inquiry into these issues for the smart home research community.

4.5.1 Future Work

Novel approaches to behavior-based identification of actors in any system are being sought in many fields. Much of this work needs to fully define what "behavior" is, to better classify behaviometrics. For smart environments, this research should include more work on sensor selection, placement, and algorithms. Developing new tools using unsupervised learning techniques, such as Deep Belief Networks [28], will mitigate the impact of annotation requirements and lead to more deployable solutions in the future.

Acknowledgments This work was supported by the United States National Institutes of Health Grant R01EB009675 and National Science Foundation Grant CRI-0852172.

References

1. E. D. Mynatt, A.-S. Melenhorst, A. D. Fisk, and W. A. Rogers, Aware technologies for aging in place: understanding user needs and attitudes, Pervasive Computing. 3(2), 36–41 (April, 2004). doi:10.1109/MPRV.2004.1316816.
2. D. Cook, Health monitoring and assistance to support aging in place, Journal of Universal Computer Science. 12(1), 15–29, (2006). doi:10.3217/jucs-012-01-0015.
3. M. Weiser, The computer for the 21^{st} century, Scientific American. pp. 94–104, (1991).
4. P. Rashidi and D. J. Cook, 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. 39(5), 949–959 (September, 2009). doi:10.1109/TSMCA.2009.2025137.
5. V. Libal, B. Ramabhadran, N. Mana, F. Pianesi, P. Chippendale, O. Lanz, and G. Potamianos. Multimodal classification of activities of daily living inside smart homes. In eds. S. Omatu, M. Rocha, J. Bravo, F. Fernández, E. Corchado, A. Bustillo, and J. Corchado, Distributed Computing, Artificial Intelligence, Bioinformatics, Soft Computing, and Ambient Assisted Living, vol. 5518, Lecture Notes in Computer Science, pp. 687–694. Springer Berlin / Heidelberg, (2009). doi:10.1007/978-3-642-02481-8_103.
6. G. Singla, D. J. Cook, and M. Schmitter-Edgecombe, Recognizing independent and joint activities among multiple residents in smart environments, Journal of Ambient Intelligence and Humanized Computing. 1(1), 57–63, (2010). doi:10.1007/s12652-009-0007-1.
7. C. Yiu and S. Singh. Tracking people in indoor environments. In Proceedings of the International Conference on Smart Homes and Health Telematics, ICOST'07, pp. 44–53, Berlin, Heidelberg, (2007). Springer-Verlag. ISBN 978-3-540-73034-7.
8. A. S. Crandall. Behaviometrics for Multiple Residents in a Smart Environment. PhD thesis, Washington State University, (2011).
9. A. S. Crandall and D. J. Cook. Using a Hidden Markov Model for resident identification. In Proceedings of the International Conference on Intelligent Environments, IE '10, (2010).
10. D. J. Cook, A. S. Crandall, G. Singla, and B. Thomas, Detection of social interaction in smart spaces, Cybernetics and Systems: An International Journal. 41(2), 90–104, (2010). doi:10.1080/01969720903584183.
11. A. M. Seelye, M. Schmitter-Edgecombe, D. J. Cook, and A. Crandall, Smart environment prompting technologies for everyday activities in mild cognitive impairment, Journal of the International Neuropsychological Society. (2013). [To appear].
12. H. Wechsler, Reliable Face Recognition Methods: System Design, Implementation and Evaluation. International Series on Biometrics, (Springer, November 2006). ISBN 978-0-387-22372-8.

13. R. T. Collins, R. Gross, and J. Shi. Silhouette-based human identification from body shape and gait. In The Proceedings of the IEEE International Conference on Automatic Face and Gesture Recognition, pp. 366–371 (May, 2002). doi:10.1109/AFGR.2002.1004181.
14. A. K. Jain, A. Ross, and S. Prabhakar, An introduction to biometric recognition, IEEE Transactions on Circuits and Systems for Video Technology. 14(1), 4–20 (January, 2004). doi:10.1109/TCSVT.2003.818349.
15. S. N. Srihari, S.-H. Cha, H. Arora, and S. Lee. Individuality of handwriting: a validation study. In Proceedings of the International Conference on Document Analysis and Recognition, ICDAR '01, pp. 106–109, (2001). doi:10.1109/ICDAR.2001.953764.
16. C. BenAbdelkader, R. Cutler, and L. Davis. View-invariant estimation of height and stride for gait recognition. In eds. M. Tistarelli, J. Bigun, and A. Jain, Biometric Authentication, vol. 2359, Lecture Notes in Computer Science, pp. 155–167. Springer Berlin / Heidelberg, (2006). doi:10.1007/3-540-47917-1_16.
17. R. Moskovitch, C. Feher, A. Messerman, N. Kirschnick, T. Mustafic, A. Camtepe, B. Löhlein, U. Heister, S. Möller, L. Rokach, and Y. Elovici. Identity theft, computers and behavioral biometrics. In Proceedings of the IEEE International Conference on Intelligence and Security Informatics, ISI '09, pp. 155–160, Piscataway, NJ, USA (June, 2009). IEEE Press. doi:10.1109/ISI.2009.5137288.
18. R. V. Rodríguez, R. P. Lewis, J. S. D. Mason, and N. W. D. Evans, Footstep recognition for a smart home environment, International Journal of Smart Home. 2(2), 95–110 (April, 2008).
19. V. Menon, B. Jayaraman, and V. Govindaraju. Biometrics driven smart environments: Abstract framework and evaluation. In eds. F. Sandnes, Y. Zhang, C. Rong, L. Yang, and J. Ma, Ubiquitous Intelligence and Computing, vol. 5061, Lecture Notes in Computer Science, pp. 75–89. Springer Berlin / Heidelberg, (2010). doi:10.1007/978-3-540-69293-5_8.
20. A. S. Crandall and D. J. Cook. Learning activity models for multiple agents in a smart space. In eds. H. Nakashima, H. Aghajan, and J. C. Augusto, Handbook of Ambient Intelligence and Smart Environments, pp. 751–769. Springer US, (2010). doi:10.1007/978-0-387-93808-0_28.
21. W. CASAS. CASAS public smart home datasets repository. http://casas.wsu.edu/datasets.html
22. E. M. Tapia, S. S. Intille, and K. Larson. Activity recognition in the home using simple and ubiquitous sensors. In eds. A. Ferscha and F. Mattern, Pervasive Computing, vol. 3001, Lecture Notes in Computer Science, pp. 158–175, Berlin / Heidelberg, (2004). Springer-Verlag. doi:10.1007/978-3-540-24646-6_10.
23. T. van Kasteren and B. Krose. Bayesian activity recognition in residence for elders. In The IET International Conference on Intelligent Environments, IE '07, pp. 209–212 (September, 2007). ISBN 978-0-86341-853-2.
24. A. Viterbi, Error bounds for convolutional codes and an asymptotically optimum decoding algorithm, IEEE Transactions on Information Theory. 13(2), 260–269 (April, 1967). doi:10.1109/TIT.1967.1054010.
25. S. Z. Li and A. K. Jain, Eds., Handbook of Face Recognition. (Springer, March 2005), $1st$ edition. ISBN 978-0-38740-595-7.
26. V. Srinivasan, J. Stankovic, and K. Whitehouse. Using height sensors for biometric identification in multi-resident homes. In eds. P. Floréen, A. Krüger, and M. Spasojevic, Pervasive Computing, vol. 6030, Lecture Notes in Computer Science, pp. 337–354. Springer Berlin / Heidelberg, (2010). doi:10.1007/978-3-642-12654-3_20.
27. G. Fritz, V. Beroulle, M. Nguyen, O. Aktouf, and I. Parissis. Read-Error-Rate evaluation for RFID system on-line testing. In Proceedings of the IEEE International Mixed-Signals, Sensors and Systems Test Workshop, IMS3TW '10, pp. 1–6 (June, 2010). doi:10.1109/IMS3TW.2010.5503016.
28. G. E. Hinton. Learning multiple layers of representation. In Trends in Cognitive Sciences, vol. 11, pp. 428–434, (2007).