Virtual Stochastic Sensors for Ambient Assisted Living - Analyzing the Effect of Generalized Resident Behavior

Vishwajeet Karumuri[1], Claudia Krull[1*]

[1]Fakultät für Informatik, Otto-von-Guericke-Universität, Postfach 4120
39016 Magdeburg, Germany; *claudia.krull@ovgu.de

Abstract. The advancements in Ambient Assisted Living (AAL) have been prompted by the growing population of elderly individuals facing diagnoses such as Dementia or Alzheimer's, aiming to enhance their overall quality of life. To provide support it is important to know their daily activities and support them. A large portion of research in the field of Human Activity Recognition uses black box learning approaches such as deep learning, but there are cases where model based methods, such as Virtual Stochastic Sensors (VSSs) are competitive. This is possible because the model based methods can include system structure in the modeling process if it is known. VSS's are derived from Hidden Markov Models (HMM) and applied to a CASAS single resident dataset, which is an apartment fitted with different types of ambient sensors. For future applications a generalization of behavior, sensors or models is necessary so that models are not just trained and used for one specific apartment and setup. In this paper we analyze the effect of generalizing the residents behavior on the reconstruction accuracy. The generalization did lead to some improvements in the reconstruction accuracy, but the implications for the actual application need to be considered.

Introduction

Advancements in the medical field led to long and healthier lives, roughly around 20% of the world population will be aged above 60 by 2050 [1], seeking to explore effective solutions that empower elderly individuals to maintain independent living. Studies of Counsel and Care in UK showed that elderly people have a preference to stay in their apartments rather than nursing homes [2]. Researchers have shown that having clinical therapy at home has no negative effect on the process [3]. There are multiple ways to make this happen, one way is Ambient Assisted Living, where some ambient sensors are installed, e.g. motion sensors, to monitor the behavior of elderly residents, a model is used to guess the behavior using the sensor readings, then, this is used to identify if everything aligns with the usual behavior, and if not, assistance can be provided. This ensures a safer living space without unduly intruding on the privacy of the residents.

For replicating human behavior [4], [5] and [6] successfully implemented machine and deep learning algorithms for this task. In [7] Virtual Stochastic Sensors (VSSs) are used, which are designed to facilitate the reconstruction of partially observable stochastic systems and enable solving backward problems in the realm of stochastic modeling and simulation. The model is based on the ideas of Hidden Markovian Models (HMM) but extends these by arbitrary non-Markovian distribution functions for multiple concurrent processes and symbol outputs at arbitrary points in time [8]. VSS discretize the time domain and use a simple iterative algorithm to discover the reachable state space of the model, therefore being very flexible. However, they cannot be applied in real-world scenarios on a large scale yet, because model parametrization is not automated, and the model needs to be trained for a specific system to be used for reconstruction. The model generalization tested in this paper is one step towards pre-trained models for unknown systems.

This research aims to develop a conceptual model for a generalized set of activities, transform the dataset (CASAS Single resident apartment (HH101)) and feed the transformed data into the model to reconstruct human behavior, and later evaluate the results using different metrics. This dataset contains activities performed by the residents and the corresponding active sensors for the activities performed. The rest of the paper is structured as follows, Section 1 contains the details of related research, Section 2 underlines the details of the dataset, Section 3 explains the conceptual model and the algorithm design, Section 4 shows the outcome of the research.

1 Related Work

This section explores different approaches for reconstructing human behavior in the field of ambient assisted living that are similar in approach or goal to this research. Additionally, it presents relevant information for Virtual Stochastic Sensors (VSSs).

[9] has mainly emphasized the duration of the activity to find the abnormality in human behavior using Explicit State Duration Hidden Markov Model (ESD-HMM). They checked the deformity of current activity which might be shorter or longer than the usual routine, the dataset used for this research was limited to the kitchen. [10] introduced a new observation probabilistic model to recognize daily activities, they incorporated temporal data which had information regarding 77 sensors. [4] tried to identify the critical features from the sensor data, these features are used to classify overlapped activities. Unsupervised K nearest neighbor (KNN) was applied to day-to-day activities but with a small set of activities. [5] implemented long short-term memory (LSTM) recurrent neural network (RNN) to perform activity recognition from wearable sensors, this implementation was not tested for activities of daily living. [11] suggested another approach using RNN on three different datasets, this approach outperformed similar approaches concerning accuracy and speed, but the dataset is not publicly available. [6] proposed an unobtrusive activity recognition classifier using deep convolutional neural network (DCNN) and publicly available CASAS Aruba dataset. [12] used a knowledge-driven approach, including a Partially Observable Markov Decision Process (POMDP) and exploited the task information, while the location is combined with the sensor events in the smart home, but the series of conditions are used to classify activity. This shows that AAL is a current research field with several approaches all with their individual features.

1.1 Virtual Stochastic Sensors

Virtual Stochastic Sensors represent a framework for analyzing partially observable stochastic systems, including different modeling paradigms and solution methods [13]. VSS can compete with some black box models when the hidden system structure information is available [7], and can incorporate such information to accurately represent dynamic system behavior and its relationship with the observable output. VSS use augmented stochastic Petri nets (ASPN) as user models that contain multiple concurrent non-Markovian transitions [14]. ASPNs generate observable output by the firing of transitions depending on the discrete system state, the discrete system output is collected in a protocol with associated time stamps, since in contrast to the Hidden Markovian Model (HMM), the model is defined in continuous time and can produce output at arbitrary points in time [7].

[7] has applied VSS on CASAS Aruba 2010 dataset and produced a very promising result, based on this VSS is considered to be a viable option for activity classification. However, the previous implementations were all trained and tested on the same use case, which is not a feasible approach for broad scale applicability. Therefore we are examining different methods of generalization to eventually enable generalized models to be applied on systems not used for the training. This paper is focusing on the impact of generalization of the activity set on the reconstruction accuracy for a single system.

2 CASAS Dataset and VSS Model

In this paper we are using a data set from the CASAS Research Project of Washington State University [15]. There are different types of datasets, one contains the daily activities of 20 participants, few other datasets include pets for single or multiple residents, and finally, HH datasets are mostly single residents but a small portion of them are two-resident apartments. In this research, the HH101 single resident apartment dataset is considered because it is multivariate, sequential, and time series. The data is collected using different kinds of sensors, like motion, door/temperature, and light switch sensors placed throughout the apartment, while the residents perform their normal routines. The dataset format is Date, Time, Sensor, Translate, and Activity (Table 1).

- Date is when the information is recorded in MM/DD/YYYY
- Time is in the 24-hour format
- Sensor is the name of the sensor

Date	Time	Sensor	Translate	Activity
08/01/2012	00:00:06	M008	LivingRoom	Watch_TV
08/01/2012	00:00:07	M008	LivingRoom	Watch_TV
08/01/2012	00:00:09	M008	LivingRoom	Watch_TV
		....		
08/01/2012	01:12:51	M012	Bedroom	Sleep
08/01/2012	01:12:54	M012	Bedroom	Sleep
08/01/2012	01:12:55	M012	Bedroom	Sleep
		....		

Table 1: Sample extract from single resident HH101 dataset

- Translate is room-level location
- Activity is what the person is doing at that specific date and time, and was tagged by the residents

The time frame of this dataset is from 20th July 2012 to 17th September 2012. From this time frame, only the data of September 3rd to 9th is used here.

2.1 Generalization of Activities

In this dataset a total of 34 activities were recorded (Table 2), describing the daily routine of a person living in this residence. For generalization, we grouped activities with similar characteristics and similar sensor outputs. These 34 activities are first generalized to 11 activities and further generalized to 6 activities. The smaller set of activities was created, to test the effect of different levels of generalization.

Before grouping similar activities, some were removed from the dataset. *Leave_Home*, *Enter_Home*, all medicine activities have very short durations are therefore not easily detectable through the sensor readings. The activities *Work_At_Table* and *Entertain_Guests* occurred very rarely and and were therefore ignored in the analysis. Some activities like *Go_To_Sleep*, *Wake_Up*, *Eat* also have very low occurrence, but can be combined with *Sleep*, other *Eat_* activities respectively, and will therefore not be omitted.

For the first type of grouping, activities like *Cook_Breakfast*, *Cook_Lunch*, *Cook_Dinner*, and *Cook* are grouped to *Cook*, even though these activities occur during different times of the day they essentially give the same idea. Similar to cook, we can combine all eat, wash dishes and sleep activities into *Eat*, *Wash_Dishes* and *Sleep* respectively. *Bed_Toilet_Transition* takes place for very little time and occurs during the *Sleep* activity hence it is grouped with sleep. *Relax* and

Activity	Occurences %
Watch_TV	26.59%
Sleep_Out_Of_Bed	7.21%
Bathe	7.06%
Cook_Breakfast	7.06%
Dress	6.09%
Toilet	5.95%
Personal_Hygiene	5.73%
Sleep	5.10%
Read	3.73%
Relax	2.51%
Cook_Dinner	2.41%
Drink	1.65%
Eat_Breakfast	1.55%
Morning_Meds	1.54%
Evening_Meds	1.46%
Wash_Breakfast_Dishes	1.39%
Cook_Lunch	1.39%
Wash_Dishes	1.39%
Leave_Home	1.35%
Cook	1.20%
Enter_Home	1.12%
Entertain_Guests	1.11%
Wash_Dinner_Dishes	1.07%
Phone	0.80%
Groom	0.78%
Step_Out	0.65%
Eat_Dinner	0.47%
Eat_Lunch	0.38%
Wash_Lunch_Dishes	0.34%
Bed_Toilet_Transition	0.31%
Eat	0.22%
Go_To_Sleep	0.18%
Wake_Up	0.16%
Work_At_Table	0.08%

Table 2: All 34 activities and their total share in occurence time the CASAS HH101 dataset

Sleep_Out_Of_Bed are combined to *Relax* because both activities are done in similar places and during similar times of the day, in similar sense *Personal_Hygiene* and *Bathe* are combined to *Freshen_Up*. This various grouping leads to 11 activities. The first grouping is shown in table 3, where the groups are separated by dashed and solid lines.

To obtain the smaller set, the activities are grouped in such a way that the exact information is not known but the general idea of the activities is not lost. *Freshen_Up* and *Toilet* are combined into *Freshen_Up*, activities like *read*, *Phone*, *Relax*, *Watch_TV* into *Personal_Activity*, as well as *Cook* and *Wash_Dishes* into *Meal_Prep*. This results in a total of 6 activities. This second grouping is shown in table 3, where the groups are separated by solid lines.

2.2 Data preparation

After cleaning the data, the next step is to modify the data into distributions and probabilities so that it can be used as input in the model. Each activity has breaks, short and long breaks for one type of Petri net and morning, evening and night breaks for another type of Petri net.

Distributions for all the breaks and activities are based on their duration in minutes. Distributions are estimated with the help of the MATLAB distribution fitter app. Probabilities for an activity to go to breaks are needed to be calculated. As mentioned in Section 3 if a break is less than or equal to 60 minutes it is considered a short break and anything longer is a long break. Depending on this probability for an activity to go to short or long break is obtained.

Once all the distributions and state transition probabilities are determined, probabilities for output symbols are calculated. This is the final input required to run the model, and this is achieved with Equation (1).

$$P(S_i|A_i) = \frac{(\Delta t|\forall S = S_i \cap A = A_i)}{(\Delta t|\forall A = A_i)} \quad (1)$$

- S_i the sensor
- A_i the activity
- $P(S_i|A_i)$ is the probability of sensor given activity

For evaluation, an unlabeled trace, with an existing ground truth, is required this contains the information regarding the time and the active sensor. This trace is created for a day, from data which is not included for distribution and probability calculation.

34 Activities	11 Activities	6 Activities
Watch_TV	Watch_TV	Personal_Activity
Sleep_Out_Of_Bed	Relax	
Relax		
Phone	Phone	
Read	Read	
Bathe	Freshen_Up	Freshen_Up
Personal_Hygiene		
Toilet	Toilet	
Dress	Dress	Dress
Groom		
Cook_Breakfast	Cook	Meal_Prep
Cook_Dinner		
Cook_Lunch		
Cook		
Wash_Dinner_Dishes	Wash_Dishes	
Wash_Lunch_Dishes		
Wash_Breakfast_Dishes		
Wash_Dishes		
Eat	Eat	Eat
Eat_Breakfast		
Eat_Dinner		
Eat_Lunch		
Drink		
Sleep	Sleep	Sleep
Bed_Toilet_Transition		
Go_To_Sleep		
Wake_Up		
Morning_Meds	–	
Evening_Meds	–	
Leave_Home	–	
Enter_Home	–	
Entertain_Guests	–	
Step_Out	–	
Work_At_Table	–	

Table 3: Activity grouping, separated by dashed lines for first grouping or solid lines for second grouping

3 Conceptual Model and Algorithm Design

3.1 Conceptual Model

There are different ways to model the daily activities of a resident. In this research, each activity has a dedicated Augmented Stochastic Petri net (ASPN) because these are simple and demand less computational power [7]. So the activities are grouped into sets, one set has 2-breaks (Figure 1) design, with a long and a short break, and the other set has 3-breaks design (Figure 2) with short, medium and long breaks. Activities *Eat*, *Cook* and *Wash_Dishes* have 3-breaks design, the remaining activities all have 2-breaks design. Since there is an extra place for 3-breaks it also has extra transitions. Both standard Petri nets have *Activity* place and *No_Activity* place. These individual Petri nets are independent of each other.

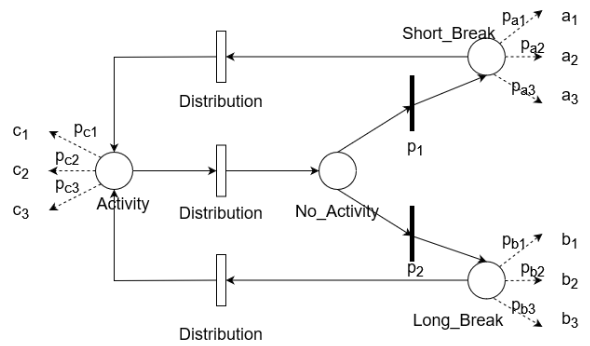

Figure 1: 2-Breaks Augmented Stochastic Petri Net

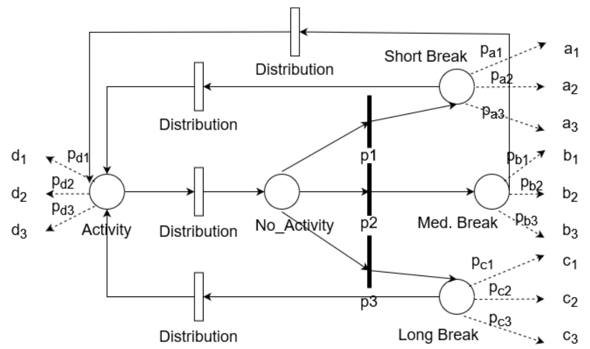

Figure 2: 3-Breaks Augmented Stochastic Petri Net

Once the structure of the Petri net is finalized, the output symbol emissions for the behavior reconstruction algorithm are added. These output symbols are linked with the Petri net places [8], as they occur, when an activity is being performed rather then when a state change occurs. From Figure 1 and 2 the output symbols are connected to all places except *No_Activity*. The p_k denote transition probabilities, p_{ak} denotes the output probability of symbol a_k. These output symbols play a crucial role in Virtual Stochastic Sensors (VSS), as they connect an observed symbol to the unobserved system state, and thus enable a behavior reconstruction.

3.2 Algorithm Design

To reconstruct the unobserved system behavior, here the residents activities, the Proxel algorithm is used. The Proxel algorithm determines possible development paths of the system and their probability [16, 17, 8]. A Proxel is a 5-tuple, which represents one point in the expanded system state space, this tuple consists of the state of a system, age intensity, current point in simulation time, route through the state space and probability. All individual models for activities are executed independently and determine output paths for all activities. This output path contains the probability of an activity occurring at a certain point in time.

Once the probabilities for each activity are computed the next thing is to classify the activities. For classification a simple decision system is incorporated. This system outputs the activity which has the highest probability for all individual models for their activity state at that point in time. For every timestep of the protocol, this decision system results an activity. This type of system works because all the activities have individual ASPNs independent of each other. If for a particular timestep the probability of all activities is zero then the model returns *Other_Activity*. Details of the solution method can be found in [7]. This procedure was employed for the two reduced activity setups and the results will be shown and discussed in the following section.

4 Experiments and Results

In this section, the performance of the models is evaluated. The metrics precision, recall and F1 score are calculated for individual activities, and then they are averaged in two different ways one by averaging for all the activities (*Average*) and the other by adding weights depending on the activity occurrences (*Weighted Average*). The average recall is also often used as an overall accuracy measure. These results are correctly classified if at a given time the reconstructed activity corresponds to the trace's ground truth. The above-discussed metrics are evaluated for all days of the week (03rd September to 9th September), to make sure that the model is not biased for a few specific days of the week. A confusion matrix is also created to get a comprehensive breakdown of the model's behavior, allowing for a detailed assessment of its performance.

All the metrics discussed above are applied for two different sets of activities, one set has 11 and the other set has 6. The experiments are preformed in a k-fold cross-validation fashion, using n-1 days for training and the one left out for testing. For both sets the range of results for F1 and Accuracy are given, and an example result is examined in detail.

4.1 Evaluation metrics with 11 Activities

The model is applied for the larger set of activities for a week using k-fold cross validation. One of the results, which was obtained on Tuesday (4th of September) of that week was chosen to be investigated in detail here.

Activity	Precision	Recall	F1 Score
Watch_TV	0.51	0.16	0.24
Dress		0.00	
Freshen_Up	0.45	1.00	0.62
Sleep	0.96	0.96	0.96
Read	0.00		
Relax		0.00	
Toilet		0.00	
Phone	0.00		
Cook	0.60	0.98	0.75
Eat		0.00	
Wash_Dishes		0.00	
Average	0.42	0.34	0.64
Weighted_Average	0.61	0.49	0.50

Table 4: Individual activity and average model performance metrics for Tuesday 4th of September, larger activity set

Table 4 contains the precision, recall and F1 measures for the same day, some values are not filled, as they cannot be calculated if there are no classifications in that specific class. Only few of the 11 generalized activities are classified well, such as *Sleep* or *Cook*, many of the less frequent activities are grossly mis-classified, such as *Relax* or *Dress*. One can see, that for precision and recall, the weighted average is much better than the average, since the weighted average is impacted by the activities with longer durations, which were classified more accurately, such as *Sleep*.

On the other hand, the model is having trouble classifying some shorter activities such as *Relax* and *Eat*. The reason is that the model is unable to distinguish between some activities because the activities take place during similar times and activate similar sensors.

Table 5 shows the minimum and maximum results from this experiment for average and weighted average recall (also accuracy) as well as F1 measure. One

Evaluation Metric	Min	Max
Average_Recall	0.27	0.44
Weighted_Average_Recall	0.47	0.62
Average_F1	0.42	0.67
Weighted_Average_F1	0.46	0.57

Table 5: Range for average model performance metrics for Tuesday tth of September, larger activity set

can see, that there is quite a large range, which shows that for further usage, more training data has to be used to make the model performance more reliable and predictable. Furthermore, the average and weighted average recall differ quite considerably, this points to the same problem shown in the single day evaluation, that activities with longer durations are classified well, but shorter activities with overall less time are not reconstructed well.

Another experiment was conducted for a set of 6 activities. The grouping procedure was explained in 2.1.

4.2 Evaluation metrics with 6 Activities

Analogous to the experiment with 11 activities, the model is applied for the smaller set of 6 activities for a week using k-fold cross validation. One of the results, which was obtained on Monday (3rd of September) of that week was chosen to be investigated in detail here.

Activity	Precision	Recall	F1 Score
Personal_Activity	0.95	0.81	0.88
Dress		0.00	
Freshen_Up	0.96	1.00	0.98
Sleep	0.89	0.90	0.90
Eat		0.00	
Meal_Prep	0.26	1.00	0.41
Average	0.77	0.62	0.79
Weighted_Average	0.90	0.85	0.86

Table 6: Individual activity and average model performance metrics for Monday 3rd of September, smaller activity set

The results for the smaller group of activities are better.

Table 6 contains the precision, recall and F1 measures for the same day, some values are not filled, as they cannot be calculated. Four of the six generalized activities are classified well, only two activities, *Dress* and *Eat*, are not classified at all, reducing their recall to

0. As in the larger activity set, for precision and recall, the weighted average is much better than the average, since the mis-classified activities are rather short and impact the weighted average less. This second experiment shows that the further generalization, combining activities with similar sensor footprint, has improved the reconstruction performance.

Evaluation Metric	Min	Max
Average_Recall	0.55	0.63
Weighted_Average_Recall	0.66	0.90
Average_F1	0.56	0.89
Weighted_Average_F1	0.68	0.89

Table 7: Range for average model performance metrics for Monday 3rd of September, smaller activity set

Table 5 shows the minimum and maximum results from this experiment for average and weighted average recall (also accuracy) as well as F1 measure. As in the previous experiment, there is quite a large range. Therefore, more training data has to be used to make the model performance more reliable and predictable.

Furthermore, the average and weighted average differ quite considerably, both for recall as well as F1 measure, for the same reason as in the larger data set. However, the overall performance for the whole week is better than for the larger set.

4.3 Experiment Discussion

The two experiments show that generalizing activities has an impact on the overall reconstruction performance of Virtual Stochastic Sensors. Leaving aside the shortcomings of the experiments conducted one can conclude the following: A semantic grouping of activities by similar time of day and living area, which corresponds to a similar sensor footprint, leads to an improvement in accuracy. However, the information content of the reconstruction is considerably less, when more activities are combined under the same label. It has to be investigated with the help of domain experts at which point the generalized set of activities with better performance measures still holds enough information to assess the residents status. Ultimately the goal is to use this ambient sensor observation to decide, whether the residents behavior is still within normal bounds, or whether outside assistance or intervention is necessary.

5 Conclusion

In conclusion, this research has a few issues which need to be overcome, but even with these issues, there are a few positive takes. This research used the a CASAS dataset for single-resident apartment, there were various data transformations for activities and sensors. Then probabilities and distributions were extracted to input into the model. As for the model, Virtual Stochastic Sensors (VSS) are used to reconstruct resident activities. Two experiments were carried out to evaluate the model behavior, one had 11 activities and the other had 6 activities. Experiments with a smaller set of activities resulted in better performance Whether the better reconstruction accuracy comes at the expense of less information on the residents behavior, has to be investigated.

5.1 Future Work

The analysis presented in this paper is only part of the research aiming at providing generalized models to be pre-trained and then applied for the reconstruction of previously unknown apartments, or systems in general.

Further steps in this direction include expanding the research to different model architectures, for example, creating Augmented Stochastic Petri nets (ASPNs) for room-level, or extend this concept to generalize multiple single-resident apartments. This implies applying data analysis and models to a broader context, like adapting algorithms for variations in apartment layouts, sensor placements and resident behaviors.

References

[1] Rashidi P, Mihailidis A. A survey on ambient-assisted living tools for older adults. *IEEE journal of biomedical and health informatics*;17(3):579–590.

[2] Sun H, Florio VD, Gui N, Blondia C. Promises and Challenges of Ambient Assisted Living Systems. In: *2009 Sixth International Conference on Information Technology: New Generations*; pp. 1201–1207.

[3] Deutsch JE, Lewis JA, Burdea G. Technical and patient performance using a virtual reality-integrated telerehabilitation system: preliminary finding. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*;15(1):30–35.

[4] Fahad LG, Tahir SF, Rajarajan M. Feature selection and data balancing for activity recognition in smart homes.

In: *2015 IEEE International Conference on Communications (ICC)*. IEEE; pp. 512–517.

[5] Ordóñez FJ, Roggen D. Deep convolutional and lstm recurrent neural networks for multimodal wearable activity recognition. *Sensors*;16(1):115.

[6] Gochoo M, Tan TH, Liu SH, Jean FR, Alnajjar FS, Huang SC. Unobtrusive activity recognition of elderly people living alone using anonymous binary sensors and DCNN. *IEEE journal of biomedical and health informatics*;23(2):693–702.

[7] Mueller LF. Feasibility and Applicability of Virtual Stochastic Sensors for Human Activity Recognition in the Context of Ambient Assisted Living. Master's thesis, Otto-von-Guericke-University Magdeburg. 2021.

[8] Krull C. Virtual Stochastic Sensors: Formal Background and Example Applications Reconstructing the Behavior of Partially Observable Discrete and Hybrid Stochastic Systems. Habilitation thesis, Otto-von-Guericke-University Magdeburg. 2021.

[9] Lühr S, Venkatesh S, West G, Bui HH. Explicit state duration HMM for abnormality detection in sequences of human activity. In: *PRICAI 2004: Trends in Artificial Intelligence: 8th Pacific Rim International Conference on Artificial Intelligence, Auckland, New Zealand, August 9-13, 2004. Proceedings 8*. Springer; pp. 983–984.

[10] van Kasteren T, Kroese B. Bayesian Activity Recognition in Residence for Elders. In: *3rd IET International Conference on Intelligent Environments*. 2007; .

[11] Singh D, Merdivan E, Psychoula I, Kropf J, Hanke S, Geist M, Holzinger A. Human activity recognition using recurrent neural networks. In: *Machine Learning and Knowledge Extraction: First IFIP TC 5, WG 8.4, 8.9, 12.9 International Cross-Domain Conference, CD-MAKE 2017, Reggio, Italy, August 29–September 1, 2017, Proceedings 1*. Springer; pp. 267–274.

[12] Chen L, Nugent CD, Wang H. A knowledge-driven approach to activity recognition in smart homes. *IEEE Transactions on Knowledge and Data Engineering*; 24(6):961–974.

[13] Krull C, Horton G. Hidden Non-Markovian Models: Formalization and Solution Approaches. In: *Proceedings of 6th Vienna Conference on Mathematical Modelling, Vienna, Austria*. 2009; pp. 682–693.

[14] Krull C. A Hybrid User Model for Virtual Stochastic Sensors. *Simulation Notes Europe SNE33*;33(1):35–43.

[15] Cook DJ, Crandall AS, Thomas BL, Krishnan NC. CASAS: A smart home in a box. *IEEE Computer*. 2013;46(6):26–33.

[16] Horton G. A New Paradigm for the Numerical Simulation of Stochastic Petri Nets with General Firing Times. In: *Proceedings of the European Simulation Symposium 2002*. SCS European Publishing House. 2002; pp. 129–136.

[17] Lazarova-Molnar S. The Proxel-Based Method: Formalisation, Analysis and Applications. Ph.D. thesis, Otto-von-Guericke-University Magdeburg. 2005.