University of Stavanger

Faculty of Science and Technology

MASTER'S THESIS

Study program/ Specialization: Computer Science	Spring semester, 2012 Open
Writer: Leila Chinaveh	…………………………………………… Leila Chinaveh
Faculty supervisor: Professor Chumming Rong External supervisor(s):	
Title of thesis: Analytic Methods for Human Activities at Home	
Credits (ECTS):	
Key words: Anomaly behavior, Rule engine, Jess, False alarm	Pages: 55 + enclosure: CD Stavanger, 15.06.2012 Date/year

Analytic Methods for Human Activities at Home

Leila Chinaveh
University of Stavanger,
l.chinaveh@stud.uis.no

June 2012

Dedication

This thesis is dedicated to my parents, husband and daughter.

Acknowledgements

I would like to express my sincere gratitude to Professor Chumming Rong and Dr. Tomasz Wiktor Wlodarczyk for their support, patience, motivation and immense knowledge. I could not finish my thesis without their supports.

Contents

List of Figures

List of Tables

Abstract

The population of elderly people increases dramatically. It means that the large amount of old people need care and cure in the future. Obviously providing the health services and the security for them needs many human resources. Preparing the high standard of life and the services also have the financial issues. Thus there is a high motivation for many organizations and companies to find the efficient solutions. One of the considerations about this problem is the design of the smart home for the elderly. The system should have ability to collect the data and make a significant decision for the different situations. Regarding the requirements of the old person, the system should be comfortable, secure and reliable.
The smart home should be powerful system to support the different topics which are related to the elderly life. One of them is the ability to analyse the activity in the home.
In this thesis, I have described an application which used an algorithm for detecting the behaviour in the home. The process is based on using the motion sensors and the duration of time. By receiving the data from different places, the application recognizes the mode of activity in the house. The process has ability to find the anomalous behaviour from the patient. The anomaly behaviour are categorized with the different levels of the emphasis.

1 Introduction

1.1 Motivation

An ageing world report[3] shows that the number of elderly (65+) in the next 30 years becomes almost double. The population of ageing reaches to the 1.3 billion. It means that the increasing average of old people in the world is around 870,000 in each month. After the second world war and improvements in the health care, the death rate of older ages is going down.
This changing has a huge implications for social and health services. So there is high motivation to find the acceptable and adaptive solutions for that changing. In addition social services and the responsible organizations should keep the high standard of life and services for the elderly.
Aging-in-place(AIP)[4] is one of the consideration for them. The goals of AIP are providing and improving the quality of life according the senior situations in their residence. The quality of life includes different items like person, home, care, financial and etc. Also the plan should have capability to maintain with changing the situations during the time.
"Smart System to support Safer Independent Living and Social Interaction for Elderly at Home"[5] is a project to produce a system to use in a smart home which is followed by the university of Stavanger (UIS). This smart home is specified for the aging. So the system should

support many exceptions and conditions for each specific person. The Reliability, flexibility and quality are the important factors in the system. Although the seniors should be comfortable and have their privacy in the home.
In the different research and studies are considered to find an optimized algorithm to reach an efficient result to support the system in the smart home of elderly. The smart home which is specified for the aging people includes different discussions. The analysis of human activities at home is one of them. The aims of using a system with ability to analyse the activity in the house are:

- Ability to check the activity of person in the house with out connecting directly.
- Preventing the risky situations.
- Providing a safe and secure feeling for the seniors in the home.

This information motivated me to work about this topic. I try to create a new algorithm and technique to collect and to integrate the data, then the data are used to detect the activity in the home. The process gives the result by evaluating the activity.

1.2 Goals

The goals of the thesis are:

- Organizing the data
- Making the data modeling
- Using the rules to analyse the data
- Providing flexibility in the application to recognize the different situations
- Preparing the communications between the process and the user in the input and the output.
- Testing the application with the different resources
- Checking the result with the expectations and trying to reduce the failures.

The structure of this thesis is as follows:

- Section 2: I represent the related works to this subject and explain the different techniques and their aims.
- Section 3: I present the definition of the rule engine, the Jess rule engine and the benefits of using the rule engine.
- Section 4: I explain the problem definition and reasons to use the behaviour modeling.
- Section 5: I present the explanation of the algorithm and the steps of the procedure.

- Section 6: I represent the reasons to choose the different levels of alarm and the techniques to reduce the false alarm.
- Section 7: I discuss the challenging to find the motion sensor and the working on it to produce the suitable data from that.
- Section 8: I present the conclusion and discussion about the result of working and the future work.

During the work, I obtained the following items:

- Ability to organize data
- Provide a Behavior modeling based on the process
- Using the capability of rules during the profile
- Reducing the false alarm by checking the result
- Process has flexibility to give the expected results with different resources (Generated data and the real data from sensor)

2 Related work

I studied different techniques and systems with related subjects to my idea. I should mention I began to search about this project from the last semester and I made a report about the related works in the computer project course. I used some parts of the report here.
Some researches and studies have a common part to implement their techniques. Recognition of activity [6] [7] [8] [1] is one of the important part which is used in the most systems. It means that the life of patient is analysed in a period of time and made pattern for that. It is called the activity of daily living (ADL). Indeed the ADL is important because the system should have a pattern to use as a reference of activity and the pattern is different from one person to another person. ADL is approached by the different algorithms and the techniques in their works. After finding the ADL, they focused on the two terms, Recognition of the anomaly behavior and/or the prediction of action for the person in the home. They used different kinds of sensor and techniques to reach their purposes. Their most methods are included, using data mining, clustering and the combination of rules. The smart home for dementia and old people [6] is considered to find the anomaly behavior and the prediction of action in the smart house. They used the motion sensor in the home. They implemented their algorithm with creating the plots to visualise the activity in the house by using the start time and the duration of time for each activity (figure 1[6]). The plots give useful information and good basis to classify the activities and to recognize the normal behavior from the anomaly. They use this benefit for the

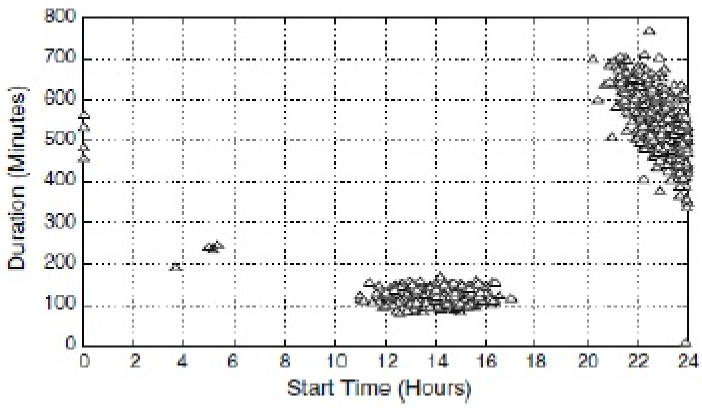

Figure 1: The plot of collecting data from bedroom

patient who has abnormal behavior and they can figure out the improvement of the health with new medication and decrease the number of abnormal behavior from the plots. They used the data clustering and the Echo State Network (ESN) to design the algorithm in their work.
To detect the anomaly from the normal behavior, they identified three categories:

- Beginning to cluster the normal data, when data can not fit in to the any clusters, it can be expected as the anomaly data. In this data clustering, any kind of clustering techniques can be used.
- They cluster the data then, if they found any data which is far away from the cluster, it is the anomaly data.
- They cluster the data, then they consider on the size of the clusters. If they have the small clusters or the spars ones. They contain the anomalies data. To find the small cluster they used a threshold value to compare the clusters.

The techniques that they used to clustering the data in this paper are Self Organizing Map (SOM), K-means clustering, Fuzzy c-means clustering. Self Organizing[9] Map is a kind of artificial neural network. It produces a low dimensional (2D). It has two mode training and mapping. K-means clustering[10] is easy way to classify of the data. It is choose k as a centroid of the each cluster and for the each point it should find nearest distance centroid. When it finished all points, it gives a new k to them and continues until the centroid cannot move any more. Fuzzy c-means clustering[**?**], in this clustering each data can belongs to the two or more clusters and it is useful to pattern recognition.
To predict of action, they collected the data from the sensors and combined them with each other which means the signal in each level belongs to the one sensor it is useful to make a prediction of activities. "Figure 2"[6] gives an example of the signals. They had considered on the different

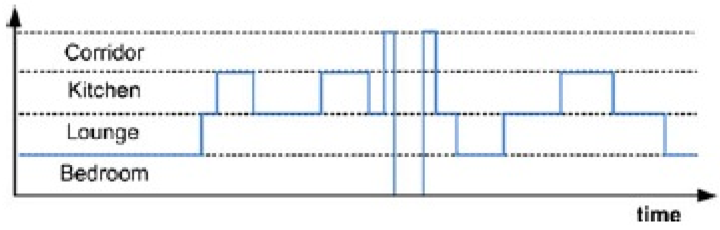

Figure 2: Activity in the different places of the home

techniques for the prediction of the stationary time series, and it included: Echo State Network (ESN), Back Propagation Through Time (BPTT) and Real Time Recurrent Learning (RTRL). They mentioned that ESN is suitable technique for their prediction problems. They have tested the application in different location. The bedroom has good results with the matching of the actual sensor and predicted values(Figure 3[6]). But in some locations which have the more

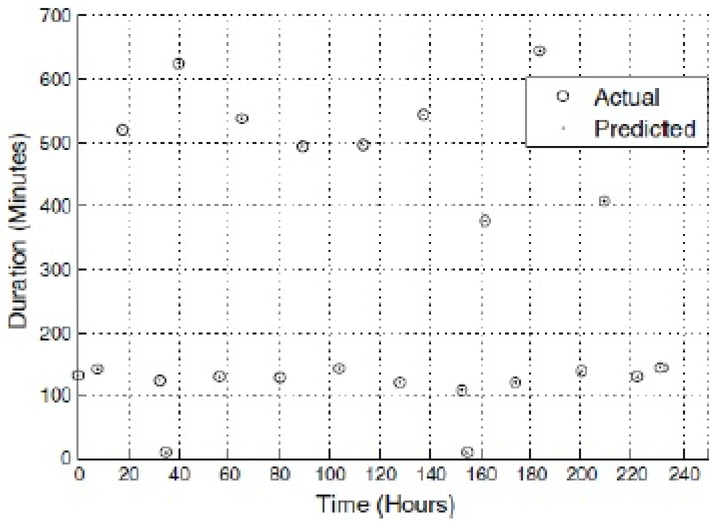

Figure 3: The prediction of action in the bedroom

chaotic signals, the results are not exactly matching ("Figure 4"[6]). Another factor which is important in the performance of ESN is the different size of the reservoir. ESN has the three layers which include input, hidden and output. Input and hidden layer connected to each other, hidden connect to the output and the output is backward connected to the hidden. ESN have the reservoir of the conventional processing of the elements. They are interconnected with the untrained of the random weights and the output layer. For training an ESN, they had an input $u(n)$, a reservoir state $x(n)$ with the M processing of the elements and an output $y(n)$. The "formula" is calculated in below which is claimed in the paper[6]:

$$x(n+1) = tansing(w_x.x(n) + w_{in}.u(n) + v(n+1))$$

and

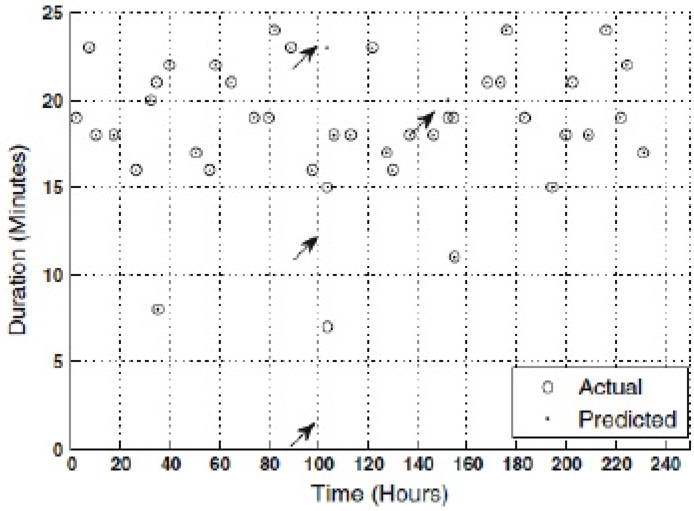

Figure 4: The prediction of action in the corridor

$$y(n) = w_x.x(n)$$

The $v(n+1)$ is shown the optimal noise vector. The w_x, w_{in} and w give the internal connection weight of the reservoir. The tansig is a hyperbolic tangent sigmoid function. It makes the element wise.

The support vector machines (SVM) is another technique with using the multi classification of ADL [7], they designed the algorithm with the classifying of the data to find a temporary frame as a part of the ADL. They also focused on the finding of the abnormal behavior during the activity at home. The seven activities are chosen from ADL and AGGIR. The activities are,Sleeping, Preparing and having a breakfast, dressing and undressing, resting, taking shower or go to the toilet, they use the position of door to realize the different using of the toilet (complete close door represents to use the toilet and partly close used to take a shower). They collected the data for the each individual separately. First they gave an introduction of SVM which is based on the classification of the data with the labels of -1 and 1 and the using of the "mathematical formula". It shows in below which is claimed in [7].

$$\begin{cases} \sum\limits_{\alpha_k>0} f(\alpha_k \langle x, x_k \langle + w_0) > 0 \Rightarrow f = 1 \\ \sum\limits_{\alpha_k>0} f(\alpha_k \langle x, x_k \langle + w_0) < 0 \Rightarrow f = -1 \end{cases}$$

More detail about obtaining the formula can be found in the reference which is given before. The SVM has the limitations to reach the aim. The equation just can use for the binary problem and the binary problem is linearly separated, but they should follow the non-linear separation and also multiclass separation. The problem is solved with using kernel function which gives $K(X_i, X_j) =< X_i, X_j >$ and it has the two kinds of techniques which are Polynomial and

Gaussian kernel. To classify the data to the each specified class, they used two ways. In the first one,the N-class problem or "one - versus all", they thought the binary data to realize the one class from all. In this case the new point used majority vote to find the class. The method has the two drawbacks. The first one had a zone of space. The other drawback was training each classifier that depended on whole of dataset.
The second way is "one-versus-one". In this method to build the classifiers, they tried to use all the pair combinations of the N classes. They obtained "the following equation" which is claimed in [7].

$$C = max_{k=1..N} Card(\{y_{i,j}\} \cap \{k\}), y_{i,j}$$

The next stage is the determining of the features in the each activity, because performing the activity is different from one person to another one. Thus they computed a large number of the parameters. Then they used principle component analysis (PCA) to extract the parameters for the various activities. So they had a set of the 42 parameters with the 5 modalities and each modality have the different indexes. They created the vector to use in the SVM algorithm.
Before the beginning of the training, they needed to normalize the set of data. They found the mean and standard deviation of the data set for the each dimension of the feature vectors. It was created a new training of the data set. Another step was the making of the training data base. So they created the XML file which contained the following information:

- The ID of the individual
- Information about sensors
- Information about the synchronization of the kinematic sensors with the other

They used the "leave- one out" algorithm which is performed the same number of tests with the number of the items in the database. The experimentation performed with 13 young persons and the duration of test for each one was 15 min. They tested the classification with two methods of kernel, Polynomial and Gaussian. They got better result with using the Gaussian kernel. The result shows in the figure 5[7]. Fuzzy clustering [8] is the other approach to detect the abnormal behaviour with the focusing on the time. In this study, they used the RFID[1] to collect the data with the time stamp. In each RFID, there are six kinds of information but in this project they used just the two of them, ID and RSSI values. They used the RFID tags in the different location of the house and put the RFID reader on the wrist-banded to carry with

[1] "Radio frequency identification (RFID) is a generic term that is used to describe a system that transmits the identity (in the form of a unique serial number) of an object or person wirelessly, using radio waves." which is claimed in[11]

		Classification Results						
		Sleeping	Resting	Dress/undress	Eating	Toilet use	Hygiene	Communication
Activity	Sleeping	**93.9%**	6.1%	0%	0%	0%	0%	0%
	Resting	13.8%	**78.1%**	1.3%	1.3%	4.2%	1.3%	0%
	Dressing/undressing	6.25%	12.5%	**75%**	0%	0%	0%	6.25%
	Eating	0%	0%	2.2%	**97.8%**	0%	0%	0%
	Toilet use	0%	0%	0%	6.25%	**93.75%**	0%	0%
	Hygiene	7.1%	0%	0%	7.1%	21.5%	**64.3%**	0%
	Communication	0%	5.3%	5.3%	5.9%	0%	0%	**89.5%**

Figure 5: The result of using The SVM

the patient. So the RFID tag and readers can send a signal to each other to collect the data based on the movement.

They collected an RSSI time from the each RFID tag. They had a problem with unreasonable zero which caused by the detecting of the time. It means that they have zero in some cases because of the during time of the undetected tag. So they focused on the values before and after that zero. If they are not zero, so they have the unreasonable zero then they calculate the average of values there and replace the average with the zero. They had a short-term and a long-term behaviour models. They designed an algorithm by the using of the clustering of data, regarding the behavior modeling. So they needed to collect the data at least for the one week and the amount of data became very huge. In this case they need to make a sampling of data. The sampling period for the short-term is five seconds and for the long-term is two hours. The each data of behavior modeling is included a segment of 10 time sequences. Behavior modeling is made by fuzzy clustering. The data points are separated into seven clusters. Data points can be in the more than the one cluster with a membership between the 0 and 1 value or it can be outside of clusters.

They used the two methods to detect the anomaly detection. The first model was based on the two kinds of thresholds, a normal and an abnormal threshold. It means that they checked the maximum of the seven membership value if it is greater than the normal threshold then they have normal behavior. In the abnormal behavior recognizing, they checked the maximum of membership, if it is less than abnormal threshold so they had abnormal behavior. If the result is between the two thresholds, they requires to find a similar data pattern from all historical data. The activity is the normal behavior if it can be found more than ten times. In the second method, they added the two largest membership values of the data point and checked them with the threshold, if it is greater than the threshold, then it is normal behavior.

They also considered on long-term behavior modeling. They make summation from the membership values of each cluster for all data in the one day to prepare the long-term modeling. This is the “equation” which is claimed in[8].

$$S_k = \sum_{j=1}^{n} w_{kj}$$

- n: total number of data point
- k: number of cluster
- w_{kj}: Value of membership for data point j in cluster k

If the result is the 20 percent larger or less than the normal values of the training data during the day, the activity is an abnormal. Alarm does not run with the first abnormal data point. They made technique to raise the alarm. They set the threshold for the number of the abnormal activity. They used it to correct the misclassified. To check the result of the system, they simulated the daily activities at home for seven days. They built a normal behavior mode based on these data with using the fuzzy clustering."Figure 6"[8] gives the example of the activities and their time. In this experiment, they made eight short-term activities (S1-S8) and three

Figure 6: The training data of ADL

long-term activities (L1-L3). I just want to explain the two short-term and the one long-term from their work.

$S1.Livingroom(10min) \rightarrow 30sec \rightarrow Toilet(15min)$

Here, they want to check the staying of the patient in the toilet in 15 minutes is normal or not. The system did not run the alarm because it is the normal activity in the training data (Figure 6).

$S2.Room1(10min) \rightarrow 30sec \rightarrow Toilet(10min) \rightarrow 20sec \rightarrow atToiletdoor(20min)$

In this part they assumed the patient fell at toilet door. This is not happened before in the training time. The system runs the alarm with this situation by using the two short-term methods. For the long-term cases of abnormal behavior, they used the clustering of the result for the 8 hours and compared them with the result of the seven days of the training data. Then they used to calculate the percentage.

L1. Using the computer without taking nap by the elderly.

In this situation the patient stayed the long time in front of the computer and did not go to the other place. Clustering the result shows, the cluster 2 is less than 20% and the cluster 7 is greater than 20%. So this is abnormal behavior.

Detection of an abnormal behavior algorithm in the long term[1] is the other study to consider on the long term period with the using of the motion sensor. They implemented the algorithm to detect the anomaly behavior with the using of the clustering. They simulated the daily activities to check the result by using the motion sensor (PIR) in the different places of home. They made a historical inactivity based on the time from a patient and put them in a table. If the patient goes out of home, they give indicator -1 because the system can recognize the situation. After collecting the data for a while they got a pattern of the patient. If the inactivity become longer than the determined duration, then the alert run. So they have an alert line to determine the problem. The figure 7 is shown an alert line. The "formula" which is claimed in [1] is given to

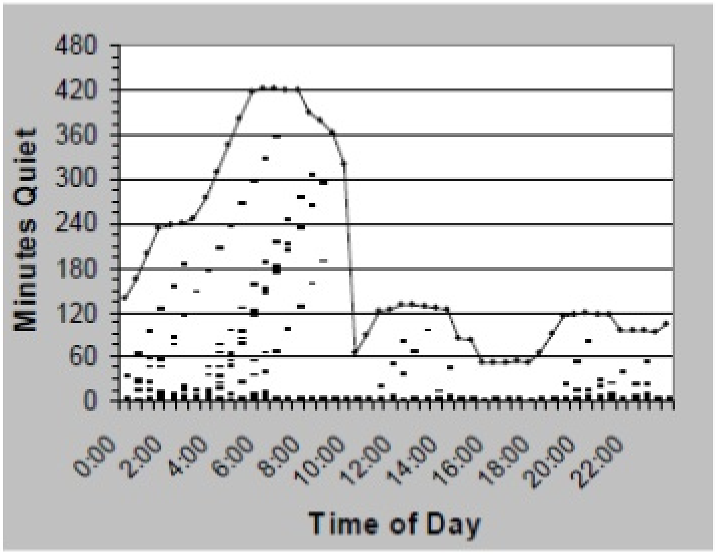

Figure 7: The Alert line[1]

detect an alert:

$$Alert_i = M_i + UB + VB_i$$

i is an index for the different time. Each of components of the equation is calculated by using the other formulas. To calculate the M_i first they calculated the m_i [1]:

$$m_i = PERCENTILE(data_i, MP)$$

The MP is maximum percentile which is used to delete the outlier elapsed of the inactive times. The $data_i$ is the historical elapsed inactive time for each interval.

$$M_i = MAX(m_{i-2}, m_{i-1}, m_i, m_{i+1}, m_{i+2})$$

To find the UB which is the uniform buffer, they followed the equation[1] in below:

$$UB = UBP \times MAX(m_i)$$

The UBP is a controller of the uniform buffer size. The last component VB which is used to increase the sensitivity during the active time is expressed [1] as:

$$VB_i = \frac{1}{\sum W_r} \left[\sum_{r=-2}^{2} (VBP \times m_{i+1} \times W_r) \right[$$

Where VBP is the variable buffer percentage. It used to control the activity level of the alert line. The W_r is an internal weight which is used to control the data of the current intervals from the last and the next intervals. They tested the algorithm for the 4 patients and got good results.

Some of the studies are concentrated more on prediction of action in the smart house for elderly people.

The paper from Malaysia University[12] is considered on the prediction of activity. Markov models is one of the techniques that was used previously to predict the user action in randomly sequence, and the SHIP algorithm is one of them. But the reliability of this algorithm is not very high around 60%. This is not acceptable for the smart home's application. The other techniques such as IPM and ONSI use the sequence of matching to predict the next user action. In this case IPM works better than SHIP. The technique which is used in this algorithm performed on the pairs of actions which happens in the sequence. IPM has a problem with the length of pattern, but ONSI solves this problem. On the other hand ONSI ignores the user preferred action and the relation between the environment state and the action. The combination of the pattern matching and the reinforcement learning help to solve the problem.

They used the simple example of smart home to explain better the techniques. They considered on time and the action is just switching off or on. They had two main components in the algorithm, the reinforcement learning and the pattern matching component. The first component used the Q-learning algorithm. The algorithm is a good choice for the smart home because it can support the online decision and does not need a model of the environment. Q-learning works by learning an action-value function. So if there is (S) such as environment device state and (A) such as action which taken in that state. The array of Q for the reinforcement learning explained as below which is claimed in[12]:

$$Q = S \times A$$

In the pattern matching component, they used the ONSI algorithm as a reference. The algorithm has the two mains directed graph data structures, multiple graphs of device usage history and the graph of most recent device which uses in sequence. The pattern matching operation begins from the recent graph with the most recent state and going back through that to find the matching with the each history graph. This function continues until the whole history graph is searched for pattern matching. The result of the process is the longest pattern matching. They check The result and it shows the algorithm have higher reliability than the other similar algorithms such as ONSI, SHIP and IPM.

The other technique that is studied to approach the prediction of action is based on the using context prediction[13]. In this research, they used the implementation of a Ubiquitous Context Aware Middleware (UbiCAM). UbiCAM makes the context model and it supports to access:

- The raw contexts from the different context sources
- Interpreting the context from the context reasoner
- Obtain the prediction of the future contexts from the past and the present context
- Ability to deliver the contexts between the two modes, push and pull

Their context prediction scheme is based on the set pair analysis (SPA) method. UbiCAM divides into five logical layers:

- Context sources layer
- Context acquirement layer
- Context management layer
- Context service layer
- Context application layer

SPA is a system analysis method. The technique is based on modifying uncertainty theory. They considered on both certainties and uncertainties as an integrated certain and uncertain system. They used "the formula" which is claimed in [13].

$$\mu(w) = \frac{S}{N} + \frac{F}{N}i + \frac{P}{N}j$$

μ is considered as connection degree of the set pair. N is the total number of the characteristics of the set pair. S is the number of the identity and P is the number of contrary characteristics. F equals with $N - S - P$. i is the uncertain value which is between -1 and 1. j represents as a contrary degree. By using $a = S/N$, $b = F/N$ and $c = P/N$, they got the following equation[13]:

$$\mu = a + bi + cj$$

The steps of analysing are:

- Constructing the class pattern set

$$C = \{C_1, C_2, ..., C_n\{$$

- Making the set pair to connect to the class path and the reference system

$$\mu(C_k, R) = a_k + b_k i + c_k j$$

- Calculating the connection degree with the prediction system and the reference

$$\mu(B, R) = a + bi + cj$$

- Calculating the IDC (identity, discrepancy, contrary)

$$d(B, C_k) = d_k = \sqrt{(a - a_k)^2 + (b - b_k)^2 + (c - c_k)^2}$$

- Classifying the prediction of the system.

$$f = \frac{\sum_{k=1}^{n} \frac{f_k}{d_k}}{\sum_{k=1}^{n} \frac{1}{d_k}}$$

In their work, they focused more on the medical issues. So they used the medical sensors to check the pulse, the blood oxygen and the body temperature of the patient. They prepared the different levels of alarm, depends on the notice mode.
Regarding studying the related subjects and the aims of the analysing activity, I decided to use the motion sensor and time stamp to design my application. I will explain the reasons and the aims of my decision in the following sections.

3 Rule engine

In this section, I defined the rule engine and the benefits of that briefly. A rule engine [14] has ability to process a dynamic set of the facts by using a set of the rules. Changing the fact can activate some other rules. Activating other rules can create the new facts. This ability is useful to process the events; it means the each event can use a new fact, by the rule set,and the event can make an action (Figure 8 [14]).
Using the rule engine provides some advantages [15] like:

- Declarative programming, It provides ability to say *"what to do"* not *"How to do it"*.
- It can separate the data from the logic. So data is in domain objects, and the rules include the logic.
- Other benefits are speed and scalability; The Rete algorithm is the battle proven.
- It has an efficient way to matching the rule patterns with the domain of the object data.

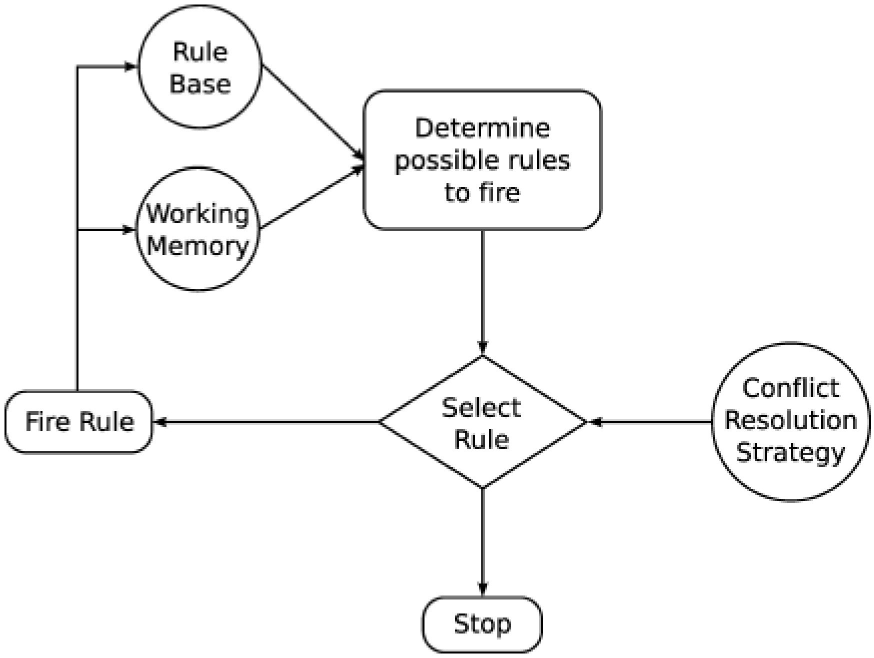

Figure 8: Basic rules engine architecture.

- The rule engine also provides abilities such as the centralization of knowledge, the tool integration, the explanation facility and the understandable rules.

The Jess Rule Engine is one of the rule engines. It has been written in Sun's Java language [14]. So it can be used in the Java environment or independent.

3.1 Using the Jess Rule Engine

According the benefits of the rule engine and the aims of application, I decided to use the Jess rule engine. The Coding in Jess is easier than the other programming software like Java. So training the user to work, modify and add the rule is more suitable and convenient than the usual programming.

The data in the jess is taken by a fact. The fact is designed like a database record [14]; it includes a number of slots. Each fact followed its structure as a template. The template is defined with the different slots which includes the different contents. The fact makes possibility to use the slots with the different values at the same time.

A rule contains of conditions and actions. It can simply define as an if-then structure [16]. The Left-Hand Side (LHS) includes the if-part and Right-Hand Side (RHS) includes the then-part. When a rule is fired, first the LHS part is checked if it was true then moves to the RHS part to make an action. The action in RHS can produce a new fact or delete the old ones or shows the result. The rules fire with the Last In First Out (LIFO) strategy. But Jess has possibility to

give the priority to fire the rules.
In beginning of my work, I did not have any experience to work with Jess. So I began to learn and work on it. As I mentioned before Jess has ability to use in Java or independent. I used both properties in the application. Learning to work with Jess was a good experience and a challenge for me. During my work, I learned to:

- Design a template of fact based on the requirements of application
- Assert the fact
- Define the rules, running the rule
- Define a function, calling the function in the rule
- Define a query, running the query
- Give the priority to the rules
- Use the different commands

4 Problem definition

In this section, I want to describe the reasons of choosing the motion sensor, the duration of time and using the Jess rule engine for my profile. With evaluating and comparing the different approaches and the aims of the *safer at home* project [5], there are important features that I should follow in my profile:

- Reliability, flexibility and quality of system
- Patient should feel comfortable in the house with the system
- Keeping the privacy of the people in the house
- The logic of the system should be simple and understandable for the user

The application should recognize the activities of patient in the home. So I need the sensor to follow the person in the house and send the data to the system to obtain the result. With consideration about the items which I mentioned above, I decided to use the motion sensors. They are installed in the different places in the house. By using this kind of sensor, the person feels free and does not have any stress or annoying about carrying the sensor with their selves or putting on the cloths. The sensor can send the detected motion signal and the process can follow the person in the different places.
Another reason to choose the motion sensor is the keeping of privacy of resident in the home. For example using the video camera to monitor the patient in the all places of the house is not a good option. It can annoy the elderly. In this case, I think the motion sensor is a good choice. As I mentioned in the related work, some studies used a pattern of activities to follow the person

in the house. The pattern should be reliable and flexible. I decided to use the time to prepare the pattern. Time is useful index to recognize the ADL and to analyse the activity. I will explain more about this part in the Behavior modeling section.

In some studies with the related subject are used the techniques which are just understandable for the skilled persons. But the maintenance and the developing of the system during the time based on changing the situation of the patient are the important features. So the system should implemented with high flexibility to use with the different alternatives, but in some cases the user needs to change some parts or add some more options for the system. It can happen because the application is used for the different persons. So the process should match based on the different situations. On the other hand, the health service has a significant and important responsible for elderly. So the application with understandable conditions for the caregiver is a useful option. This benefit makes an opportunity to improve the system directly by the unskilled responsible person. So according the definitions and benefits of the Jess rule engine, I decided to use the Jess to evaluate the result.

With collecting the data from the sensor and using the time, the application can analyse the activity. The behavior modeling is used to make the conditions to recognize the situation of the resident in the specific time and the application gives the result to the user.

4.1 Behaviour modeling

Before I begin to explain the definition of bahvior modeling in my profile, I will describe the reasons to use the behavior modeling and the benefits of that. The health service has direct relation with the smart home for elderly. If they have an opportunity to study the behavior of the person in the house, it can help them to make better decision in the terms of health for the patient. For example the person is under medication to regular the sleeping during the night. The caregiver can survey the result by studying the duration of sleeping. So they can check the reaction of person with medicine and they can decide to reduce or change the medicine. Also the system can give the result in the urgent situations to them by using the behavior modeling. The behavior modeling is a bridge between the caregiver and the system. So it should be easy to understand for the user with different knowledge and proficiencies.

To define the behaviour modeling, I consider about the realizing of the activity in the home. Recognizing the usual pattern of the activities and keeping them as areference are important to evaluate the mode of behaviours.

The modeling is based on the duration of time for the different activities. The duration of time is divided into the long and the short term. I give the examples to declare my opinion.

Time of Day	8-10	10-12	12-14	23-8
Activity	Eating breakfast	Watching TV	Going out	Sleeping
Motion	Normal	Normal	Abnormal	Abnormal
No motion	Abnormal	Abnormal	Normal	Normal

Table 1: example of daily living

Sleeping in the night is the long term activity, but using the toilet is the short term activity. The application should have flexibility to follow the two terms.

In the daily living, there are the different activities with the short and the long term. Some of them are repeated the several times in the day. So there are some behaviours which are not depended on the exact time, like using the toilet. It can happen in any time of day. But some behaviors are more depended on the exact time like sleeping in the night.

First step to make a daily living pattern is considering about the different activities of a person at home in the several days. Another important factor is the place of motion in the different time. I used this factor as the basis of my application. It means by using the pattern of daily living, it is possible to analyse the movement in the specific place (ex. sitting room) for the specific time (ex. after midnight). If the behavior is detected as an unusual behavior, regarding the initial definition. So an anomaly behavior is happened in that time. However the same behavior in the same place but in the different time can be a usual activity. Table 1 is shown an example for a daily living and the result of the analysing of behavior.

As we know all the people have the different habits and routines for their life. The retirements and the old people also have the different plans for their daily living activities based on their situations and health. So I decided to make an ability for my application to follow the different patterns. The user can enter the specific pattern for one person and get the result by analysing that activity. I will explain more about this property in the other sections. Thus the application is produced the result just for one person with the specific behavior.

I did not have a real motion sensor for my study in the beginning. So It was not possible to create a pattern recognition for a real situation. But I used the related study as a reference. It helps to make an example of ADL pattern just to test the application. In my thesis, I just considered about two activities. One of them is the depended time activity, it can be a short or a long term, and the other one is not the depended time activity. It can be repeated several times. The activities happen in the two separated places but the process is following them simultaneously.

5 Algorithm explanation

In this section, I want to explain the steps of the algorithm and the procedures to gain the result. In my work as I mentioned before, I did not have real data which is collected from the sensors in the beginning of my work. So I should prepare a data simulator for the process. Simulating data is necessary to make a test and to check the result for the different situations. It helps to improve the algorithm and to find the solutions for the problems which caused during the test. I explain all the steps of my work and challenging which I have during them in the following sections.

5.1 Overview of the algorithm

I divided the procedure to the 8 major modules, each module includes the own parts. The items in below represent the steps of the process. Each step needs the out put from previous one to begin its function.

1. Read the data from the input file
2. Simulate data for processing based on conditions from the input file
3. Synchronize the time with the generated data
4. Begin to assert the facts by using Jess
5. After each assertion check the facts with the Jess rules and get the new fact as a result
6. Make the query to get the final result
7. Evaluate the result to make the output
8. Make the different levels of alarm based on the result

5.2 Generating data

In this section, I explain why and how I made a data simulator. The reasons of using the simulator are:

- Covering the lack of sensors
- Testing the algorithm
- Checking the result with the expectation
- Solving the problem which caused by unexpected result

First I tried to make the random data to obtain the result. But It caused the unwanted data in the places based on my scenarios, and there was no logic between the data in the two places as well. So I began to prepare a simulator for the data. The simulated data must have the following properties:

- Following the logic based on the input file.
- Arbitrarily
- Having ability to synchronized with each other and the real time

As I mentioned before, by studying the life of the patient in the house in the several days, the pattern of daily living can be recognized. I assume that the user have the pattern of ADL from the patient and he/she wants to consider about the activity in the specific place in the house for the specific time. Before analysing the request, the process needs some information form user about the pattern. The user should have to enter the information about the activity in a form. The form should be easy to access and clear to filling out. The application generates the data based on some information from the form. Other information in the form are used to analyse and make an output which I explain later. Figure 9 is shown the form.
The activity information form includes:

- Location: the user should insert a place of the house. The place is used to follow the activity of patient during the time. The simulator produces the data for this place as a major place.
- Duration of time: in this part the user can entered the minute or hours, because I want to check the one activity which is depended on the time but it can be a short or a long term activity. I give some examples for different situations to clear my aim:
  - Staying the patient in the location for couple of hours or minutes and checking the activity there.
  - Check the sleeping time for the person under medications.
- Start time: the time of beginning the activity to following the patient.
- Type of activity: the user should insert the aim of activity, as I discussed in the behavior modeling, and by focusing on the example in the table 1, the application needs to get this information to analyse the data. In the figure 9, the value in the type of activity part is moving. It means that the usual activity is concerned on the moving. So the result with normal behavior means there is usual moving activity there during the time. If this part includes no-moving, it means that the normal behavior concerns on no-moving.
- e-mail of contact person 1: in this part the user should enter an email from the family member of the patient
- e-mail of contact person 2: the user should write an email from the care giver of the patient

After filling out the form, the user saves the file and the application can use it as an input file (the instruction is declared in Appendix A).

	A	B	C	D	E
1	The activity information				
2	location	sitting room			
3	Duration of time	7 minute			
4	start time	13			
5	Type of activity	moving			
6	e-mail of contact person1	leila.chinaveh@gmail.com			
7	e-mail of contact person2	yousef.niknam@gmail.com			
8					
9					

Figure 9: Form of Input file.

The simulator should produce a data which is represented of a motion signal. So I use the number 1 instead of the detected motion and 0 when there is no motion.

The process should have ability to work in the real time. I decided to check every second of activity in the process to increase the reliability and the quality of the system. So the application can use the second of the time to evaluate the activity. It helps the system to work with the high precision.

To prepare this feature, the process uses the value of the duration of time to generate the length of data, the application makes the value from minute or hour to the seconds. By the number of the seconds, the process produces a list of data which includes 0 and 1.

The data simulator should have ability to produce the normal or the abnormal activity data. It means that the application has the alternatives to choose between the two behaviours randomly. It helps to test the algorithm in the different situations. I made the two functions to prepare the data for the moving or the no-moving activity. The process works with the two activities. So it needs to make the two lists for the two places. Each list should be unique and randomly, but I have to hold the logic between them. It means when the person moves in one place, from the other place should not be detected any motion signals at the same time. The application makes the first list by using the input form and the moving or the no-moving instructions. The normal moving instruction is shown in the formula:

$$f(x) = \begin{cases} 1 & \text{if } x \text{ less than } 7 \\ 0 & \text{if } x \text{ greater than } 7 \end{cases}$$

It means that the moving list approximately is included the 70% motion signals. The x is chosen randomly between 0 and 10. So the result of the list is different for the each running. The *f1(x)* makes the no-moving list for the process. It works opposite the *f(x)*.

$$f1(x) = \begin{cases} 0 & \text{if } x \text{ less than 7} \\ 1 & \text{if } x \text{ greater than 7} \end{cases}$$

To coordinate between the two lists, the simulator keeps the first list as a reference. Then it makes the second list according the first one. The second list which is represented for the second place. It should have the frequently data of an activity which is repeated during the time. The second activity, is using the toilets and second place is the bathroom. To produce the second activity, I considered about the motion signals which are received continuously for the period of time. The period of time in my application is chosen for 28 seconds. So in each 28 seconds if we have the persistent signals, it means the person is using the toilet. The period of time for the second activity is just an assume to check the activity. The activity can happen in several times. By the following steps in the below, the first list becomes organized with the second.

1. If the function equals zero
2. Check the rest of the length of list from the last element which is inserted, if it has enough places to insert 28 elements because the generator should insert 28 zero
3. Begin to insert the zeroes
4. After finishing the inserting, continues the function
5. The steps 1-4 are repeated randomly over producing the first list.

Figure 10 is shown a diagram of output from *f(x)* with a period of using the toilet is seen between the activity. Figure 11 is represented a diagram of result with using *f1(x)*. In this diagram also

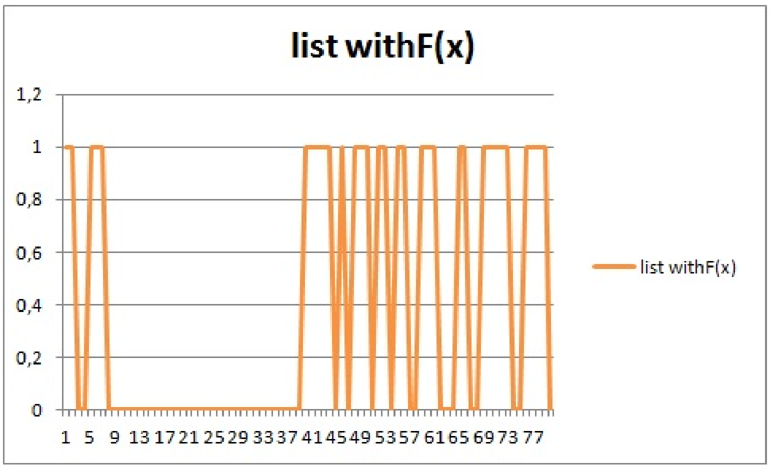

Figure 10: First list with using *f(x)*.

there is a period of using toilets. With comparing the two diagrams, it is obvious that the *f(x)* makes more motion signals (which is represented as number 1) than *f1(x)* regardless the using

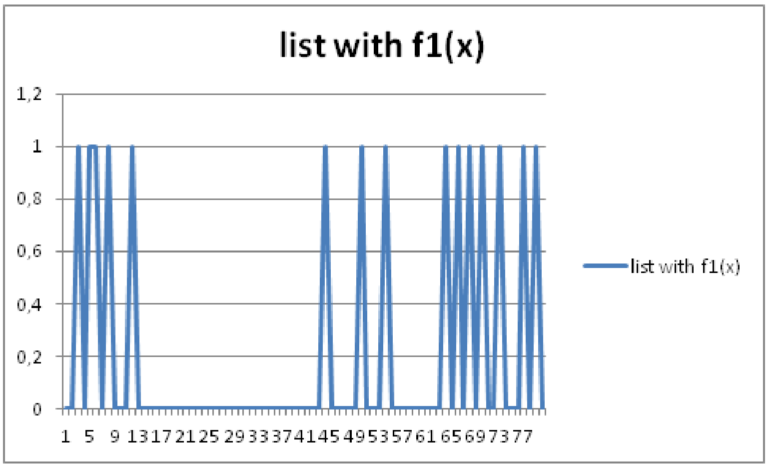

Figure 11: First list with using *f1(x)*.

of the toilet between the activity.
After producing the first list, the simulator begins to make the second one by using the first list and the other conditions to organized the two lists. I will present the steps of the process in the next section.

Filtering the simulated data

As I mentioned, the synchronization between two activities is important to get the result. So the simulator should follow the steps in below to prepare the second list which is represent the activity in the second place and filtering the unwanted data.

- Making a unlike list from the first list.

  The figure 12 shows the vice versa signals for the second list. But it is not suitable for my scenario. So the simulator should make a filtering to remove the unneeded data in the next step.
- Checking the second list and delete all the single and unneeded 1, and just keep the group of 1 which includes the 28 elements.

  The figure 13 is shown the signals after using the filtering.

I also make a condition to produce a second list without any motion signals. This kind of list is made randomly as well. It shows that the patient does not use the toilet during the activity. The figure 14 is shown a sample of two lists after the producing and the filtering. The first list shows the motion signal in the sitting room. The second list shows the motion signal of using the toilets. Also the figure shows the smooth coordination between the two activities.

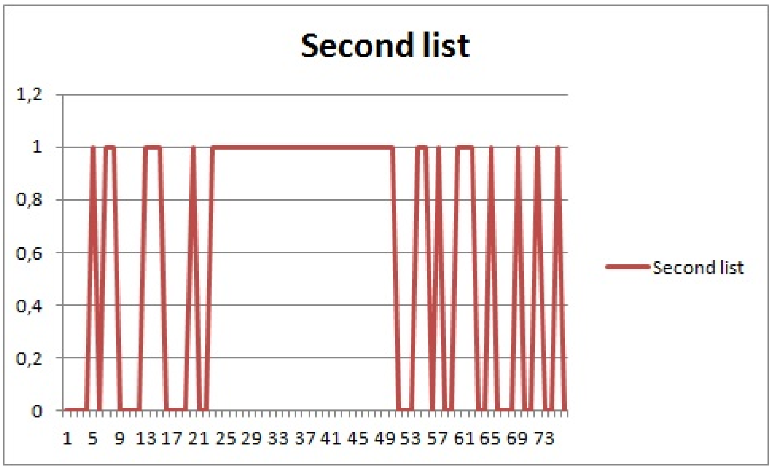

Figure 12: The second list before filtering.

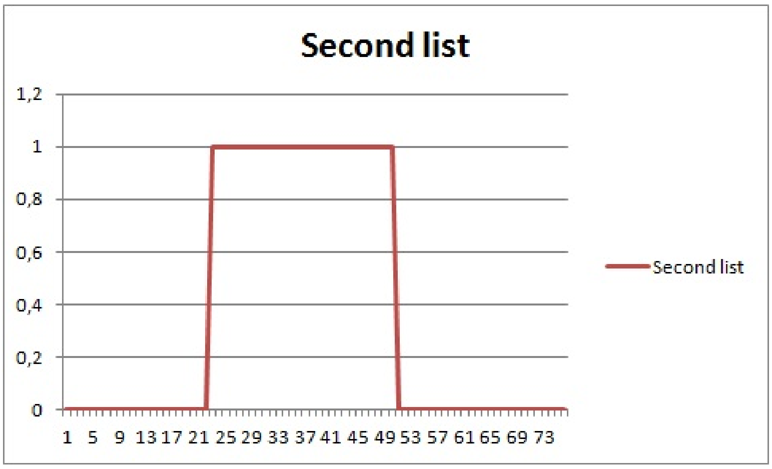

Figure 13: The second list after filtering.

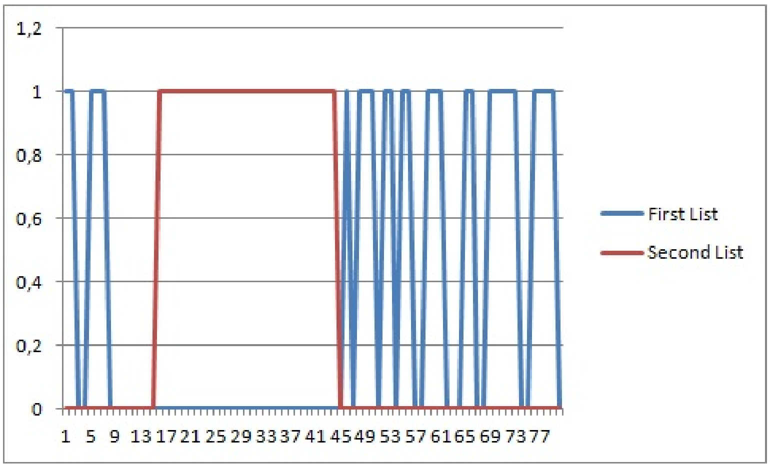

Figure 14: The result of two lists for two activities.

5.3 Synchronizing the data with the time

The algorithm should have ability to work in the real time. To test this ability, I should synchronize the real time with the generated signals. It means that the application should produce the hour, minute and the second in the real time to add to the simulated data. They are used to assert the fact. I prepared a procedure to produce the real time to add as a time stamp to the data.

The signals should be collected in every second. Each element in the list is represented of the one signal and they should be adjusted with the each second. This process should be performed for the two places and each second should be appointed to the two lists of signals. So the result is the two signals from the two specific places for the each second. Figure 15 is represented the mode of signals from the two lists in each second.

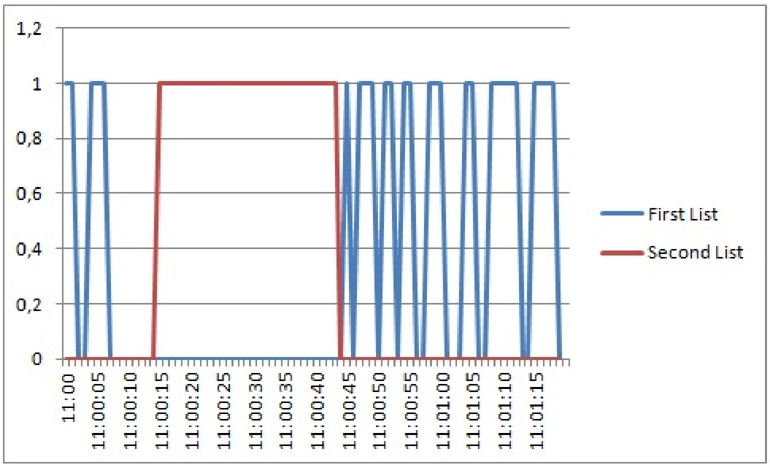

Figure 15: Synchronization the signal with the time.

5.4 Assertion of the facts

First I want to explain the syntax of the fact and the definition of the fact. The fact is divided to the three categories [16].

- Ordered facts
- Unordered facts
- Shadow facts

In this application is used only the unordered facts. Creating the unordered fact needs a template. The template includes the name and slots or multi-slots [14]. The slot is a name to store a simple value. It can be limited to one type of value. The multi-slot is a name to save a list of values. This one also can be restricted to one type of value.

The process should keep the signals with their time stamps after collecting them from the sensors in the real system. This is necessary to evaluate the activity. So I should define a fact which includes those properties. The *sensor* fact has ability to store the values of the time and the signal from the specific sensor(figure 16). The template includes a name of fact, *sensor*, and five

```
(deftemplate sensor (slot place) (slot hour) (slot minute) (slot second)
(slot data) )
```

Figure 16: The template of sensor fact.

slots. The *place* slot indicates the place of the received signals. The time is saved in the slots of *hour, minute* and *second*. The last slot, *data*, takes the value of signal for that specific time. The *sensor* facts should be asserted from Java because the data is simulated by Java.
With beginning of the process, the fact is asserted. The figure 17 shows the facts for a second of time. It shows the two facts from different places with data at the one moment. The assertions

```
(MAIN::sensor (place 23) (hour 13) (minute 40) (second 49) (data 0))
(MAIN::sensor (place 22) (hour 13) (minute 40) (second 49) (data 1))
```

Figure 17: The sample of the asserted sensor facts in the one second.

of the facts continue in the real time until the duration of time is over.
Another facts which the rules need to start the running are the start time and the duration of activity. I divided the duration of time into hour and minute. So I made two facts to save the specific time for them. The template of facts show in the figure 18. The *start-stop-hour* and

```
(deftemplate start-stop-hour (slot start) (slot stop) )
(deftemplate start-stop-minute (slot start) (slot stop) )
```

Figure 18: The template of facts.

start-stop-minute facts are used to save information from the input form. As I mentioned before, the information of the start time and duration of activity are required to analyse the data.
The *start-stop-hour* fact is used when the user wants to check the activity in the several hours. The process takes the start hour and calculate the finished time by the value in the duration of time and saved in the *stop* slot.
If the user wants to focus on a activity in the minutes, the *start-stop-minute* fact should be

used. In the *stop* slot is inserted the minutes between the 1 to 59. The *start* slot in the both facts contains the start time of activity. The facts in the figure 19 shows the assertion of start and stop time. The example shows the user considers to evaluate the activity in 34 minutes.

```
(MAIN::start-stop-hour (start 12) (stop 0))
(MAIN::start-stop-minute (start 12) (stop 34))
```

Figure 19: The sample of the asserted facts .

After inserting the first pair of *sensor* facts and the required facts, the rules begin to check them. So based on the conditions and the results of rules, I defined the other facts. I represent them during the explanation of rules.

5.5 Processing with the rules

Defining the rule is required two things, the format of rule in Jess and the logic behind the rule. Defining the rules need to look at the problem in a different way. The rules have two parts, the condition and the action. Thus the rule needs the fact to check the part of the condition.
The syntax of the rule is shown in the figure 20.

```
(defrule name of the rule
make IF-part
=>
make THEN-part)
```

Figure 20: The syntax of rule.

The properties of rules in my application are:

- Checking the signal in the different places in the same time
- Recognizing the dissimilar states based on the definition
- Checking the duration of time during process
- Preparing the new facts to continue the processing

I considered about the different states which are created by the signals from two places. Table 2 is shown the possible states of the signals in the two places. So the rules should have ability to cover the states and to produce the new facts based on them.

Place 1	0	1	0
Place 2	0	0	1

Table 2: the possible states of receiving data in a second for the two places

The rules are written in the Jess file. The process from the Java connects to the Jess file to run them for the each second in the real time. After asserting the required facts, the rules begin to run. The Jess file includes:

- The facts which are required to produce after running each rule.
- The queries to find the specific value in the facts.
- The rules, according the different situations.

To recognize the mode of signal in the different places, I should check the following values in the rules:

- Checking the slot of *data* in the *sensor* fact
- Checking the start time and the duration of time by the two options, minutes or hours
- Making a decision for the place of signal and produce the new fact

If the *data* slot is equal 0 in the first *sensor* fact, it means that there is no motion signal in the one place. So the rules should check the situation of the second place in that time. The rules should follow the algorithm in below to find the mode of signal.

if $sensor_{.data} = 0$ **then**

  $A \leftarrow sensor$

end if

if $start - stop - minute_{.stop} > 0$ **then**

  $B \leftarrow start - stop - minute$

end if

make query to get($A_{.timeStamp}, A_{.place}$)

$A_{.newplace} \leftarrow find\,value\,of\,another\,place\,from(A_{.place})$

make query to find($A_{.timeStamp}, A_{.newplace}, A_{.data} = 1$)

make query to get($B_{.start}, B_{.stop}$)

if $Compare(B_{.start}, B_{.stop}$ and $A_{timeStamp}) = True$ **then**

  if $newplace = Toilet$ **then**

    Assert a new *Motion-out-SittingRoom* fact

  end if

end if

make query to find($A_{.timeStamp}, A_{.newplace}, A_{.data} = 0$)

make query to get($B_{.start}, B_{.stop}$)

if $Compare(B_{.start}, B_{.stop}$ and $A_{timeStamp}) = True$ **then**

if $newplace = Sittingroom$ **then**

Assert a new *not-moving* fact

end if

end if

The rule make a new fact. The fact shows the mode of signal and the time of that action. So I prepared the template to make a new fact. Figure 21 shows the template and an example of running the rule. It shows that the movement signal recognized out of the sitting room. When

```
(deftemplate Motion-out-SittingRoom (slot hour1) (slot minute1) (slot second1))
(MAIN::sensor (place 22) (hour 12) (minute 32) (second 33) (data 0))
(MAIN::sensor (place 23) (hour 12) (minute 32) (second 33) (data 1))
(MAIN::Motion-out-SittingRoom (hour 12) (minute 32) (second 33))
```

Figure 21: The example of running the rules.

the rules recognize the *data* slots of both places are equal 0. It means that there is no motion in the both places. So the new fact should show the time and mode of signal. The figure 22 shows the template and the example of running the rules. The example shows the rules did not recognize any motions from the two places in that moment. If the *data* slot are equal in the first

```
(deftemplate not-moving (slot hour3)(slot minute3) (slot second3))
 (MAIN::sensor (place 22) (hour 12) (minute 32) (second 41) (data 0))
 (MAIN::sensor (place 23) (hour 12) (minute 32) (second 41) (data 0))
 (MAIN::not-moving (hour3 12) (minute3 32) (second3 41))
```

Figure 22: The example of running the rules.

sensor fact. The rules implemented by using the algorithm in below.

if $sensor_{.data} = 1$ **then**

$A \leftarrow sensor$

end if

if $start - stop - minute_{.stop} > 0$ **then**

$B \leftarrow start - stop - minute$

end if

make query to get($A_{.timeStamp}, A_{.place}$)

$A_{.newplace} \leftarrow find\, value\, of\, another\, place\, from(A_{.place})$

make query to find($A_{.timeStamp}, A_{.newplace}, A_{.data} = 0$)

make query to get($B_{.start}, B_{.stop}$)

if $Compare(B_{.start}, B_{.stop}$ and $A_{timeStamp}) = True$ **then**

  if $newplace = Toilet$ **then**

    Assert a new *moving* fact

  end if

end if

make query to find($A_{.timeStamp}, A_{.newplace}, A_{.data} = 0$)

make query to get($B_{.start}, B_{.stop}$)

if $Compare(B_{.start}, B_{.stop}$ and $A_{timeStamp}) = True$ **then**

  if $newplace = Sittingroom$ **then**

    Assert a new *Motion-out-SittingRoom* fact

  end if

end if

The algorithm uses to recognize the moving in the sitting room and the motion out of the sitting room (figure 23). The algorithms work properly to recognize the mode of signal. The order of

```
(deftemplate moving (slot hour2)(slot minute2) (slot second2))
(MAIN::sensor (place 22) (hour 12) (minute 32) (second 34) (data 1))
(MAIN::sensor (place 23) (hour 12) (minute 32) (second 34) (data 0))
(MAIN::moving (hour2 12) (minute2 32) (second2 34))
```

Figure 23: The example of running the rules.

asserting the *sensor* facts with the different places does not make any problems to recognize the mode by using the algorithms.

After running the rules to dedicate the mode of signals, the process needs some other rules to check the evaluating of activity. To check the activity, the rules should calculate the number of detected signals in the different places. The function should run in the specific time of activity. The rules for counting the moving or no-moving signals checked the *moving* and *not-moving* facts which happen in the sitting room in the duration of time. So the next algorithm represents the counter of the *moving* facts.

if $start - stop - minute_{.stop} > 0$ **then**

$B \leftarrow start - stop - minute$

end if

if $Compare(B_{.start}, B_{.stop}$ and $moving_{timestamp}) = True$ **then**

$Counter \leftarrow Counter + 1$

Assert a new *count-move* fact

end if

The figure 24 shows the sample of running the rules to count the *moving* facts. The implementing of the counter for the no-moving facts is the same as moving. The rules use the *not-moving* fact instead of the *moving* one. The example shows the counting of the detected no-moving in figure 25. I should mention the algorithms which are expressed, used the short term to work.

```
(MAIN::sensor (place 23) (hour 10) (minute 33) (second 37) (data 0))
(MAIN::sensor (place 22) (hour 10) (minute 33) (second 37) (data 1))
(MAIN::moving (hour2 10) (minute2 33) (second2 37))
(MAIN::count-move (exist1 1) (count3 1))
```

Figure 24: The example to count the moving.

```
(MAIN::sensor (place 22) (hour 10) (minute 32) (second 48) (data 0))
(MAIN::sensor (place 23) (hour 10) (minute 32) (second 48) (data 0))
(MAIN::not-moving (hour3 10) (minute3 32) (second3 48))
(MAIN::count-No-move (exist2 1) (count2 1))
```

Figure 25: The example to count the no moving.

But I prepared the other rules to implement the long term with the same logic as well.

During my work with the Jess rule, I have a challenge to implement a part of the rules. I will explain them in the next section.

Challenging with Jess

The application should have ability to recognize the activity which is not depended on the time. This activity can happen repeatedly during the time. To calculate the number of events, I tried to make a loop in the Jess rule to recognize that motion which is repeated for a specific time.

After recognized that action, the rule should make a new fact. The fact shows the number of events which happen during the time. But I could not find any useful utility in the Jess to implement the loop in the real time. After checking the different sources, I got the following information about the loop in the Jess:

- The rule fires for each matching fact in the whole memory [17]
- The Jess rule has ability to run once [16].
- The rule can run for while, the loop is implemented by an global variable to make counter[16].

I tried to use the third item to make a counter in the rules but it did not work. Because the global variable is reseated before the beginning of the each run and the rules should run in each second. So there is no way to keep the updated value of the global variable in the real time.

Then I considered to get the result by using another way. So I decided to keep the time stamp as a value and checked that time continuously until the activity is done. But the Jess rule engine does not have any support to get time as a value[17].

The other effort to get my expected result from running the rule, was the creating of the new fact with a multi slot. The multi slot includes a list. I defined the template of fact with list. So instead of one value, the process can insert a list of values in the slot. The list shows the number of motion signals in the place. The problem with this fact is the query. So there is no way to make query from that list to count the elements of list.

Eventually, I implemented a counter in Java to count the detected motion and check the time stamp which happens continuously. A new fact is asserted with the result of counting. The figure 26 shows the template and the fact with the new value.

```
(deftemplate Using-toilet (slot done)(slot number))
(MAIN::Using-toilet (done 1) (number 1))
```

Figure 26: The template and a asserted fact

5.6 Result of processing

I decided to show the result of the processing in the useful ways. After recognizing the signals and finishing the evaluation of them. The application should make result for the situation of activity. When the duration of time is over, the application make queries from the facts to get the last result which is expressed in the following formulas:

$$sum_{mov} = \sum_{j=0}^{n} S_{movj}$$

$$sum_{no} = \sum_{j=0}^{n} S_{noj}$$

S_{movj} is the moving signal and S_{noj} is the no-moving signal. The data simulator makes the moving and the no-moving data with the different percentages randomly. The application uses the result of rules and the conditions to recognize the mod of activity. If the aim of activity is considered about the ***moving*** activity, so the result of simulated data is evaluated as below:

$$NormalBehavior \simeq \begin{cases} 70\% moving \\ 30\% no-moving \end{cases}$$

$$AbnormalBehavior \simeq \begin{cases} 30\% moving \\ 70\% no-moving \end{cases}$$

But if the aim of activity is concerned on the ***no-moving***activity. The process follows the structure which is:

$$NormalBehavior \simeq \begin{cases} 70\% no-moving \\ 30\% moving \end{cases}$$

$$AbnormalBehavior \simeq \begin{cases} 30\% no-moving \\ 70\% moving \end{cases}$$

Although during the activity in the sitting room, the application can recognize the number of using the toilets. The two results give opportunity to make the different levels of alarm.

According the result of processing, I prepared two levels of alarm. Preparing the alarms needs some information from the user. The application takes the information of contact persons from the input form. In this case the user inserts the emails. The emails belong to the two persons with the different priorities to contact. One person is a member of family and other one is the caregiver. I chose the email because nowadays the developed mobile phone receives the email as simple as a SMS. So the person does not need to sit beside the PC to check the email.

The reason of making two levels of alarm is giving the priority to the risky situations. The application with one alarm for all the abnormal situations increases the rate of alarm. It causes the decreasing of consideration about the alarm. Thus the different levels of alarm help to increase the attention and decreasing the false alarm. I gave the colours to the alarm to distinguish the levels. The alarm with the low level of risky is yellow and the alarm with the high level of risky is red. The purpose of the low level alarm is to recognize an abnormal behavior in the one place and the normal behavior in the second place. In this situation the process sends an email to the family member to send a warning about the patient. Table 3 shows the situations to make the yellow alarm with the different situations. As you can see the red alarm is happened when the both places have the abnormal situations. The red alarm means that application does not

Sitting room	Abnormal	Normal	Abnormal
Using toilet	Normal	Abnormal	Abnormal
Result of alarm	Yellow	Yellow	Red

Table 3: Result of running alarm with the consideration about the Moving or No moving evaluation

recognize any normal behaviours in the two places. So the application should prepare more reaction for this situation. The application sends the emails to the both of the contact persons. Also it runs an audio alarm. I tried to make a voice call from the application to the person as well. I checked to use the library from Skype or Google talk. I tried the codes but I could not get any result from them. I studied other possibilities to make voice call from a system. It is possible but it needs some special devices or technologies. One technique is using the IP phone. I made a voice speech to run for the voice call as well. But I did not have the enough stuffs to finish that process.

I should mention the application has other conditions to reduce the false alarm. I will explain more about it in the reducing false alarm section.

Another output from the application is ability to save the result and the details of activity with the date and the time for the each processing. It can help the user to keep them as history of the patient in his archive. It is useful to check the activity for the several days. For example, if the person have an urine infection, he or she uses the toilets more than the usual day. The user can figures it out from the history files. The figure 27 shows an output file with the details about the activity.

6 Reducing false alarm

Reliability is the important aim of the system. One way to make the system more reliable is reducing the rate of false alarm. For this purpose, I studied the different techniques with related subjects.

One technique is represented the different priorities for the anomaly situations to make an alarm[18]. They made four levels of alarms for the system based on the result. The alarm system includes mild abnormality, caution, high Alert and emergency. This is helpful way to reduce the false alarm. I also gave the two priorities to run the alarm in my application to reduce the false alarm.

Conditions to evaluate the number of using the toilet as a normal or an abnormal activity is

	A	B	C
1	**Result of Date**	**Wed Apr 11 12:46:00 CEST 2012**	
2	**location**	sitting room	
3	**Duration of time**	45 minute	
4	**start time**	12	
5	**Type of activity**	moving	
6	**Result of activity during the time**	Abnormal Activity	
7	**Result of using tiolet during the time**	1	

Figure 27: The result of the activity in the one processing.

another opportunity to reduce the false alarm. So only the number of using the toilet is not enough to recognize the mode of behaviour. Thus I made the condition to check the number of using the toilet depends on the duration of time. The result is depended on the duration of activity which is short or long term. The patient also can help to reduce the false alarm. The system can make a phone call to the patient in the house and waits to reply from patient in a specific time or using a vibrator on the wrist with a button to turn it off [19]. So if a risky situation is recognized by the system, and the patient answers the phone or turns the vibrator off, It means there is no need to run the alarm. I also recommend this benefit to use in the real system.

Using the threshold is another way to reduce the false alert[2]. But determining the level of threshold is very important to gain the result. The threshold with the high level decreases the detection. The threshold with the low level increases the false alarm. So finding an optimal threshold is very important to evaluate the result which is caused the alarm in the application. The figure 28 shows the different levels of threshold to detect the real result. The figure is represented the level a is too high and the level c and d is too low to use as a threshold. According the definition of abnormal and normal behavior in my profile, I chose a range to make decision. The summation of moving or no-moving should be more than 65% of the total signals to make a decision.The figure 29 shows the threshold to realize the abnormality. The

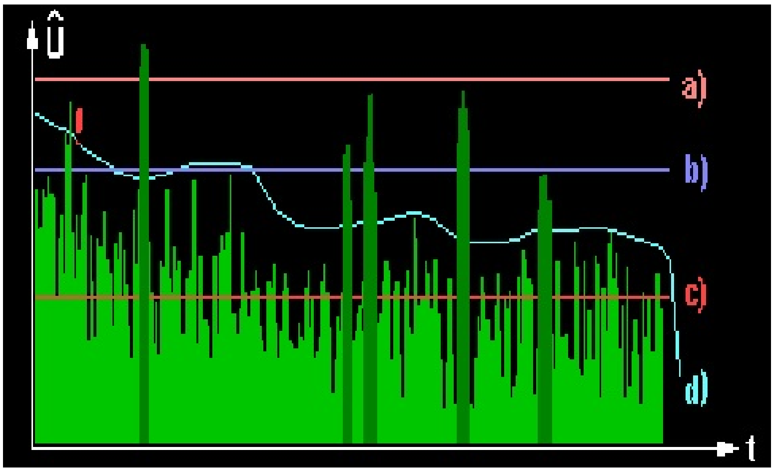

Figure 28: Level of threshold[2]

level of threshold works properly with the definition of the simulated data. But the threshold with real data can be different. The result from the real sensor should be evaluated to recognize the optimal threshold. After running the application with the different patterns and the aim of

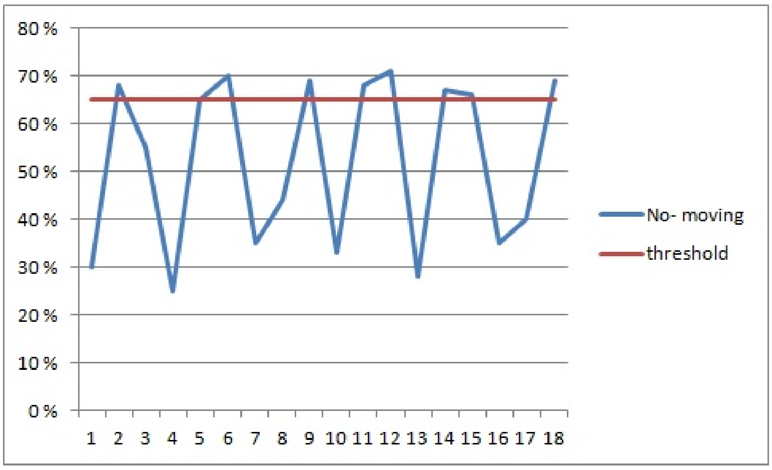

Figure 29: The result of abnormal behavior by using the threshold for the simulated data

the activities and checking them, I got the result which is shown in the figure 30. The result shows that the probability of arising the alarm in the short term is less then the long term. Also the red alarm runs less than yellow one in the long term and there is no running for the red alarm in the short term. It means that the red alarm just runs in the worst cases. The result indicates that the conditions work properly to reduce the false alarm.

7 Motion sensor

The application gives the expected result by using the simulated data. Then I decided to evaluate the application with the real sensor and the real data. Finding the sensor is not my responsibility

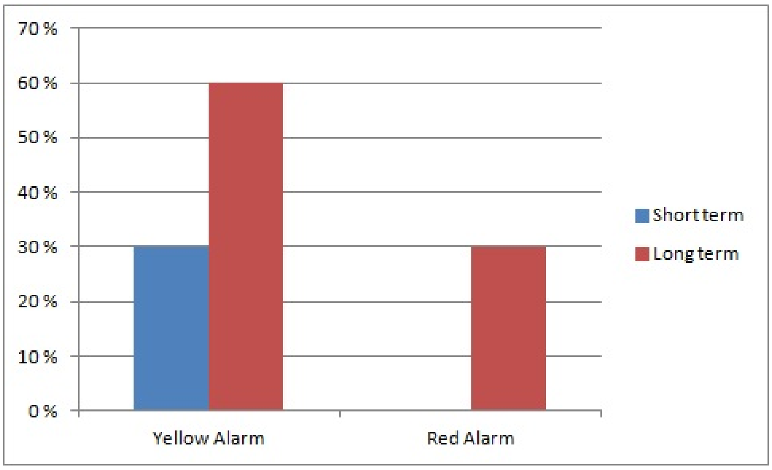

Figure 30: Result of arising the alarm in the abnormal situations.

for this project. But I tried to find some thing simple and useful to check my application. The sensor should have the following Specifications:

- The sensor can detect the motion
- Simple to connect to the PC
- The signal can transfer easy from the sensor to the application

The other things which is important for me to choose the sensor is the cost and the time of shipping. During my searching to find a sensor, I found the motion detectors, like PIR detector[20], 1111 - Motion Sensor [21]. But I could not use them, because their specifications are not enough with my requirements, or have the limitation to send to Norway.
When I could not find the suitable sensor, I tried to find another thing which I can use it instead of sensor. Thus I considered to find a software with using the video camera to detect the motion. Because the software is easier to buy and I do not need any extra or special stuffs except the video camera. So preparing a simple video camera which can connect to the PC is not a hard issue. I found some software with the ability to detect the motion by using the video camera. The most of this kind of software are implemented for the security of home or office. So they just checked the area until detecting a motion and run the alarm or send a warning to the user. I should focus to find a software which can give the detected moment and continue the detecting. As I mentioned from beginning, the system should keep the privacy of patient in the house. So I do not want to use the image or the video clip directly to detected the activity. According my idea, I desired to find an application which can send the moment of moving to my process as a time stamp. Finally I found an application which is more consistent with my idea. I describe the application and technique which I used to get the signal from the monitoring software in

the next section.

7.1 Using web motion sensor

The application which I used to produce the signals is the *Zebra-Media Surveillance System*[22]. The software includes several properties. I mention some of them in below:

- Motion capture software
- Ability to use IP and USB cameras
- Motion detection
- Sending SMS, MMS and Email alerts
- Recording the detected moment
- Saving video and captured image into the archive file
- Giving ability to change the configuration

I used some features of the application which are useful to my profile. The *Zebra-Media Surveillance System* has ability to take the picture in the moment of detected motion. It saves the picture in the archive folder with the time stamp. The time stamp is saved by the name of file. So I used the archive folder to get the time stamp from the file when the system detects a motion. First I tried to work with the files from one web cam. I succeeded to get the time stamp from each file. Then I worked to prepare the time and the data as the process needs for the *sensor* fact. The *Zebra-Media Surveillance System* just gives the moving signals. So the application should make the no-moving signals by itself between two detections. Also there is a little delay between the taking of capture from the process and the saving of file in the archive. Thus the time stamp of file is not equal with the real time from the system. So I should implement a process to synchronize the time between the application and the real time. The delay also caused some gaps between the detected signals. The figure 31 shows the signals before filtering. It shows some signals are missed during the collecting data. Thus preparing a filter to check the time stamp and filling the gap between the detected signals, is necessary. The figure 32 shows the signals after filtering. As you can see in the figure 32, the gaps between the signals is covered by filtering. The *Zebra-Media Surveillance System* have ability to adjust the sensitivity of motion detection. The sensibility can be very high to detect any motion even blinking. But I adjusted it to detect the motion like moving hands or walking. So the signal shows the dramatic moving. My application evaluates the signals from the two places. So I need to add another video camera to get the signals for the second place. With the two cameras, I should focus on the detected signals from the two sources. According my process, I need the signals from the two places in the same moment to run the rules. So the synchronization of

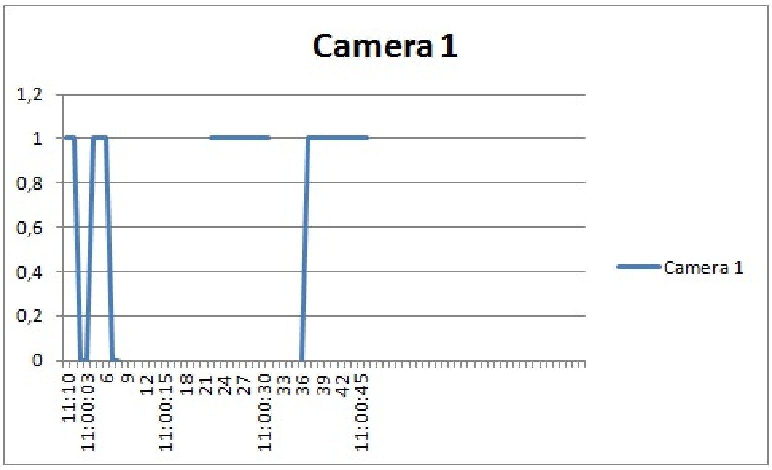

Figure 31: The diagram of signals before filtering

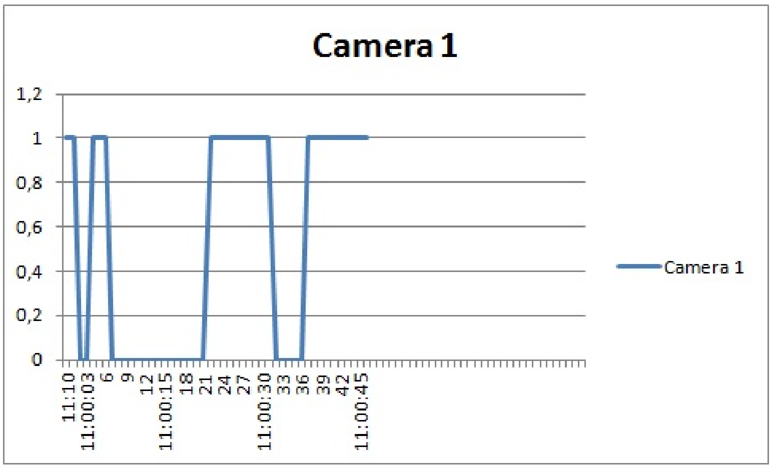

Figure 32: The diagram of signals after filtering

signals in a specific time is important.

The delay between the saving file and the real time is a problem to use the time library from Java directly. But the application needs to use the real time if there is no-moving before and after the detected signals. It should send them as a no-moving signal with the time stamp. So the application was tested many times to check the result by the different conditions and delays to organize the signals with the time in the two places. The figure 33 shows the output of processing after synchronizing the data. After checking the several outputs with the sources, I figured out that in some cases the *Zebra-Media Surveillance System* missed one second to save the time stamp. It is appeared as a tiny gap in the figure 33. It does not happen often. But I made a filtering for the process during the collecting of data to cover the gap as much as possible. It works very better but some tiny gaps can create between the signals rarely. If it happens, it is not more than one second gap and with comparing to the total signals, it is negligible. Now the application has enough data to assert the *sensor* fact and get the result from running the

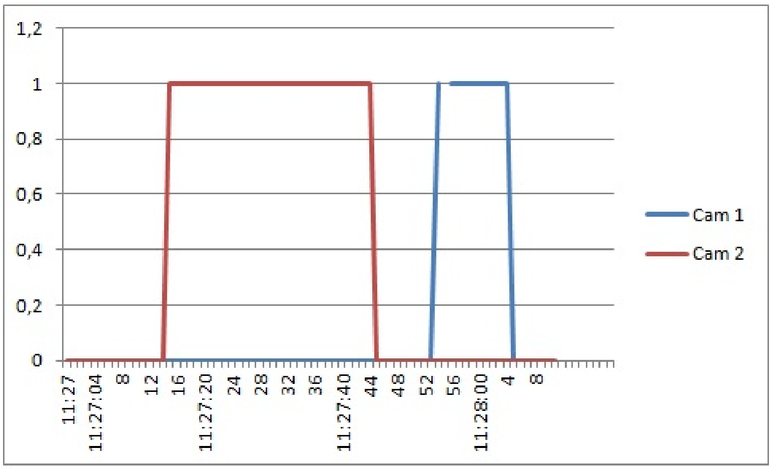

Figure 33: The signals from two sensors(video cameras)

rules. But the process had the same problem to count the repeated activity. So I produced a counter as well for the data from archive folder which is collected data from the bathroom's camera, to count the activities. After producing the counter the process works properly with the rules like the simulated data. So I made two options for the application to run. The user can choose the real sensors or the simulated data to run the process. The figure 34 shows the changing of procedure. I give a sample of running with using the sensors(video cameras). The

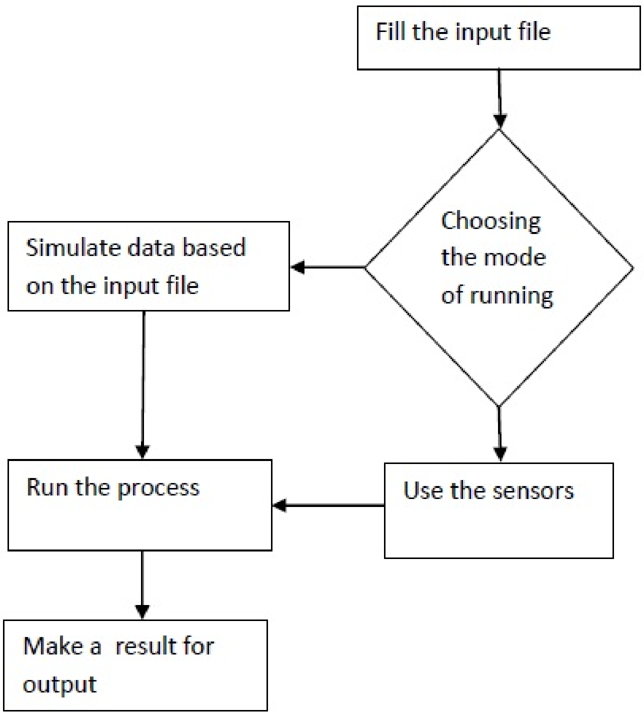

Figure 34: The flowchart with the options to run the process

figure 35 shows the signals which detected from the sensors(video cameras). In this example the aim of processing was evaluating the ***moving*** activity. As the diagram shows the moving is obviously during the time with the *Cam 1*. Using the toilet also is shown with the *Cam 2*. The expected result should be a normal activity for this digram. The figure 36 shows that the result of evaluating which is the same with expectation.

The sensor(video cameras) have a benefit to collect the data in any time but the simulated data

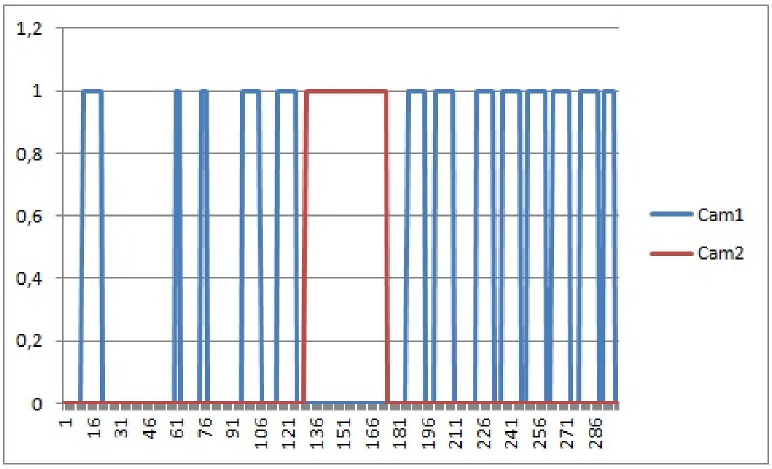

Figure 35: The normal behavior signals.

	A	B	C	D
1	**Result of Date**	**Thu May 24 13:05:02 CEST 2012**		
2	**location**	sitting room		
3	**Duration of time**	4 minute		
4	**start time**	13		
5	**Type of activity**	moving		
6	**Result of activity during the time**	Normal Activity		
7	**Result of using tiolet during the time**	1		

Figure 36: The normal behavior result.

can just gives the data for the duration of time. Comparing the diagrams of signals from the simulated data and the real data shows that the sensors(video cameras) can give more smoothly and logically signals than the simulated data which is produced randomly. The diagram of signals from the sensor also helps to study the activities. So by using the sensor in the real situation and study the diagram of signals, the ADL of patient can be recognized easier. This feature helps to determine the rate of false alarm and the optimal threshold. But this kind of sensor uses the remarkable parts of memory to save the files during the time. The application can remove the files after finishing the events. But I should mention this kind of software are used just to test the application instead of a real source. It helps to check the ability of the process to run and to work with the external devices.

8 Discussion and conclusion

8.1 Discussion

According the aims and requirements of this thesis, I obtained to organize the data by using the facts in the rule engine. The asserted fact have the different values in its slots. It gives the possibility to use the values from one fact. It helps to make a logical package of data with out worrying about their syntax. I should mention during the process the large amount of fact are asserted by running the rules, but it does not have a significant effect on the speed of processing. The behaviour modeling also is defined by using the start time, duration of time and the place of activity. The rules and the facts also are used to analyse the activity. After running the rules the user can get the mode of activity in the house. The rules have capability to evaluate the different situations. Also the application gives opportunity to the user to check the different patterns. It is possible by using the input form. I tried to keep the communication between the user and the application in the both side, the input and the output. So I tried to make the useful output when the processing is over. In this case, I also considered about reducing the false alarm. The result of testing with simulated data in the figure 30 shows the rate of false alarm after making conditions and using the threshold.

After getting the result with simulated data, I decided to check the application with real sensor and the real data. By using the *Zebra-Media Surveillance System*[22], I succeed to connect my application to an external device which works like sensors. It shows that the application have sufficiency to connect to the external devices and gives the acceptable result. By testing the application with the real data, the result shows that the level of threshold which the application uses for the simulated data, is not optimal for the real data (figure 37). It causes to increase the rate of false alarm and it decreases the reliability of the system. So I should determine a new level of threshold for the real data. So reducing the rate of false alarm for the real data will be one of my future work.

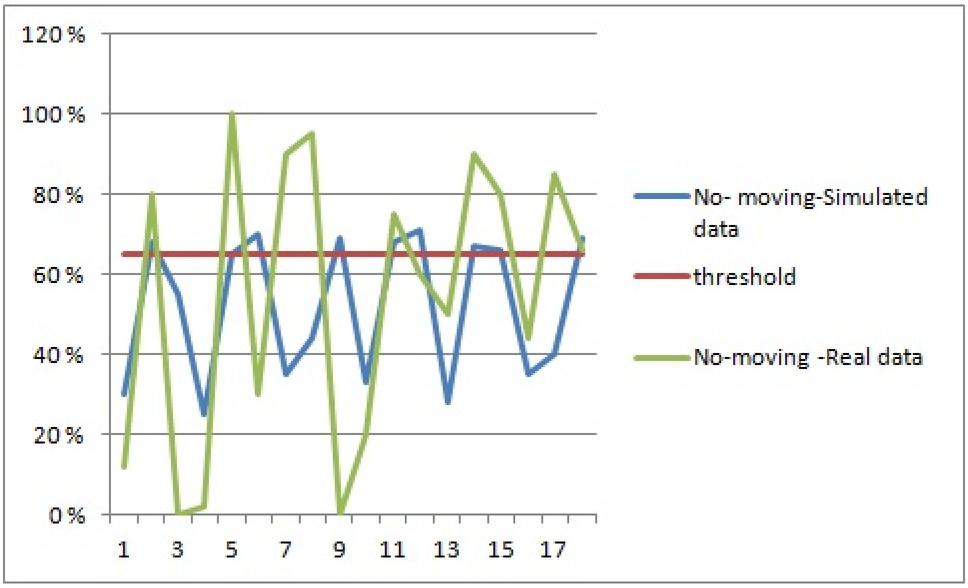

Figure 37: The result of recognizing the abnormality with using the real data and the simulated data

8.2 Future work

In the future work, I will work to reduce the rate of false alarm for the real data. The application should test in the real situations. It helps to calculate the detection rate and the false alarm rate for the real data and comparing them with each other. So the optimal threshold can realize with them. Preparing a reference of the risky situations is another subject which I want to work on it in the future. The application can compare each result with the all historical results to find the similar pattern to prevent any missing of risky situation or arising the false alarm. So the system should keep the history of the risky situations as a reference to compare with result. Then the process can make decision with higher assurance.
The application has a high potential to improve the algorithms and rules by considering about the more exceptions which important in the daily living. The future work aims at:

- Adding more places to check by the application
- Checking the result with considering about some exceptions like weather and holidays which have an effect directly on the pattern of daily life
- Preparing the ability to recognize the noise signal with real sensors, for example if the patient has visitors or pet in the house

8.3 Conclusion

The result presented in this thesis shows that using the rule engine is an effective way to analyse the activity in the home. Also time and duration of time are the important factors to recognize the behaviours and to evaluate them to get the result. I tried to give the priorities to the result of the abnormal behaviour to increase the reliability and to reduce the false alarm of the system. The using of the rule engine is a significant way to training the unskilled persons. So It gives a good opportunity to maintenance and improve the system by the user.

The rule engine gives the ability to work with different kind of sources and gets the acceptable results. It helps the system to be more flexible. It means that the application does not need to change the rules or facts with the same scenarios by using the different kinds of motion sensors. The result of testing the application shows that there is a difference between arising the alarm between the simulated data and the real data because of the level of threshold. So I should focus more on the reducing of the rate of false alarm.

Appendix A

Instructions of using application

The application needs to change some configuration before running. In addition of the file of codes, the application needs the two more files to run. One of them is the Jess file,*test10*, and the other is the Excel file,*Book2*. The file should save in their directories, E:\master\Book2, E:\jess\test10. The *Book2* file is used to fill the input form. Filling the form is explained in the *Generating data* section. I should mention to insert the email, the emails' accounts should be made from Gmail.

The application also needs to download the *Zebra-Media Surveillance System* [**?**]. The software uses the video cameras. So the application needs the two video cameras. They have to have the USB port to connect to the computer. Also the *Zebra-Media Surveillance System* needs some adjustments before using by application. The adjustments includes:

- Defining the archive folder for each video camera, the E:\sensor directory for the camera in the sitting room and the E:\sensor1 for the camera in the toilet.
- Changing the configurations of cameras:
  * To choose enabling the motion detection, figure A.1
  * To select the type of action after detecting the motion, figure A.2
  * To change the configuration of taking the picture. It shows in the figure A.3 and figure A.4 for each places
- Clicking on the cameras to start working.(Or the user can adjust the time table of cameras for the specific time which the user wants to test. So it runs automatically in that time.)

When the user is done with the all items which mentioned above, the process can run. The application can run from the *WriteExcel* class. I should mention that the codes need some external libraries to run. So the libraries should add to them. After running, the user should choose between the two options which shows in below by entering the number.

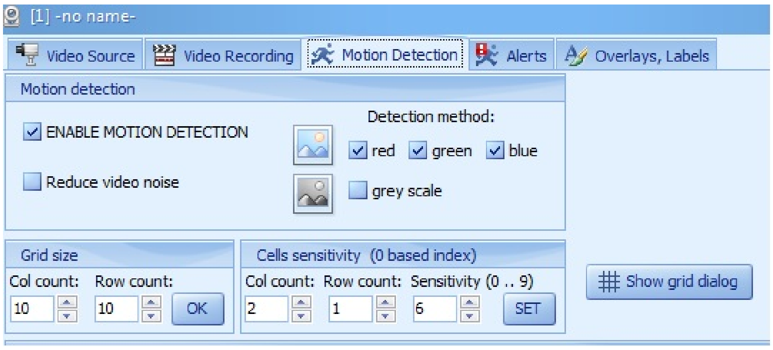

Figure A.1: Enabling the camera

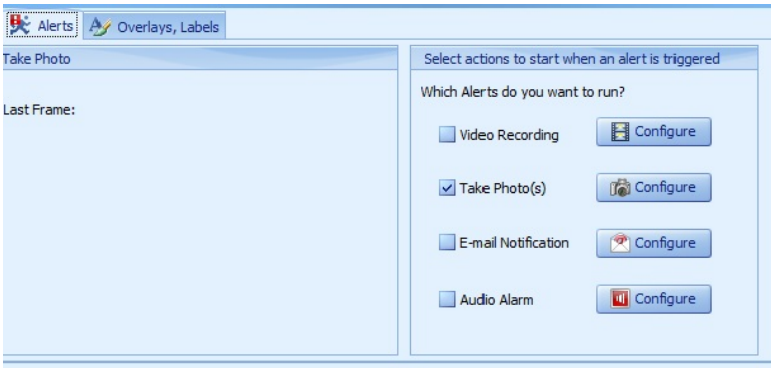

Figure A.2: Selecting the type of Alert

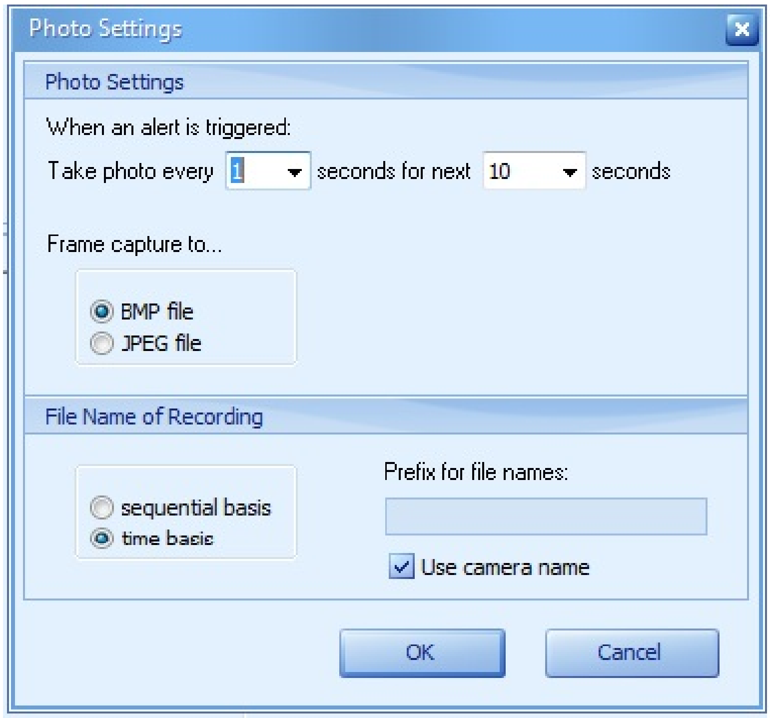

Figure A.3: The configuration of camera in the sitting room

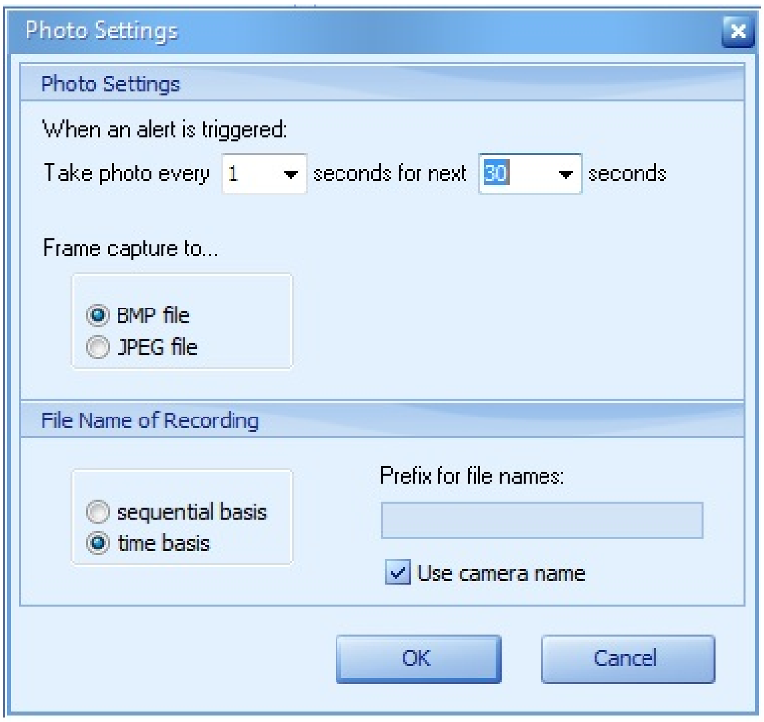

Figure A.4: The configuration of camera in the toilet

`To run the application with simulated data press 1`

`To run the application with real data from sensor press 2`

As it is shown by inserting number 1 the application run with simulated data which produces by simulator in the application. By pressing number 2, the application collects the data from the cameras. The result is shown after processing.

References

[1] P. Cuddihy, J. Weisenberg, C. Graichen, and M. Ganesh, "Algorithm to automatically detect abnormally long periods of inactivity in a home," in *Proceedings of the 1st ACM SIGMOBILE international workshop on Systems and networking support for healthcare and assisted living environments*, HealthNet '07, (New York, NY, USA), pp. 89–94, ACM, 2007.

[2] C. Wolff, "False alarm rate." `http://www.radartutorial.eu/18.explanations/ex10.en.html`.

[3] E. Pikington, "Population of older people set to surpass number of children, report finds," *The Guardian*, July 2009.

[4] "Aging in place." `http://ageinplace.com/aging-in-place-basics/what-is-aging-in-place/`.

[5] P. D. C. Rong, "Smart system to support safer independent living and social interaction for elderly at home (safer@home)," *University of Stavanger, N-4036 Stavanger, Norway*, 2012.

[6] A. Lotfi, C. Langensiepen, S. Mahmoud, and M. Akhlaghinia, "Smart homes for the elderly dementia sufferers: identification and prediction of abnormal behaviour," *Journal of Ambient Intelligence and Humanized Computing*, pp. 1–14. 10.1007/s12652-010-0043-x.

[7] A. Fleury, M. Vacher, and N. Noury, "SVM-Based Multi-Modal Classification of Activities of Daily Living in Health Smart Homes: Sensors, Algorithms and First Experimental Results," *Information Technology in Biomedicine, IEEE Transactions on.*

[8] H.-H. Hsu, K.-C. Lu, and M. Takizawa, "Abnormal behavior detection with fuzzy clustering for elderly care," in *Computer Symposium (ICS), 2010 International*, pp. 6 –11, dec. 2010.

[9] "Self-organizing map - wikipedia, the free encyclopedia.." `http://en.wikipedia.org/wiki/Self-organizing-map`.

[10] "A tutorial on clustering algorithms." http://home.dei.polimi.it/matteucc/Clustering/tutorial-html/kmeans.html.

[11] A. for Automatic Identification and Mobility, "What is rfid." http://www.aimglobal.org/technologies/RFID/what-is_rfid.asp.

[12] A. I. M. C. F. M.-Y. F. Reaz, M.B.I.M; Assim, "Smart home device usage prediction using pattern matching and reinforcement learning," Conference Publications, 2008.

[13] Y. N. Ma Shou-ming, Wang Ru-chuan, "Using context prediction for elderly health monitoring in pervasive computing environments," *International Journal of Digital Content Technology*, pp. 16–25.

[14] B. Eggert, "Using the jess rule engine to evaluate authorization policies," *University of Hamburg, Department of Informatics Security in Distributed Systems, Sicherheit in Verteilten Systemen (SVS)*, September 2007.

[15] "Chapter 1. the rule engine." http://www.jbug.jp/trans/jboss-rules3.0.2/ja/html/ch01s02.html.

[16] N. Maatjes, "Automated transformations from eca rules to jess." http://essay.utwente.nl/559/, July 2007.

[17] S. N. Laboratories, "Jess the rule engine for the javatm platform." http://www.jessrules.com/jess/FAQ.shtml, 2008.

[18] A. Arcelus, M. Jones, R. Goubran, and F. Knoefel, "Integration of smart home technologies in a health monitoring system for the elderly," in *Advanced Information Networking and Applications Workshops, 2007, AINAW '07. 21st International Conference on*, vol. 2, pp. 820 –825, may 2007.

[19] J. A. B. A.-A. C. K. F. E. K. N. B. D. F. K. K. Ilse Bierhoff, Ad van Berlo, "Smart home environment." http://www.tiresias.org/cost219ter/inclusive-future/%28%1429.pdf.

[20] "Usb plug-n-play motion detector." http://www.kadtronix.com/digio/motion/usb-motion.htm.

[21] P. Inc.2009, "Phidgets 1111 - product manual." http://www.phidgets.com/documentation/Phidgets/1111-0-Product-Manual.pdf.

[22] "Zebra-media surveillance system." http://www.zebra-media.com/zebra-media-surveillance-system.html.