Behavior Recognition for Elderly People in Large-Scale Deployment

Romain Endelin[1,2,3], Stéphane Renouard[3], Thibaut Tiberghien[2,4], Hamdi Aloulou[2,4], and Mounir Mokhtari[1,2,4]

[1] Laboratoire d'Informatique, de Robotique et de Microélectronique de Montpellier - LIRMM, France
[2] Institut Mines-Télécom, France
mounir.mokhtari@mines-telecom.fr
[3] Handco, France
{romain.endelin,stephane.renouard}@handco.fr
[4] CNRS, Image & Pervasive Access Lab (IPAL UMI), Singapore
{thibaut.tiberghien,hamdi.aloulou}@ipal.cnrs.fr

Abstract. Behavior recognition through ambient assisted living solutions for elderly people represents an ambitious challenge for actimetry. Numerous and versatile solutions have been deployed. However, a commercial adoption is still pending, due to scalability and acceptability constraints. Most research in Ambient Assisted Living (AAL) appears to have a heavy design, where precise features are first selected, and hardware architecture is designed accordingly. Although it may provide interesting results, such approach leads to a lack of scalability. This is why we experimented a lighter approach for a real deployment. The complexity is shifted from hardware to software, and we aim to make meaningful information emerge from simple and generic sensor data, in order to recognize abnormal and dangerous situations. In this paper, we will describe how to retrieve consistent information, so that residents' behaviors may be observed. This work might serve as a proof of concept that a light and generic approach fits in large scale deployments, with acceptable cost and scalability.

Keywords: Ambient Assisted Living, Human Abnormal Activity Detection, Sensor Deployment, Actimetry, Context Awareness.

1 Introduction

As population is getting older, concerns are raised about eldercare [9]. For various reasons, many families are not able to support their elders, and nursing houses are not the first choice for most of them. A reasonable solution for elders and their family is to let them be independent, at their own home. However, residents suffering from mild dementia are subjects to dangerous situations. Even though these residents may be supported by caregivers, they remain vulnerable during their absence. In this context, Ambient Assisted Living (AAL) systems are used to provide useful healthcare services for elderly people with continued

J. Biswas et al. (Eds.): ICOST 2013, LNCS 7910, pp. 61–68, 2013.

supervision. An interesting feature would be to detect abnormal behaviors, such as falling or running away, in order to notify caregivers. Many prototypes of smart spaces have been realized, however, only few of them involve long-term deployments in real conditions [8].

AAL requires context awareness, handled by deducing situational information from a datastream of sensor events [6]. It also requires the introduction of a formal unit for measurement of the amount of activity, as it can be defined through actimetry [5]. In most deployments, the system is expected to detect precise situations, such as "open the fridge", "watch TV", etc. [10,2]. For this level of actimetry, specific sensors are required on numerals objects, and algorithms observe the emerging patterns in those sensor data to deduce activities. Such approach may be relevant for specific applications, e.g. medical treatments [1] or ergonomic studies.

However, the goal of our research is to detect high-level abnormal behaviors, like falling or running away, as well as sleep disorders or changes in life rhythm. The major constraint lies less in the coarseness of data than in the adaptability of deployment in several houses. Therefore, a complete and adaptive coverage on residents' house is difficult to be performed with a fine-grained and complex hardware deployment. Therefore we have chosen to perform behavior detection from a low-cost and scalable deployment, which is our key differentiator from the state of the art.

2 A Design for Large-Scale Deployment

A large-scale deployment requires simple and generic sensors to be installed, rather than sensors attached to specific objects. The purpose being to make as many information as possible emerge from the deployment of simple sensors. It may appear like a blind approach, because sensors are deployed before developing reasoning algorithms. However, features such as "fall detection" are set as requirements first to guide a selection of sensors. This paradigm is interesting for industrial deployments, as it requires neither upstream calibration nor prior knowledge of the environment.

In order to work on AAL, we need real deployments, that provide real data and feedbacks. These deployments take place through the Quality of Life project, focusing on elderly people with mild dementia, who live independently, but receive daily help from a caregiver. Currently, our platform has been deployed in three houses, generating two months of data, and target up to five hundred houses within five years. Each house contains motion sensors in each room. As an input for algorithms, sensor will send "1" when it detects a movement, or "0" after it did not detect any movement for one minute. In each house, we detect an average of 800 events per day. We aim to develop algorithms to detect risks in residents' behaviors, but it is also interesting to perform life logging in order to detect residents' shifts of habits as an important clinical information.

To ensure the relevance of the detected risks, the data analyzed will then be sent to a call center, and monitored by a human supervisor. In case a dangerous behavior is detected by the system, the supervisor will receive an alert. He will then call the resident to check on his safety. In case the resident does not pick up the phone, the supervisor considers the alert relevant and notifies a caregiver or the family. As our project is dealing with the health of people, it would be irresponsible to leave the control solely to a computer system. This is why a human supervisor has to be placed at the end of the line, to keep under control the errors in the reasoning. However, the data processing is still necessary in order to improve the accuracy of the supervisor.

3 Statistical Analysis of Actimetry

3.1 Input Data

The system receives several unrelated data as input, and provides consistent information as output. The selection and calibration of inputs have a major influence on algorithms. Therefore, before running the algorithms themselves, we need to pay a particular attention on input data.

Reducing the deployment costs is a challenge, we have experimented a deployment with low-cost motion sensors, however, when looking at the data, these appeared to have failures, and data is not totally consistent. Noise in data comes from sensor failures, but also from unpredictable events, such as motion sensors triggered by a pet. Hence, the ground truth is not entirely reliable, and we have to deal with the uncertainty in the data. Some solutions involve changes in the hardware design, e.g. sensor collaboration [7], but in order to keep our hardware architecture simple, we have chosen to deal with uncertainty from the software side, and are currently experimenting Dempster-Shafer algorithm [3].

3.2 Data Treatment

In deployment, we receive a sequence of signals, each of them notifies an event in a room at a specific time. Signals are defined as a triple $< time; location; signal >$. In Fig. 1, we receive data from sensors in Location 1 and Location 2, over 900 seconds. This format is useful for data transmission, but too raw to be processed.

$$\begin{array}{rc} & \begin{matrix} t_0 & t_{15} & t_{40} & t_{45} & t_{60} & t_{120} & t_{130} & t_{190} & t_{510} & t_{540} & t_{720} & t_{720} & t_{780} & t_{840} & t_{860} & t_{900} \end{matrix} \\ \begin{matrix} \text{Location} \\ \text{Signal} \end{matrix} & \begin{pmatrix} 1 & 2 & 1 & 2 & 2 & 1 & 2 & 1 & 2 & 2 & 1 & 2 & 2 & 1 & 1 & 1 \\ 1 & 0 & 1 & 1 & 1 & 0 & 1 & 1 & 0 & 0 & 0 & 1 & 0 & 1 & 0 & 0 \end{pmatrix} \end{array}$$

Fig. 1. Initial data stream

First, we split this sequence by location so that each location gets a vector with sequence of pairs $< time; signal >$ (Fig. 2).

$$\begin{array}{rcrc} & \begin{matrix} t_0 & t_{40} & t_{120} & t_{190} & t_{720} & t_{840} & t_{860} & t_{900} \end{matrix} & & \begin{matrix} t_{15} & t_{45} & t_{60} & t_{130} & t_{510} & t_{540} & t_{720} & t_{780} \end{matrix} \\ \text{L1:} & \begin{pmatrix} 1 & 1 & 0 & 1 & 0 & 1 & 0 & 0 \end{pmatrix} & \text{L2:} & \begin{pmatrix} 0 & 1 & 1 & 1 & 0 & 0 & 1 & 0 \end{pmatrix} \end{array}$$

Fig. 2. Data vectors by location

The second step consists in making events as a $i \times j$ matrix, where i is the location, and j the time. $M[x, y]$ shows the status $\{1, 0\}$ at time y, in location x.

We retrieve every event time from each room, and sort the union of them. When a location l contains an event e at time t, then $M[l, t] = e$. We obtain a partially filled matrix, so we need to extrapolate signals (Fig. 3). For each line of this matrix (each location), we apply the Last Observation Carried Forward (LOCF) algorithm [11], to fill the blank with last observation.

$$\begin{array}{rc}
 & \begin{array}{ccccccccccccccc} t_0 & t_{15} & t_{40} & t_{45} & t_{60} & t_{120} & t_{130} & t_{190} & t_{510} & t_{540} & t_{720} & t_{780} & t_{840} & t_{860} & t_{900} \end{array} \\
\begin{array}{r} \text{Location1} \\ \text{Location2} \end{array} & \begin{pmatrix} \textit{1} & 1 & \textit{1} & 1 & 1 & \textit{0} & 0 & \textit{1} & 1 & 1 & \textit{0} & 0 & \textit{1} & \textit{0} & \textit{0} \\ ? & \textit{0} & 0 & \textit{1} & \textit{1} & 1 & \textit{1} & 1 & \textit{0} & \textit{0} & \textit{1} & \textit{0} & 0 & 0 & 0 \end{pmatrix}
\end{array}$$

Fig. 3. Binary situational vector of the house

The next treatment is the most meaningful. Indeed, when a data value is 0, the resident may either be absent, or be present but immobile. So, rather than binary signals, we would like to have Absent/Immobile/Movement signals (A/I/M). To infer such information, with reliable and reasonably complex algorithms, we apply several rules to each element of the matrix. First, when sensor sends "1", it means we have a Movement. Then, when it sends "0", if the resident is in another room, we consider he is Absent from the current room. However, if sensor sends "0" and the resident is not currently seen in another room, we search where he has last moved: if he was previously in the current room, we consider he is still there and Immobile; otherwise, we consider he is Absent from the current room.

$$\begin{array}{rc}
 & \begin{array}{ccccccccccccccc} t_0 & t_{15} & t_{40} & t_{45} & t_{60} & t_{120} & t_{130} & t_{190} & t_{510} & t_{540} & t_{720} & t_{780} & t_{840} & t_{860} & t_{900} \end{array} \\
\begin{array}{r} \text{Location1} \\ \text{Location2} \end{array} & \begin{pmatrix} M & M & M & M & M & A & A & M & M & M & A & A & M & I & I \\ ? & A & A & M & M & M & M & M & A & A & M & I & A & A & A \end{pmatrix}
\end{array}$$

Fig. 4. Situational vector of the house, handling Presence and Absence

Finally, we want to detect when the resident is not alone as this situation should be handled differently. Indeed, although multiuser is harder to handle than single user in terms of context understanding, this is not an issue since we can consider that the resident is in company of a caregiver or family who is taking care of him. Hence, the system has no need to process the alerts anymore. Detecting multiuser is quite easy as we simply need to check whether more than one location shows a presence (Immobile or Movement). However, when the resident is moving from Location A to Location B, there is always a delay when Location A has not yet sent 0, and location B already sent 1. Multiuser will be triggered only if it is still detected after this delay. From our sequence, with a delay defined as 60 seconds, we observe two occurrences of multiuser (Fig. 5).

$$\begin{array}{rccccccccccccccc} & t_0 & t_{15} & t_{40} & t_{45} & t_{60} & t_{120} & t_{130} & t_{190} & t_{510} & t_{540} & t_{720} & t_{780} & t_{840} & t_{860} & t_{900} \\ \text{MultiUser} & \multicolumn{15}{l}{\left(\begin{array}{ccccccccccccccc} 0 & 0 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{array}\right)} \end{array}$$

Fig. 5. Situational vector of the house, with multiuser detection

As the statistical analysis of data performs best on data with regular time granularity, we may also want to get a matrix of the situation at regular time intervals. To perform this transformation, we will again use the LOCF algorithm. First, we need to define the time vector, with a regular interval, for example 60 seconds. We then place all existing data of the original matrix into the new matrix at the closest time sequence. In case several events would go to one element, for instance t_{840} and t_{860}, we need to reduce them to the most significant, so we decided to keep the maximum, considering $M > I > A >?$. However, such case implies a loss of data, thus we must therefore be careful of not using an interval too loose compared to some densities of events. Finally, because we are facing a partially filled matrix, we apply the LOCF algorithm to fill the blanks. We regularized our sequence with an interval of 60 seconds and present the result in Fig. 6.

$$\begin{array}{r} \\ \text{Location1} \\ \text{Location2} \\ \text{MultiUser} \end{array} \begin{array}{c} \begin{array}{cccccccccccccccc} t_0 & t_{60} & t_{120} & t_{180} & t_{240} & t_{300} & t_{360} & t_{420} & t_{480} & t_{540} & t_{600} & t_{660} & t_{720} & t_{780} & t_{840} & t_{900} \end{array} \\ \begin{pmatrix} M & M & A & M & M & M & M & M & M & M & M & M & A & A & M & I \\ M & M & M & M & M & M & M & M & A & A & A & A & M & I & A & A \\ 0 & 1 & 0 & 1 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix} \end{array}$$

Fig. 6. Situational vector of the house at regular intervals of 60 seconds

3.3 Statistical Methods

From previous treatments, we performed a clean vectorization of data, we can now use these vectors to perform a statistical analysis. The first application of statistics for risk detection is to observe variation from routine. When the resident's behavior is too different from his daily routine, we may suspect an alert.

To perform such analysis, we observe the routine on a period of two weeks. Over this period, we perform the processing previously described, and then we calculate the average amount of activity for each 20 seconds times of the day. We consequently obtain an average routine day from 00:00:00 to 23:59:40.

Finally, we analyze one specific day and compute the difference in each room between the expected behavior and the actual behavior of the resident. We also compute the total difference as an indication on how close the resident is from his routine.

4 Results

4.1 Abnormality Detection

From our deployment under the Quality of Life project, we gathered two months of data for three families. Each resident generated an average of 800 events per

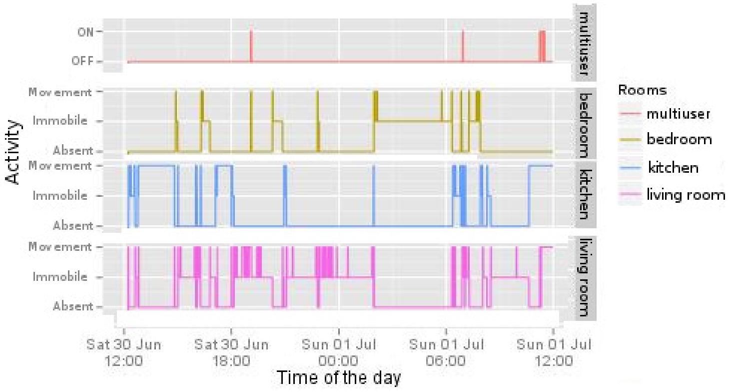

Fig. 7. Graph generated with data analysis algorithms

day. We then applied previously described algorithms, using these data as input. The resulting Fig. 7 contains several meaningful informations, and we may for example observe the night of the resident, between 0:00 and 6:00. We also observe that he spends most of his day in the living room, although he has lunch around 13:00 and probably a dinner at 17:00. When looking at it a human can infer basic life activities and notify major issues.

The following graph in Fig. 8 shows the variation in the resident's behavior within a day, compared to his daily routine.

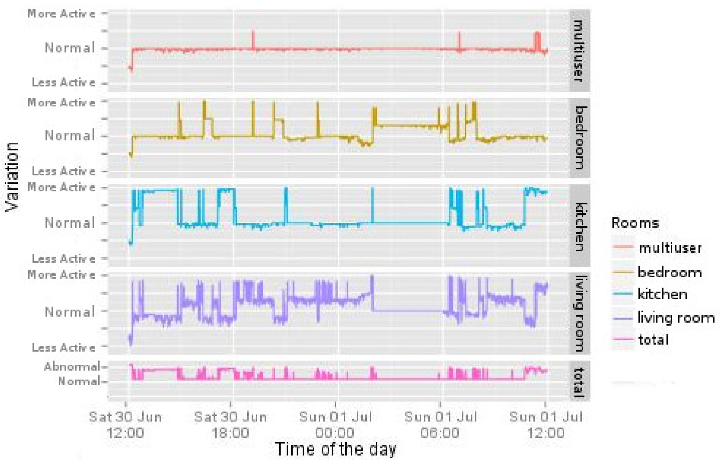

Fig. 8. Statistical analysis of the variation of one specific day over normal routine

If for one room the line is around "Normal", it means than the resident is having a normal behavior, according to his routine. However, a line below "Normal" means that the resident is less active than he usually is at this time, and a line above "Normal" means that he is more active than expected. The graph named "Total" represents the average distance between the resident's behavior and his daily routine in the whole house.

From this graph, we observe that the resident has a regular sleeping rhythm since his night is considered normal. However, his meals generate an abnormal behavior in the kitchen, implying this is not part of his daily routine. More generally, almost every behaviors from the resident are considered as abnormal, as they do not fit the routine.

4.2 Discussions and Perspectives

Our conclusion facing this graph is that a special attention must be payed when we determine a routine. Indeed, an inaccurate routine leads to a bad classification between normal and abnormal behavior. Routine-based statistical analysis first appears to be error-intolerant, but we may highlight the importance of the time granularity in the accuracy of the result. A granularity of 20 minutes would probably provide more error-tolerant results than this granularity of 20 seconds. In that case, processing would need to be adapted, looking for the average of movement on each interval, instead of the maximum level of movement withing each interval as it is currently. Our current results based on a fine-grained time granularity may be useful as an alert filtering, to highlight potential risk. When the resident's behavior fits with calculated routine, we may reasonably consider there is no alert.

Moreover, Hidden Markov Model (HMM) appears as a promising method to make pattern emerge from data [4]. HMM are error-tolerant, and provide results in terms of probability, so that the supervisor would have reliable indicators for detecting alerts.

So far, most observation has been performed on large periods of time, to perform life logging. It would then be interesting to observe shorter periods of time, in order to detect more precise and critical behaviors and events, such as falls.

5 Conclusion

The Quality of Life project demonstrates the validity of a light approach for the deployment of AAL solutions. So far, we detect the presence of resident and multiuser situations from the data provided by motion sensors, so that a human supervisor is able to manually detect activities.

Stating that one is able to detect activities from simple sensor data is a proof of concept. It lets us suppose the existence of patterns in human movements, correlated with daily activities, and visible through actimetry. Such statement is positive for the potential technology transfer of AAL solutions into mainstream eldercare, since simple and low-cost hardware appears to be reliable enough for large-scale deployment. By transferring complexity from hardware to software processing, we provide a scalable and reusable design for AAL solutions. From this statement, we now need to provide automated risk detection, through behavior recognition algorithms.

Acknowledgements. This work is part of the Quality of Life project, funded through the Chaire QoL of the Fondation Mines-Télécom in France.

References

1. Aloulou, H., Mokhtari, M., Tiberghien, T., Biswas, J., Kenneth, L.J.H.: A semantic plug&play based framework for ambient assisted living. In: Donnelly, M., Paggetti, C., Nugent, C., Mokhtari, M. (eds.) ICOST 2012. LNCS, vol. 7251, pp. 165–172. Springer, Heidelberg (2012)
2. Biswas, J., Mokhtari, M., Dong, J.S., Yap, P.: Mild dementia care at home – integrating activity monitoring, user interface plasticity and scenario verification. In: Lee, Y., Bien, Z.Z., Mokhtari, M., Kim, J.T., Park, M., Kim, J., Lee, H., Khalil, I. (eds.) ICOST 2010. LNCS, vol. 6159, pp. 160–170. Springer, Heidelberg (2010), http://dx.doi.org/10.1007/978-3-642-13778-5_20
3. Braun, J.J.: Dempster-shafer theory and bayesian reasoning in multisensor data fusion. In: AeroSense 2000, pp. 255–266. International Society for Optics and Photonics (2000)
4. Bruckner, D., Velik, R.: Behavior learning in dwelling environments with hidden markov models. IEEE Transactions on Industrial Electronics 57(11), 3653–3660 (2010)
5. Couturier, P.: Place de l'actimétrie dans la gestion médicale du sujet âgé fragile. Gérontologie et société (2), 13–23 (2005)
6. Hristova, A., Bernardos, A., Casar, J.: Context-aware services for ambient assisted living: A case-study. In: First International Symposium on Applied Sciences on Biomedical and Communication Technologies, ISABEL 2008, pp. 1–5 (October 2008)
7. Huuskonen, P., Mäntyjärvi, J., Könönen, V.: Collaborative context recognition for mobile devices. In: Handbook of Ambient Intelligence and Smart Environments, pp. 257–280. Springer (2010)
8. Mokhtari, M., Aloulou, H., Tiberghien, T., Biswas, J., Racoceanu, D., Yap, P.: New trends to support independence in persons with mild dementia–a mini-review. Gerontology 58(6), 554–563 (2012)
9. Prince, M., Jackson, J.: World Alzheimer Report 2009. Alzheimer's Disease International (2009)
10. Tiberghien, T., Mokhtari, M., Aloulou, H., Biswas, J., Zhu, J., Lee, V.Y.: Handling user interface plasticity in assistive environment: UbiSMART framework. In: Abdulrazak, B., Giroux, S., Bouchard, B., Pigot, H., Mokhtari, M. (eds.) ICOST 2011. LNCS, vol. 6719, pp. 256–260. Springer, Heidelberg (2011)
11. Zeileis, A., Grothendieck, G., Ryan, J.A., Andrews, F., Zeileis, M.A.: Package 'zoo' (2012)