A Discussion on Fall Detection issues and Its Deployment

When Cloud meets Battery

Samad Barri Khojasteh, José R. Villar and Enrique de la Cal, Víctor M. González
Computer Science Department
University of Oviedo
Oviedo, Spain
{UO267536, villarjose, delacal,vmsuarez}@uniovi.es

Qing Tan, Morteza Kiadi
School of Computing and Information Systems
Athabasca University
Edmonton, Canada
qingt@athabascau.ca
mkiadi2002@yahoo.com

***Abstract*—Fall detection for elder people is a very challenging task that has not been solved yet. In this population, fall detection devices must not introduce extra constraints, like carrying with a belt fixed device or a mobile phone. This paper describes one fall detection approach accomplishing with these constraints. Afterwards, a discussion on some model decisions concerning the computational restrictions according to where the data processing and classification are performed is also included.**

Wearable device; Fall Detection; Neural Network Modelling; Cloud Services

I. INTRODUCTION

In the context of the elder population, the falls represent a main problem "causing a tremendous amount of morbidity, mortality and use of health care services including premature nursing home admissions" [1]. Fall Detection (FD) is a very active research area, with many applications to healthcare, work safety, etc. Commercial products only report up to a 80% of success [2, 3]. There are basically two types of FD systems [4, 5, 6]: context-aware systems - i.e. video systems [7]- and wearable devices [8, 9, 10, 11]. However, the use of wearable devices is crucial because the high percentage of elder people desire to live autonomously in their own house [12].

Wearables-based solutions include, mainly, tri-axial accelerometers (3DACC) either alone or combined with other sensors. Several solutions incorporate more than one sensory element; for instance, Sorvala et al [13] proposed two sets of a 3DACC and a gyroscope, one on the wrist and another on the ankle, detecting the fall events with two defined thresholds. The use of 3DACC and a barometer in a necklace was also reported in [14]; similar approaches have been developed in several commercial products.

The aims of this study are two-fold: on the one hand, a description of a wearable-based solution focused on wrist devices; on the other hand, a discussion on the issues related with the computational design, with the pros and cons of each solution, is included.

II. FALL DETECTION SOLUTIONS IN THE LITERATURE

Abate et al [15, 16] proposed the following scheme to detect a candidate event as a fall event (refer to Figure 1. A time t corresponds to a **peak time** (point 1) if the magnitude of the acceleration a is higher than $th_1=3g$, with $g=9.8\ m/s^2$. After a peak time there must be a period of 2500 ms with relatively calm (no other a value higher than th_1). The **impact end** (point 2) denotes the end of the fall event; it is the last time for which the a value is higher than $th_2=1.5$g. Finally, the **impact start** (point 3) denotes the starting time of the fall event, computed as the time of the first sequence of an $a <= th_3$ ($th_3 = 0.8g$) followed by a value of $a >= th_2$. The impact start must belong to the interval [impact end - 1200 ms, peak time]. If no impact end is found, then it is fixed to peak time plus 1000 ms. If no impact start is found, it is fixed to peak time.

Whenever a peak time is found, the following transformations should be computed:

- Average Absolute Acceleration Magnitude Variation, AAMV=$\sum_{t=is:ie}(a_{t+1}-a_t)/N$, with *is* being the impact start, *ie* the impact end, and N the number of samples in the interval.
- Impact Duration Index, IDI = impact end - impact start. Alternatively, it could be computed as the number of samples.
- Maximum Peak Index, MPI=$\max_{t=is:ie} a_t$.
- Minimum Valley Index, MVI=$\min_{t=is-500:ie} a_t$.
- Peak Duration Index, PDI = peak end - peak start, with peak start defined as the time of the last magnitude sample below $th_{PDI}=1.8g$ occurred before peak time, and peak end is defined as the time of the first magnitude sample below $th_{PDI}=1.8g$ occurred after peak time.
- Activity Ratio Index, ARI, measuring the activity level in an interval of 700 ms centered at the middle time between impact start and impact end. The activity level is calculated as the ratio between the number of samples not in [$th_{ARIlow}=0.85g$, $th_{ARIhigh}=1.3g$] and the total number of samples in the 700 ms interval.
- Free Fall Index, FFI, computed as follows. Firstly, search for an acceleration sample below $th_{FFI}=0.8g$ occurring up to 200 ms before peak time; if found, the sample time represents the end of the interval, otherwise the end of the interval is set 200 ms before

peak time. Secondly, the start of the interval is simply set to 200 ms before its end. FFI is defined as the average acceleration magnitude evaluated within the interval.

- Step Count Index, SCI, measured as the number of peaks in the interval [peak time - 2200, peak time]. SCI is the step count evaluated 2200 ms before peak time. The number of valleys are counted, defining a valley as a region with acceleration magnitude below $th_{SCIlow}=1g$ for at least 80 ms, followed by a magnitude higher than $th_{SCIhigh}=1.6g$ during the next 200 ms. Some ideas on computing the time between peaks [17, 18] were used when implementing this feature.

Evaluating this approach was proposed as follows. The time series of acceleration magnitude values are analyzed searching for peaks that marks where a fall event candidate appears. When it happens to occur, the impact end and the impact start are determined, and thus the remaining features. As long as this fall events are detected when walking or running, for instance, a Neural Network (NN) model is obtained to classify the set of features extracted.

In order to train the NN, the authors made use of an Activities of Daily Living (ADL) and FD dataset, where each file contains a Time Series of 3DACC values corresponding to an activity or to a fall event.

Therefore, each dataset including a fall event or a similar activity -for instance, running can perform similarly to falling- will generate a set of transformation values. Thus, for a dataset file we will detect something similar to a falling, producing a row of the transformations computed for each of the detected events within the file. If nothing is detected within the file, no row is produced. With this strategy, the Abbate et al obtained the training and testing dataset to learn the NN.

The Abbate solution has been modified as follows. As stated in [19, 20], the solutions to this type of problems must be ergonomic: the users must feel comfortable using them. We considered that placing a device on the waist is not comfortable, for instance, it is not valid for women using dresses. When working with elder people, this issue is of main relevance. Therefore, in this study, we placed the wearable device on the wrist. This is not a simple change: the vast majority of the literature reports solutions for FD using waist based solutions. Moreover, according to [21, 22, 23, 24] the calculations should be performed on the smartwatches to extend the battery life by reducing the communications. Therefore, these calculations should be kept as simple as possible.

A second modification is focused on the training of the NN. The original strategy for the generation of the training and testing dataset produced a highly imbalanced dataset: up to 81% of the obtained samples belong to the class FD, while the remaining belong to the different ADL similar to a fall event.

To solve this problem a normalization stage is applied to the generated imbalanced dataset, followed by a SMOTE balancing stage [25]. This balancing stage will produce a 60%(FALL)-40%(no FALL) dataset, which would allow to avoid the over-fitting of the NN models. As usual, there is a compromise between the balancing of the dataset and the synthetic data samples introduced in the dataset.

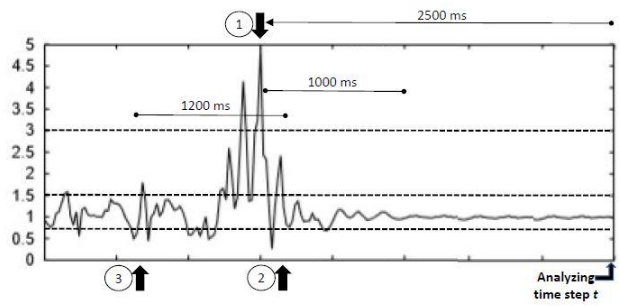

Figure 1. Evolution of the magnitude of the acceleration -y-axis, extracted from [15, 16].

These above mentioned changes have already been studied in [26]. In this research we show the already obtained results to, finally, discuss on the consequences of these results in the different design options.

III. Experiments and Results

A. Data and methods

A ADL and FD dataset is needed to evaluate the adaptation, so it contains time series sample from ADL and for falls. This research made use of the UMA-FALL dataset [27] among the publicly available datasets. This dataset includes data for several participants carrying on with different activities and performing forward, backward and lateral falls. Actually, the falls are not real falls -demonstrative videos have been also published-, but they can represent the initial step for evaluating the adapted solution problem. Interestingly, this dataset includes multiple sensors; therefore, the researcher can evaluate the approach using sensors placed on different parts of the body.

The thresholds used in this study are exactly the same as those mentioned in the original paper. All the code was implemented in R [28] and caret [29]. The parameters for SMOTE were perc.over set to 300 and perc.under set to 200 –i.e. 3 minority class samples are generated per original sample while keeping 2 samples from the majority class-. These parameters produce a balanced dataset that moves from a distribution of 47 samples from the minority class and 200 from the majority class to a 188 minority class versus 282 majority class (40%~60% of balance).

Two modelling types are compared for this task in [26]: a feed forward Neural Network and a C5.0 decision tree, both using the caret package in R. The parameters for the models were, for the NN, size set to 20, decay set to 10^{-3} and maximum number of iterations 500, the absolute and relative tolerances set to $4x10^{-6}$ and 10^{-10}, respectively.

The C5.0 parameters found optimum for this classification problem are cf set to 0.25, bands set to 2, the fuzzy threshold parameter set to TRUE, the number of trials set to 15, and winnow set to FALSE.

B. Performance obtained for the different models

Using 5x2 cross validation (cv) shows the performance of the system with an increase in the number of unseen samples.

TABLE I. 5X2 CROSS VALIDATION RESULTS FOR THE TWO MODELS: FEED-FORWARD NN AND C5.0 DECISION TREE.

	Feed-forward NN					
Fold	**Acc**	**Kp**	**Se**	**Sp**	**Pr**	**G**
1	0,9277	0,8474	0,9645	0,8723	0,9189	0,9415
2	0,9532	0,9033	0,9433	0,9681	0,9779	0,9604
3	0,9149	0,8208	0,9504	0,8617	0,9116	0,9308
4	0,8894	0,7711	0,8936	0,8830	0,9197	0,9066
5	0,8936	0,7834	0,8652	0,9362	0,9531	0,9081
6	0,9447	0,8846	0,9574	0,9255	0,9507	0,9541
7	0,9277	0,8463	0,9787	0,8511	0,9079	0,9426
8	0,9149	0,8214	0,9433	0,8723	0,9172	0,9302
9	0,9149	0,8246	0,9078	0,9255	0,9482	0,9278
10	0,9404	0,8754	0,9574	0,9149	0,9441	0,9507
mean	0,9221	0,8378	0,9362	0,9011	0,9349	0,9353
median	0,9213	0,8354	0,9468	0,8989	0,9319	0,9361
std	0,0209	0,0424	0,0357	0,0382	0,0230	0,0182
	C5.0 decision Tree					
Fold	Acc	Kp	Se	Sp	Pr	G
1	0,9234	0,8376	0,9716	0,8511	0,9073	0,9389
2	0,9234	0,8416	0,9220	0,9255	0,9489	0,9354
3	0,9064	0,8029	0,9433	0,8511	0,9048	0,9238
4	0,9319	0,8546	0,9929	0,8404	0,9032	0,9470
5	0,9617	0,9201	0,9716	0,9468	0,9648	0,9682
6	0,9404	0,8754	0,9575	0,9149	0,9441	0,9507
7	0,9404	0,8741	0,9787	0,8830	0,9262	0,9521
8	0,9234	0,8376	0,9716	0,8511	0,9073	0,9389
9	0,9234	0,8410	0,9291	0,9149	0,9425	0,9357
10	0,9489	0,8929	0,9716	0,9149	0,9448	0,9581
mean	0,9323	0,8578	0,9610	0,8894	0,9294	0,9449
median	0,9277	0,8481	0,9716	0,8989	0,9343	0,9430
std	0,0159	0,0335	0,0227	0,0386	0,0225	0,0129

The results are shown in TABLE I. The boxplots for the statistical measurements Accuracy, Kappa factor, Sensitivity, Specificity, Precision and G are shown in Figure 2.

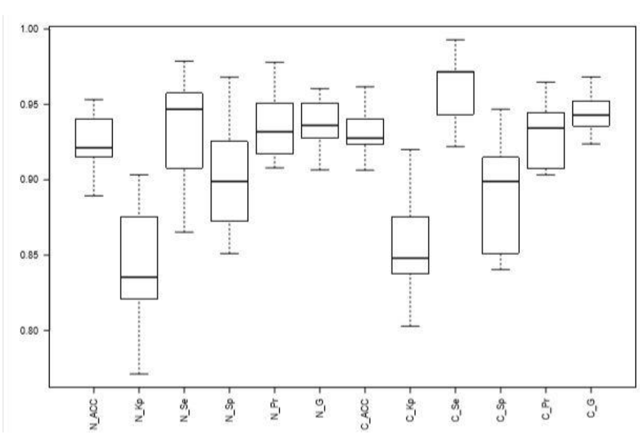

Figure 2. 5x2 cv Boxplot for the different measurements -Accuracy (Acc), Kappa (Kp), Sensitivity (Se) and Specificity (Sp), Precision (Pr) and the geometric mean of the Acc and Pr, G=(Pr*Acc)2-, both for the feed-forward NN (six boxplots to the left, with the N_ prefix) and C5.0 (six boxplots to the right, with the C_ prefix).

With 5x2 cv, the size of the train and test datasets are of similar number of samples, producing a worse training and, what is more interesting, introduces more variability in the test dataset. Therefore, the robustness can be analyzed. Comparing with published results suggests the FD task is not solved yet as the number of false alarms increased unexpectedly. For instance, real world problems must be trained with data from real falls, in the sense that they must be genuine falls. Up to our knowledge, all the published datasets includes real falls but in no real scenario: people falling on a mattress doesn't represent the real scenario. However, the dataset published in [30] represents a promising source of valid data to obtain better models.

IV. DISCUSSION ON THE DESIGN TOPICS

Typical solutions for FD, including the one describe above, are thought to develop a model. However, as seen from the different datasets analyzed, specially, the one presented in [27], a fall can come from a vanish or a collapse or even a hit against some obstacle, either standing still, seated, running or walking. This means that the signal previous to the fall might be of different nature, having a significant effect in the signal sampled from a fall.

Therefore, one interesting solution would be developing FD models for different activity levels -say, scenario-; this is a new idea that has not been studied so far. Consequently, there will be models that would fit for certain scenario independently of the subject. But there will be scenarios for which it will be needed some model tuning to adapt to the current subject.

Can this model tuning be accomplished in a wearable device? It is very unlikely that this can be developed as long as the capabilities restrictions and the battery consumption that might be needed.

So, what to do? We think the solution will end up with a good description of the different scenarios and the development of set of models for each of them. Those models requiring tuning will oblige to store data in the wearable device so this data can be delivered to a service so the tuning can be performed. This data delivering can be performed when charging the device, for instance.

The problem is where to perform the model update. It could be done in the cloud by the central services of the system, such as eHealth cloud system –if it is the responsible of this service-. Nevertheless, we believe a better approach can be applied in this case: using Fog computing.

Now imagine a nursery home with several subjects using this kind of device and reduced cost computational units delivered to the nursery home so they tackle with the data relaying and model update. Using reduced cost devices, like a Raspberry Pi, might ensure enough computation for any home, even up to 100 inhabitants. The advantages are clear: low cost, diminishing of the cloud resources that might be needed for sporadic model updating but that can positively be carried out simultaneously.

Finally, the tuned models are sent back to the wearable device, which will continue testing as usual with the range of tests designed in order to detect a fall.

In our opinion, this is the scenario that the next generation of FD detection methods are to developed in order to obtain scalable, robust and updatable models in FD.

ACKNOWLEDGMENT

This research has been funded by the Spanish Ministry of Science and Innovation, under project MINECO-TIN2014-56967-R and MINECO-TIN2017-84804-R.

REFERENCES

[1] L. Rubenstein, “Falls in older people: epidemiology, risk factors and strategies for prevention,” *Age Ageing,* vol. 35, pp. 37-41, 2006.

[2] Purch.com, *Top Ten Reviews for fall detection of seniors,* 2018.

[3] A. Hakim, M. S. Huq, S. Shanta and B. S. K. K. Ibrahim, “Smartphone Based Data Mining for Fall Detection: Analysis and Design,” *Procedia Computer Science,* vol. 105, pp. 46-51, 2017.

[4] S. S. Khan and JesseHoey, “Review of fall detection techniques: A data availability perspective,” *Medical Engineering and Physics,* vol. 39, pp. 12-22, 2017.

[5] R. Igual, C. Medrano and I. Plaza, “Challenges, issues and trends in fall detection systems,” *BioMedical Engineering OnLine,* vol. 12, 2013.

[6] R. Igual, C. Medrano and I. Plaza, “A comparison of public datasets for acceleration-based fall detection,” *Medical Engineering and Physics,* vol. 37, pp. 870-878, 2015.

[7] S. Zhang, Z. Wei, J. Nie, L. Huang, S. Wang and Z. Li, “A Review on Human Activity Recognition Using Vision-Based Method,” *Journal of Healthcare Engineering,* vol. 2017, 2017.

[8] T. Zhang, J. Wang, L. Xu and P. Liu, “Fall Detection by Wearable Sensor and One-Class SVM Algorithm,” in *Intelligent Computing in Signal Processing and Pattern Recognition,* vol. 345, D. S. Huang, K. Li and G. W. Irwin, Eds., Springer Berlin Heidelberg, 2006, pp. 858-863.

[9] F. Wu, H. Zhao, Y. Zhao and H. Zhong, “Development of a Wearable-Sensor-Based Fall Detection System,” *International Journal of Telemedicine and Applications,* vol. 2015, p. 11, 2015.

[10] J. R. Villar, M. Menéndez, E. Cal, V. M. González and J. Sedano, “Identification of abnormal movements with 3D accelerometer sensors for its application to seizure recognition,” *accepted for publication, International Journal of Applied Logic,* 2016.

[11] J. R. Villar, M. Menéndez, J. Sedano, E. Cal and V. M. González, “Analyzing Accelerometer Data for Epilepsy Episode Recognition,” in *10th International Conference on Soft Computing Models in Industrial and Environmental Applications,* Springer International Publishing, 2015, pp. 39-48.

[12] P. Kumari, L. Mathew and P. Syal, “Increasing trend of wearables and multimodal interface for human activity monitoring: A review,” *Biosensors and Bioelectronics,* vol. 90, pp. 298-307, 4 2017.

[13] A. Sorvala, E. Alasaarela, H. Sorvoja and R. Myllyla, “A two-threshold fall detection algorithm for reducing false alarms,” in *Proceedings of 2012 6th International Symposium on Medical Information and Communication Technology (ISMICT),* 2012.

[14] F. Bianchi, S. J. Redmond, M. R. Narayanan, S. Cerutti and N. H. Lovell, “Barometric Pressure and Triaxial Accelerometry-Based Falls Event Detection,” *IEEE Transactions on Neural Systems and Rehabilitation Engineering,* vol. 18, pp. 619-627, 2010.

[15] S. Abbate, M. Avvenuti, P. Corsini, J. Light and A. Vecchio, “Wireless Sensor Networks: Application - Centric Design,” G. V. Merrett and Y. K. Tan, Eds., Intech, 2010, p. 22.

[16] S. Abbate, M. Avvenuti, F. Bonatesta, G. Cola, P. Corsini and AlessioVecchio, “A smartphone-based fall detection system,” *Pervasive and Mobile Computing,* vol. 8, pp. 883-899, 12 2012.

[17] J. R. Villar, S. González, J. Sedano, C. Chira and J. M. T. Gabriel-Galán, “Improving Human Activity Recognition and its Application in Early Stroke Diagnosis,” *International Journal of Neural Systems,* vol. 25, pp. 1450036-1450055, 2015.

[18] J. R. Villar, P. Vergara, M. Menéndez, E. Cal, V. M. González and J. Sedano, “Generalized models for the classification of abnormal movements in daily life and its applicability to epilepsy convulsion recognition,” *accepted for publication, International Journal of Neural Systems,* 2016.

[19] J. R. Villar, S. González, J. Sedano, C. Chira and J. M. Trejo, “Human Activity Recognition and Feature Selection for Stroke Early Diagnosis,” in *Hybrid Artificial Intelligent Systems,* vol. 8073, J. Pan, M. M. Polycarpou, M. Wozniak, A. P. L. F. Carvalho, H. Quintián and E. Corchado, Eds., Springer Berlin Heidelberg, 2013.

[20] S. González, J. Sedano, J. R. Villar, E. Corchado, {. Herrero and B. Baruque, “Features and Models for Human Activity Recognition,” *Neurocomputing,* vol. in press, 2015.

[21] P. Vergara, J. R. Villar, E. Cal, M. Menéndez and J. Sedano, “Fuzzy rule learning with ACO in epilepsy

crisis identification," in *Proceedings of 11th International Conference on Innovations in Information Technology*, 2015.

[22] P. M. Vergara, E. Cal, J. R. Villar, V. M. González and J. Sedano, "An IoT Platform for Epilepsy Monitoring and Supervising," *Journal of Sensors,* vol. 2017, p. 18, 2017.

[23] S. González, J. R. Villar, J. Sedano, J. Terán, M. L. Alonso-{\'A}lvarez and J. González, "Heuristics for Apnea Episodes Recognition," in *accepted for Proceedings of the International Conference on Soft Computing Models in Industrial and Environmental Applications*, 2015.

[24] S. González, J. R. Villar, J. Sedano and C. Chira, "A preliminary study on early diagnosis of illnesses based on activity disturbances," in *Distributed Computing and Artificial Intelligence, 10th International Conference,* 2013.

[25] N. V. Chawla, K. W. Bowyer, L. O. Hall and W. P. Kegelmeyer, "SMOTE: synthetic minority over-sampling technique," *Journal of artificial intelligence research,* pp. 321-357, 2002.

[26] S. Khojaste, J. Villar, E. de la Cal, V. M. González, J. Sedano and H. R. Yazgan, "Evaluation of a Wrist-based Wearable Fall Detection Method," in *submitted to Hybrid Artificial Intelligent Systems 2018*, Oviedo, 2018.

[27] E. Casilari, J. A. Santoyo-Ramón and J. M. Cano-García, "UMAFall: A Multisensor Dataset for the Research on Automatic Fall Detection," *Procedia Computer Science,* vol. 110, pp. 32-39, 2017.

[28] R Development Core Team, "R: A Language and Environment for Statistical Computing," Vienna, 2008.

[29] M. Kuhn, *The caret package,* 2017.

[30] F. Bagala, C. Becker, A. Cappello, L. Chiari, K. Aminian, J. M. Hausdorff, W. Zijlstra and J. Klenk, "Evaluation of Accelerometer-Based Fall Detection Algorithms on Real-World Falls," *PLoS ONE,* vol. 7, p. e37062, 2012.