Machine learning approach to detect falls on elderly people using sound

Armando Collado-Villaverde, María D. R-Moreno, David F. Barrero, and Daniel Rodriguez

Universidad de Alcalá, Departamento de Automática
Crta. Madrid-Barcelona, Alcalá de Henares, Madrid, Spain

Abstract. One of the most notable consequences of aging is the loss of motor function abilities, making elderly people specially susceptible to falls, which is of the most remarkable concerns in elder care. Thus, several solutions have been proposed to detect falls, however, none of them achieved a great success mainly because of the need of wearing a recording device. In this paper, we study the use of sound to detect fall events. The advantage of this approach over the traditional ones is that the subject does not require to wear additional devices to monitor his or her activities. Here, we apply *machine learning* techniques to process sound simulated the most common type of fall for the elderly, i.e., when the foot collides with an obstacle and the trunk hits the ground before using his/her hands to absorb the fall. The results show that high levels of accuracy can be achieved using only a few signal processing techniques.

Keywords: Fall detection, Feature extraction, Machine Learning, Classification, Supervised Learning, care for the elderly

1 Introduction

Falls are one of the most important health problems for the elderly [1]. They are a significant source of problems mainly because the great damage that they can cause which usually leads to hip injuries. Falls in this group of people have two main sources, the loss of motor functions making them prone to accidental falls and loss of consciousness as a symptom of a hearth attack or other diseases.

The problem has been approached from different angles [2]. During the last few years a wide variety of solutions have been proposed, the majority of them use recorded accelerations from an accelerometer [3] to detect fall events as well as monitoring home rehabilitation [4]. One of the main advantages of this approach is the accelerometer's small size and availability in most modern cell phones, also, they respect the people privacy, unlike other systems. Some systems exploit smartphone's popularity which are used by a lot of elderly people nowadays [5, 6], few others prefer the use of dedicated devices which are usually placed on the trunk [7].

Another interesting approach is the processing of images captured by a camera [8, 9], however, this system has an inherent problem, which is the invasion of privacy. People are usually not willing to have cameras in their private spaces, even if they do not transmit the recorded images. Furthermore, the need to cover every blind spot and all angles is also a problem worth mentioning.

In order to overcome the usage disadvantages of previous devices, we propose the use of a microphone. An important advantage of this approach is that there is no need to wear any device, by getting rid of this need its adoption resistance will be reduced by a large margin, since wearing additional devices to monitor the healthcare of the elders was one the major factor by which other systems were refused. The invasiveness of a microphone in some areas of the house is way lower than having a camera, since with one microphone we can entirely cover an area whereas with cameras we need to cover several blind spots depending on the house distribution.

Many researches have tried to detect events, falls included, by processing sound. Some approaches use human mimicking dolls in the data acquisition tasks to achieve a high accuracy [10]. Other projects use the floor vibrations in addition to the data collected through the microphone achieving a significant accuracy increase [11]. Another interesting approach has its foundation on the difference of the recorded sound depending on the height, in order to exploit that we can use two microphones placed at different heights, considering the difference between both heights of the recorded sound to determine if a fall has happened [12]. Many more projects use sound but only as a secondary source of data, while they use the data obtained using an accelerometer as the main source [13]. We present a Machine Learning (ML) fall detection system which uses only one microphone achieving high accuracy of the classifiers used over the datasets generated under the supervision of professionals in the field.

Other recent works in this field can also be mentioned: ambient assisted living using audio sensing technology [14]; advances on the exploitation of the use of more than one microphone and comparing the sound at different heights [15]; the use of even more microphones, specifically four, in order to detect the 3-D sound source location [16], this approach also uses floor sensors in combination with the microphones to classify the recorded events.

The rest of the paper is structured as follows. First, we address the problem of the data acquisition and its subsequent preprocessing, then we present an explanation of the features analyzed and extracted from the sound waves is described, later on we apply ML algorithms to create a classifier with the obtained data, we also evaluate the selected features and its predictive value. The paper finishes with conclusions and future work.

2 Data acquisition

One of the most critical problems in any ML process is the acquisition of high quality datasets. Recording falls using elderly people was not a reasonable option, since the risk of injuring an actual elder subject was too high considering their fragile physical condition, so we decided to simulate the falls as realistically as possible. A typical fall in an elderly is originated by a trip over, i.e. the collision of a foot with an object while the person is walking, losing the equilibrium and falling over. Then, the trunk bends forward, and given the increased reaction time of elderly, they hits the ground without using their hands for cushioning, resulting in very dangerous falls. We recorded the sound of several falls using a microphone, with a sampling frequency of 44.1kHz.

Using a healthy subject we recorded falls with all possible realism while also trying to avoid risks. We consulted geriatric experts who informed us about the general fall process of elderly people, which involves an increased reaction time, unnatural to the subject we initially planned to use, to overcome that adversity, the experts trained the subject to fall like an older person would.

Therefore, the trained subject was placed on a tatami for safety. Using a thick pad to simulate the obstacle the subject will trip over with as well as serving as a safety method to avoid any damage. The subject started walking and after a couple of steps he would hit the pad with a foot and fell over the pad. The experts supervised all falls recorded, validating only the ones that were similar to the falls that an elderly person would experience. Considering that the recorded data may differ considering which foot hit the pad, the process was repeated to record the same number of falls with each foot.

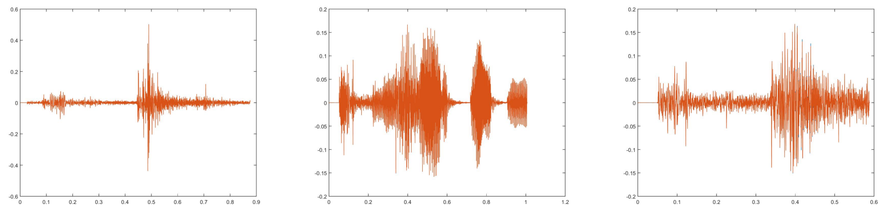
The volunteer simulated 47 falls in total, but the geriatric experts validated only 40, 20 for each feet. Fig. 1 (a) shows the sound wave of a recorded fall. We can easily appreciate when the fall starts, as the first major variation in the wave, when the foot collides with the pad and the moment when the trunk falls and hits the pad as the biggest peak of the wave.

In order to create a classifier using ML algorithms we also need to use sound where no falls happen, that sound will be compared to the recorded sound from the falls. We extracted the sound from two different sources. The first one was a conversation, which had several moments without sound in between each phrase from the speakers. The second one were war sounds extracted from an action video game, this clip had fewer silent moments since there were constant background sounds, the first wave has much more time between each sounds while the second one has its sounds much more closer to each other.

3 Data preprocessing

Data needs some preprocessing in order to apply the machine learning classifier. First of all, we need to cut the recorded sound from the falls, since it contains a lot of absence of sound before and after the fall, otherwise we would consider the absence of sound as falls.

Once the cut was performed, after some initial tests we faced another problem. The classifier exploited the high amount of silence contained in the fall, so they were easy to differentiate from the conversation and other sounds. To avoid this problem we mixed the sound from the falls, adding either the conversation or the game sounds as background. In the Figures 1 (b) and (c) we can see the wave sounds of the fall after the mix, they are much more similar to the waves



(a) Example of the sound wave of a fall after the cut.

(b) Mix of the fall sound with the conversation

(c) Mix of the fall sound with the war sounds

Fig. 1: Original and mixed fall sound waves.

with which we are going to compare them. We also had to cut the '*no fall*' sounds so their duration would similar to the fall sounds.

An important issue about dataset is that it is unbalanced. Since falls are hard to simulate, there were much more data coming from conversations and game sounds than from simulated falls. To face this issue we undersampled the 'not-fall' class, getting the same number of instances for each class.

4 Feature extraction by processing the signal

We need to process the recorded sound. Using different processing techniques we were able to extract 10 features from each audio signal. As our first step we need to process the signal, we will separate each audio clip into portions with less duration, we will use frames of 2048 samples, if we consider the sampling frequency of 44.1 kHz, each frame will contain 46.3ms of sound. This duration has not been chosen arbitrarily, because of the aleatory and non-stationary nature of the sound we need to use small frames to analyze the spectrum created by the sound properly. By using such an small frame, we will obtain a practically stationary signal, facilitating the posterior feature extraction. In addition, by separating the sound into frames, we will obtain two features per equation applied, since we can calculate the mean and standard deviation of all the frames generated from the original sound wave.

The feature extraction is a mathematical process which can characterize an audio signal, we will group our features into two main groups: (i) temporal and (ii) spectral features. When analyzing *temporal features* we will consider frequency and level (decibels), these features include the energy of the signal and the zero-crossing feature. *Spectral features* which will be extracted by computing the Fast Fourier Transform, which will consist on the spectral centroid, the Rolloff factor and the spectral flux, all of them apply mathematical equations to the discrete-time signal $x[n]$ of each sound frame:

- **Energy of the signal** (E_k): the energy is calculated as the squared mod of the window as shown in the equation (1), it basically informs about the strength of the signal. In addition, we introduced a threshold of 9.2376^{-6} Joules, which eliminates the windows whose energy does not surpass it.

$$E_k = \sum_{n=0}^{N} |x[n]|^2 \tag{1}$$

- **Zero-crossing** (Z_k)**:** this parameter informs about the amount of noise contained in the signal by counting the amount of times where the sign of the signal changes from positive to negative. The higher its value, the more noisy the signal is. We will use the equation (2) to count the amount of times each window changes its sign, as shown in the equation we need to halve our result since we are only interested in the changes from positive to negative.

$$Z_k = \frac{1}{2} \cdot \sum_{n=0}^{N-1} |sign(x[n]) - sign(x[n+1])| \tag{2}$$

We used the following spectral features:

- **Spectral centroid** (C_k)**:** the centroid is the middle point of the spectrum, the frequency that divides the spectrum into two equal parts, sound signals formed by mainly high frequency samples have higher centroid values. The centroid is calculated by the average frequency weighted by amplitudes, divided by the sum of the amplitudes, as shown in the equation (3).

$$C_k = \frac{\sum_{n=0}^{N} F_k[n] \cdot n}{\sum_{n=0}^{N} F_k[n]} \tag{3}$$

Where $F_k[n]$ is the amplitude of the fast Fourier transform of the n frequency applied to the k window.

- **Rolloff factor** R**:** R frequency below which is 85% of the spectrum, this feature informs about the form of the signal spectrum as a whole, in order to calculate it we will use the equation (4).

$$\sum_{n=0}^{R} F_k[n] = 0.85 \cdot \sum_{n=0}^{N} F_k[n] \tag{4}$$

- **Spectral flux** (F_k)**:** this feature indicates how quickly the energy of the spectrum changes calculated by comparing the the squared difference of the module of spectrum for one frame against the power spectrum from the previous frame, as illustrated in the equation (5).

$$F_k = \sum_{n=0}^{N} (F_k[n] - F_{k-1}[n])^2 \tag{5}$$

Where $F_k[n]$ and $F_{k-1}[n]$ are the module of the fast Fourier transform of the k and k-1 window.

Table 1: Features used to detect fall sounds.

Energy Mean	Energy Standard Deviation (Std)
Number of Zeros Mean	Number of Zeros Std
Spectral Flux Mean	Spectral Flux Std
Roll off Factor Mean	Roll off Factor Std
Spectral centroid Mean	Spectral Centroid Std

Table 2: Evaluation of the classifiers using the falls sound mixed with the conversation as 'falls' and sounds from another conversation as 'non-falls' dataset.

Parameter	*Class*	C4.5	*1-NN*	*Log Reg*	*Naïve Bayes*	*PART*	*Random Forest*	*SVM*
Precision	Fall	78.9%	77.5%	79.5%	79.1%	86.5%	87.5%	83.3%
Recall	Fall	75.0%	77.5%	87.5%	85.0%	80.0%	87.5%	87.5%
F-Measure	Fall	76.9%	77.5%	83.3%	81.9%	83.1%	87.5%	85.4%
Precision	NonFall	76.2%	77.5%	86.1%	83.8%	81.4%	87.5%	86.8%
Recall	NonFall	80.0%	77.5%	77.5%	77.5%	87.5%	87.5%	82.5%
F-Measure	NonFall	78.0%	77.5%	81.6%	80.5%	84.3%	87.5%	84.66%
Overall	Both	77.5%	77.5%	82.5%	81.25%	83.75%	**87.5%**	85.0%

Once we have obtained all of the features from all the frames created by dividing the original sound wave, we will calculate the mean and standard deviation of each feature considering the values obtained by processing each frame, in addition, each sample was labeled as 'fall' or 'not-fall' depending of the analyzed sound, the combination of the extracted features and the class label will serve as input for the classifier.

5 Detection of fall events

Detection of fall events can be summarized as a binary classification problem: Considering the extracted features we classify each sound sample as '*fall*' or '*non-fall*', thus we used some classical classification algorithms implemented in Weka such as C4.5 (J48), 1-NN, Logistic regression, Naïve Bayes, PART, Random Forest and Support Vector Machines (SVMs). Some of them did not achieve a high performance, but we included them for comparison purposes. The performance of these algorithms can be seen as a benchmark given their high performance without requiring an excessive training or evaluation time. The features that feed the classifiers are summarized in Table 1. The evaluation of the classifiers was carried out using 10-fold cross-validation.

We performed three experiments, we compared the recorded sound from the falls with war ambient sounds and a recorded conversation and later on we mixed the fall sounds with the war and conversation sounds using the two last ones as background sound. First, we compared the fall sounds with one conversation sounds representing the 'fall' class, and sounds from other conversation as the 'non-fall' class. The performance of the previously listed algorithms using this

Table 3: Evaluation of the classifiers using the falls sound mixed with the war as 'falls' and other war sounds as 'non-falls' dataset.

Parameter	*Class*	C4.5	*1-NN*	*Log Reg*	*Naïve Bayes*	*PART*	*Random Forest*	*SVM*
Precision	Fall	87.2%	87.8%	97.4%	91.9%	82.2%	87.5%	94.3%
Recall	Fall	85.0%	90.0%	95.0%	85.0%	92.5%	87.5%	82.5%
F-Measure	Fall	86.1%	88.9%	96.2%	88.3%	87.1%	87.5%	88.0%
Precision	NonFall	86.4%	90.5%	95.5%	87.0%	92.1%	88.4%	85.4%
Recall	NonFall	88.4%	88.4%	97.7%	93.0%	81.4%	88.4%	95.3%
F-Measure	NonFall	87.4%	89.4%	96.6%	89.9%	86.4%	88.4%	0.91%
Overall	Both	86.7%	89.1%	**96.3%**	89.1%	86.7%	87.9%	89.15%

Table 4: Evaluation of the classifiers using the dataset mixing the falls sounds with the war and conversation as 'falls' and other war and conversation sounds as 'not falls'.

Parameter	*Class*	C4.5	*1-NN*	*Log Reg*	*Naïve Bayes*	*PART*	*Random Forest*	*SVM*
Precision	Fall	82.3%	83.6%	85.4%	72.7%	89.6%	87.2%	76.5%
Recall	Fall	81.3%	76.3%	87.5%	90.0%	75.0%	85.0%	77.5%
F-Measure	Fall	81.8%	79.7%	86.4%	80.4%	81.6%	86.1%	77.0%
Precision	NonFall	82.1%	78.9%	87.7%	87.5%	79.2%	85.9%	78.0%
Recall	NonFall	83.1%	85.5%	85.5%	67.5%	91.6%	88.0%	77.1%
F-Measure	NonFall	82.6%	82.1%	86.6%	76.2%	84.9%	86.9%	77.6%
Overall	Both	82.2%	80.9%	**86.5%**	78.5%	83.4%	**86.5%**	77.3%

dataset is summarized Table 2 including the precision, recall and F-measure for each class respectively and the overall performance which is quite high, achieving in all cases more than 75%, the highest performance algorithm is the Random Forest, in which we used 100 iterations of the algorithm.

In the second case we mixed the fall sounds with fragments from the war sounds clip representing the 'fall' class, while fragments which were not used previously in the mixing part represent the 'not-fall' class. We analyzed the dataset using the same algorithms that we used previously, we can see the performance in the Table 3. The performance in this case is higher than in the previous one, mainly because the sounds in the conversation are more intermittent, similar to the falls, while the war sounds are more continuous, contrary to the falls case. We can highlight the Logistic Regression algorithm whose performance is the highest among all the analyzed algorithms, achieving an overall performance of 96.3%.

Finally we mixed the two datasets that were analyzed individually previously to get a better generalization. Performance will be a bit lower than the classifier created mixing the fall sounds with the war ones. The required time to create the classifiers and the time to perform the classifications will also be higher. Table 4 shows the performance is higher than the conversation only dataset, and lower than the war only dataset. All the algorithms have a similar performance, the Logistic Regression and Random forest are still the best performing algo-

Table 5: Ranking of the worth of information gained per attribute with respect to the class.

% Inf.	Attrib.	% Inf.	Attrib.
0.441	Centroid std	0.136	Number of Zeros std
0.291	Energy std	0.119	Flux mean
0.186	Flux std	0	Centroid mean
0.168	Rolloff std	0	Rolloff mean
0.146	Energy mean	0	Number of Zeros mean

rithms but the difference is not as high as it was previously, although they have the same overall performance. The logistic regression missclassifies less '*falls*' instances, classifying them as '*non falls*' and the main source of mistakes is the missclasification of '*non falls*' instances, while in the random forest algorithm it goes the other way. Finally, the worst performing algorithm is Naïve Bayes.

6 Overview of attributes classification power

Although the number of used attributes is low, we estimated the predictive power of each attribute in order to determine which features perform the best. We ranked the attributes using the Information Gained per Attribute Evaluator/ which evaluates the worth of an attribute by measuring the information gain with respect to the class, afterwards, we ranked them according to their individual evaluations.

Following Table 5, the best attributes are the ones which measure the standard deviation of the analyzed features. The most powerful one is the standard deviation of the spectral centroids, if we consider that, this feature gives us information about the overall shape of the sound wave its value is logical; the second best attribute is the standard deviation of the energy of the signal. After this two features, the other ones have a significant fewer predictive power, if we consider the group that the two top features belong to, we can deduce that both temporal and spectral features are important and that we can not exclude either one of them.

Table 6 shows the performance of the algorithms if we remove the three attributes less correlated to the class according to the previous analysis (see Table 5). The 'Impact of the selection' row calculates the difference in *overall performance* with respect to the Table 4. We can observe that some algorithms, 1-NN and Random Forest, have their performance intact, since they did not use the removed attributes, however, the PART and C4.5 algorithms experience a minor improvement, increasing their overall performance by 1.2% and 1.8% respectively. The performance of the Logistic regression and Naïve Bayes algorithms has suffered a loss of performance by 3% and 0.6% respectively.

It can be concluded that we will need to remove those algorithms with a poor contribution of information if we plan to use either the C4.5 or PART algorithms. It is advisable to remove them if the plan to use the 1-NN or Random Forest

Table 6: Evaluation of the classifiers using the dataset mixing the two previous ones after removing the less informative attributes.

Parameter	*Class*	*C4.5*	*1-NN*	*Log Reg*	*Naïve Bayes*	*PART*	*Random Forest*
Precision	Fall	85.5%	82.7%	81.2%	72.4%	92.3%	87.2%
Recall	Fall	81.3%	77.5%	86.3%	88.8%	75.0%	85.0%
F-Measure	Fall	83.3%	80.0%	83.6%	79.8%	82.8%	86.1%
Precision	NonFall	82.8%	79.5%	85.9%	79.6%	79.2%	85.9%
Recall	NonFall	86.7%	84.3%	80.7%	67.5%	94.0%	88.0%
F-Measure	NonFall	84.7%	81.9%	83.2%	75.7%	86.2%	86.9%
Overall	Both	84.0%	80.9%	83.4%	77.9%	**84.6%**	**86.5%**
Impact of the selection		**+1.8%**	±0%	**-3.0%**	-0.6%	**+1.2%**	±0%

algorithms, since we will relive some computational load by reducing the number of attributes that need to be evaluated. However, if we are planning to use either the Naïve Bayes or Logistic regression algorithms, we must not remove them since it would lead to a loss of performance.

7 Conclusions and future work

In this paper we described a ML application to detect falls by analyzing the produced sound. The aim is to implement a fall detection system oriented to the care of the elderly. This population group is prone to suffer the analyzed type of fall that we simulated and recorded. Data, along with recordings from a conversation and war ambient sound were divided into window frames and then five features were extracted from each window, allowing us to calculate the mean and standard deviation of all the windows in each sound sample. Those served as input for the ML classifiers that we used to create the classifiers.

In the near future we expect to expand the detection with new kind of falls and new features to improve the classifiers accuracy.

Acknowledgements

The authors thank the contribution of Isabel Pascual Benito, Francisco López Martínez and Helena Hernández Martínez, from Department of Nursing and Physiotherapy of the University of Alcalá, for their help designing and supervising the simulated falls procedure as well as Diego López Pajares and Enrique Alexandre Cortizo for their help regarding the signal processing tasks. This work is supported by UAH (2015/00297/001), JCLM (PEII-2014-015-A) and EphemeCH (TIN2014-56494-C4-4-P) Spanish Ministry of Economy and Competitivity projects.

References

1. Sadigh, S., Reimers, A., Andersson, R., Laflamme, L.: Falls and fall-related injuries among the elderly: a survey of residential-care facilities in a swedish municipality. Journal of community health **29** (2004) 129–140
2. Blasco, J., Chen, T.M., Tapiador, J., Peris-Lopez, P.: A survey of wearable biometric recognition systems. ACM Comput. Surv. **49** (2016) 43:1–43:35
3. Noury, N., Fleury, a., Rumeau, P., a.K. Bourke, Laighin, G.O., Rialle, V., Lundy, J.E.: Fall detection - Principles and Methods. 2007 29th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (2007) 1663–1666
4. Tao, Y., Hu, H., Zhou, H.: Integration of vision and inertial sensors for 3d arm motion tracking in home-based rehabilitation. The International Journal of Robotics Research **26** (2007) 607–624
5. Luštrek, M., Kaluža, B.: Fall Detection and Activity Recognition with Machine Learning. Informatica **33** (2008) 205–212
6. Albert, M.V., Kording, K., Herrmann, M., Jayaraman, A.: Fall classification by machine learning using mobile phones. PLoS ONE **7** (2012) 3–8
7. Gibson, R.M., Amira, A., Ramzan, N., Casaseca-de-la higuera, P., Pervez, Z.: Multiple comparator classifier framework for accelerometer-based fall detection and diagnostic. Applied Soft Computing Journal **39** (2016) 94–103
8. Miaou, S.G., Sung, P.H., Huang, C.Y.: A customized human fall detection system using omni-camera images and personal information. Conference Proceedings - 1st Transdisciplinary Conference on Distributed Diagnosis and Home Healthcare, D2H2 2006 **2006** (2006) 39–42
9. Auvinet, E., Multon, F., Saint-Arnaud, A., Rousseau, J., Meunier, J.: Fall detection with multiple cameras: an occlusion-resistant method based on 3-D silhouette vertical distribution. IEEE transactions on information technology in biomedicine **15** (2011) 290–300
10. Zigel, Y., Litvak, D., Gannot, I.: A method for automatic fall detection of elderly people using floor vibrations and soundProof of concept on human mimicking doll falls. IEEE Transactions on Biomedical Engineering **56** (2009) 2858–2867
11. Litvak, D., Zigel, Y., Gannot, I.: Fall detection of elderly through floor vibrations and sound. Conference proceedings : ... Annual International Conference of the IEEE Engineering in Medicine and Biology Society. IEEE Engineering in Medicine and Biology Society. Annual Conference **2008** (2008) 4632–5
12. Popescu, M., Member, S., Li, Y., Skubic, M., Rantz, M.: Information to Reduce the False Alarm Rate. (2008) 4628–4631
13. Doukas, C., Maglogiannis, I.: Advanced patient or elder fall detection based on movement and sound data. Proceedings of the 2nd International Conference on Pervasive Computing Technologies for Healthcare 2008, PervasiveHealth (2008) 103–107
14. Vacher, M., Portet, F., Fleury, A., Noury, N.: Development of audio sensing technology for ambient assisted living: Applications and challenges. Digital Advances in Medicine, E-Health, and Communication Technologies (2013) 148
15. Li, Y., Ho, K., Popescu, M.: A microphone array system for automatic fall detection. IEEE Transactions on Biomedical Engineering **59** (2012) 1291–1301
16. Chaudhuri, S., Thompson, H., Demiris, G.: Fall detection devices and their use with older adults: a systematic review. Journal of geriatric physical therapy (2001) **37** (2014) 178