Indian Journal of Science and Technology, *Vol 10(5), DOI: 10.17485/ijst/2017/v10i5/111274, February 2017*
ISSN (Print) : 0974-6846
ISSN (Online) : 0974-5645

Smart Phone Based Fall Detection using Auto Regression Modeling in a Non-Restrictive Setting

Maria Panhwar, Syed M. Shehram Shah, Syed M. Zafi S. Shah, Syed M. Z. Abbas Shah and Bhawani Shankar Chowdhry

Department of Software Engineering, Telecommunication Engineering, Electronic Engineering, Institute of Information and Communication Technologies (IICT), Mehran University of Engineering and Technology, Pakistan; mariapanhwar76@gmail.com, shehram.shah@faculty.muet.edu.pk, zafisherhan.shah@faculty.muet.edu.pk, zaigham.shah@faculty.muet.edu.pk, bhawani.chowdhry@faculty.muet.edu.pk

Abstract

Fall detection is an important aspect of the field of accident prevention, ambient assisted living as well as care of the elderly. To address this issue, researchers have employed several approaches including vision based systems, setups that require deployment in a special environment and inertial sensors. Inertial sensors have the advantage of being deployable in mobile systems such as wearable devices and smart phones. An important consideration in using inertial sensors for fall detection is the need to develop techniques that would work without enforcing positional requirements of the sensor device. This paper presents a method for the detection of falls using inertial sensors readings of the smart phone, a tri-axial accelerometer, tri-axial gyroscope and orientation data. We consider inertial sensor data for two falls and three activities of daily living. Using Auto-Regressive (AR) modeling to characterize the measurements from the sensors, we compare Support Vector Machines (SVM) and Neural Networks for use in classifying between these five events. Results indicate that the Neural Network provides better classification accuracy compared to SVM for the purpose of differentiating between falls and the activities of daily living.

Keywords: Fall Detection, Inertial Sensor, Machine Learning, Mobifall

1. Introduction

Falls can be defined as an unintentional or sudden going down of a body on the ground or towards a lower level. The fall is a main cause of the injuries of the elderly person and the chances of these falls increases with the increase in age. The Center for Disease Control and Prevention states that falls cost the US economy $30 billion per year in medical care costs for older adults[1] with a person being treated in the emergency department of the hospital every 17 seconds with injuries due to a fall[2].

To prevent these losses in health and money, different types of technologies have been employed to detect falls while adapting to a variety of application scenarios. Generally, two types of fall detection schemes exist, vision based approaches which use image devices for detecting the falls and other one is sensor based approach which uses sensors placed in the area where fall has to be made and/or worn on the body. The second type of detection method in which sensors are worn on the body have the advantage that they are mobile and can be attached to the person who needs to be monitored. Such systems have historically dedicated hardware[3] that contain sensors such as accelerometers and gyroscopes. However with the ubiquity of smartphones which have these built in to them, smartphones have been increasingly used for the purpose of fall detection. The paper is organized as follows, Section 2 provides a summary of previous work in this area, Section 3 describes the Mobifall and MobiAct datasets, Section 4 describes the methodology, Section 5

**Author for correspondence*

discusses the results and the paper is concluded in Section 6.

2. Literature Review

Since the issue of fall detection has been of great interest to field of assisted living and accident prevention. There have several approaches to address this task which include techniques that use imaging or a sensor mesh which are used for fall detection within a limited environment and others which are mobile and can be attached the to subject.

The authors in[4] use radio tomographic imaging[5] to detect falls within a room. They make use of a two-level array of RF sensor nodes which were placed around the room perimeter while the shadowing losses in the signals exchanged between sensors were measured to form an estimate of the horizontal and vertical position of the subject. The authors in[6-8] proposed methods for robust fall detection through computer vision. In[7], the authors use an RGB depth camera to track key joints of the human body utilizing a tree algorithm for key joint extraction. This scheme can is shown to work in a dark room as well as low on computational cost. In[8], Microsoft Kinect[9] is used to capture videos of elderly people in homes. They characterize the vertical state of a person for depth image frames segmented from the ground events and with decision trees being used to determine whether a fall has occurred or not. These techniques have been shown to work in a limited environment only and require specialist hardware for their use. Rougier et al.[6] detect falls by tracking the trajectory of head movement of a person. To work, this method requires initialization of the head position. However, this has been shown to have high complexity and a high misjudgement rate.

Another set of approaches for fall detection systems that can be deployed use inertial sensors. T.R. Hansen et al.[10] presented a project to design Information Technology For Assisted Living At Home (ITALH). They incorporate a 3-axis accelerometer in a system that is worn by the user. An embedded sensor is used to analyze the data with Bluetooth wireless connection to create a system to develop and evaluate of fall detection algorithms using Matlab programs. Ge Wu and Shuwan Xue[11] introduce a portable technique for detection of falls. They make use of an inertial sensor and a data logger. Falls are detected by using a threshold applied on the the inertial frame vertical velocity. The authors in[12] utilize a triaxial accelerometer put on the thigh and trunk of a person to differentiate between different falls and daily living activities through a thresholding approach. The authors in[13] use two inertial-sensors, an accelerometer and a gyroscope in smart phones to compare various fall detection algorithms.

As mentioned before, to allow fall detection in practical scenarios, it is imperative that the apparatus used does not put restriction on the movement of the subject whose fall is to be detected. Most of the previous methods of fall detection have used imaging devices or an array of sensors which can only be deployed in a limited area where the fall detection has to be made. Moreover, approaches using inertial sensors require them to be placed in specific ways thus making the devices inconvenient to use.

3. Mobifall and Mobiact Datasets

The Mobifall and MobiAct datasets were developed by the Biomedical Informatics & Health Laboratory of the Technological Educational Institute of Crete[13] for the development of algorithms for detecting of Activities of Daily Living and falls. The MobiFall dataset contains data from 35 volunteers (age range: 22 to 47) performing different activities of daily living and falls. In the dataset, Data from the accelerometer and gyroscope sensors (plus orientation data) were recorded by using a Samsung Galaxy S3 (having a 3D accelerometer and gyroscope). The device was placed in the pocket of the subject's trouser with no orientation restriction, thereby posing no pre-specified requirements on device placement. Recently a dataset called the MobiAct dataset[14] was developed based on the Mobifall dataset that provides trials for a total of nine activities of daily living and four falls for 57 subjects and over 2500 trials in total. Since the MobiAct dataset provides similar data acquisition environment, we use the MobiAct dataset to develop and test the algorithm in this work. In this work we have considered two falls, Forward Lying (FOL) and Back Sitting Chair (BSC) and three activities of daily living which are Standing (STD), Walking (WAL) and Sitting (SCH). Table 1 lists the details of falls and daily living activities that have been considered in this work.

4. Methodology

The proposed work follows three stages as is followed in a typical classification scenario, preprocessing, feature extraction and classification. Pre-processing consists of

Table 1. Considered falls and activities[14]

Code	Activity	Trials	Duration	Description
FOL	Forward lying	3	6 sec	Fall forward from standing, use of hands to dampen fall
BSC	Back sitting chair\	3	6 sec	Fall backward while trying to sit on a chair
SCH	sitting chair	3	10 sec	Sitting on the chair
STD	Standing	1	5 min	Standing with suitable movement
WAL	walking	1	5 min	Normal walking

down sampling and windowing which is followed by feature extraction consisting of Autoregressive modeling and lastly, classification is performed using Support Vector Machines (SVM) and a Pattern Net Neural Network as shown in Figure 1. We chose SVM due to its relatively cheap computational complexity[15] and speed and Neural networks since they have been found to be performing well in previous works[14].

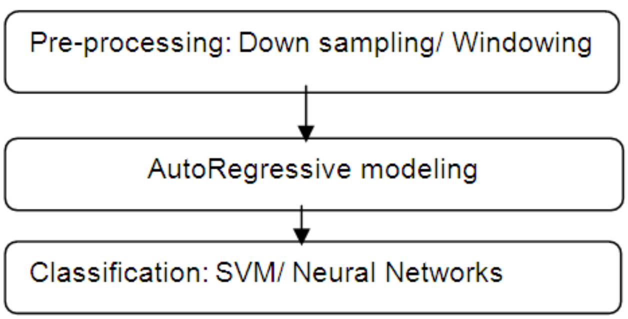

Figure 1. Flow chart of proposed work.

4.1 Pre-Processing

In this stage one or two operations are performed depending on the signal being considered.

4.1.1 Re-Sampling

The Mobifall and MobiAct datasets contain sensor readings which vary over trials[13] (due to the underlying processes open in the Android phone). Therefore, in order to process them we needed to resample them to a suitable frequency. The signals were resampled to a frequency of 20Hz as was carried out in[14]. A frequency of 20Hz has been found to be sufficient to capture the activities and falls suitably.

4.1.2 Windowing

The trials for the activities of Walking and Standing consist of data for 5 minutes. Therefore, appropriate windowing of the signal needs to be carried out before it is passed on to the feature extraction stage. We have chosen to segment the activities in to windows of 10s each. A similar approach has been followed in[14].

4.2 Feature Extraction

An advantage of the dataset is the non-specific directional setting of the smart phone while data is being collected. This requires that a suitable feature extraction process be used that does not take in to consideration the relativeness between the readings of the different axes of the sensors. Typically used feature vectors are formed using statistical and other mathematical measurements on the sensor readings[16] which are not suitable for data of such type. To ensure that the feature extraction process is independent of axes, we use Auto-Regressive (AR) modeling for each sensor reading. AR modeling has been used in various fields such as ECG disease detection[17].

An Auto-Regressive Model is a linear prediction formula that predicts an output y_t of some system taking in to account previous inputs as given in Equation 1.

$$y(t) = \sum_{i=1}^{m} a(i).y(t-i) + \varepsilon\,(t) \qquad (1)$$

Where a(i) represent coefficients of the computed AR model, the series under investigation is represented by y(t-i), output of uncorrelated errors by ε(t) and 'm' is the order of model which indicates the number of past samples which have been used to estimate the present value of signal. The feature vector is formed by combining the AR coefficients of the AR model of each sensor reading. An AR model of order 3 was computed for each signal segment. With three sensor reading for each of the three Cartesian axes, the feature vector for each activity and fall consisted of 36 values. The Yule-Walker method[18] was used to compute the AR model coefficients.

4.3 Classification

Classification has been performed using two different algorithms, the SVM and a two layer feed forward neural network.

4.3.1 Support Vector Machines

SVM is a learning machine is used to analyze data and is used for classification and regression analysis. It is a linear classifier which tries to fit a line to separate two classes of data. In order to use SVM to classify data from the sensors for activities and falls, we use a Gaussian Kernel (Equation 2).

$$k(x,y) = exp^{\left(-\frac{||x-y||^2}{2\sigma^2}\right)} \quad (2)$$

Where 'y' is the center and σ is the x spread of the blob and k is the value of the Gaussian for a specific x, y, σ.

4.3.2 Neural Network

Neural networks are used to extract patterns and trends from complex data[19]. In this work, we have a two layer feed-forward Neural Network shown in Figure 2 which consists of a hidden layer with a log sigmoid activation function as shown in Equation 3[20].

$$a = \frac{1}{1+e^{-n}} \quad (3)$$

Where 'a' is the output and n is the input matrix. The output layer consists of a Softmax transfer function in the output layer whose behaviour is described by Equation 4[21].

$$\mu_k = \frac{e^{(\eta_k)}}{1+\sum_j e^{(\eta_k)}} \quad (4)$$

Where η_k is a k-sized vector, μ_k is also a k-sized vector having values between 0 and 1 that sum up to 1. Furthermore the hidden layer (Sigmoid activation function) has 23 neurons and the output layer (softmax activation function) with 5 neurons as shown in Table 2.

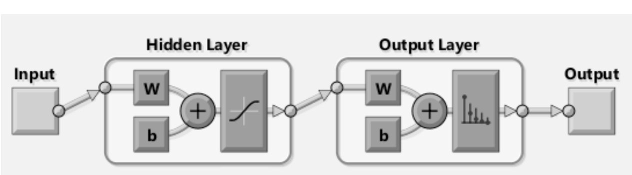

Figure 2. Two layer feed-forward Neural Network.

5. Results and Discussion

Table 3 lists the number of segments of each activity and fall which have been used for training and testing for both algorithms. Trials of some activities and falls were not performed by some of the subjects which results in a non-uniform set of samples. Furthermore, a subset of the total possible samples were considered for the activities of walking and standing.

Table 2. Neural Network details

Input layer	Hidden Layer (Log-sigmoid)	Output Layer (Softmax)
36 lines	23 neurons	5 neurons

Table 3. Number of samples for each activity used for training and testing

Fall/ Activity	Total number of sensor samples	Number of sensor samples used for training	Number of sensor samples used for testing
BSC	160	112	48
FOL	162	113	49
SCH	138	97	41
STD	257	180	77
WAL	261	183	78

The sensor samples used for training were chosen at random from the pool of the considered values and a SVM with a Gaussian kernel was trained to classify between the five categories. The remaining samples were then given as inputs to the trained model to test its performance. Table 4 shows the confusion matrix for the classification result for the SVM. It can be observed that the SVM was able to classify the activities of STD and WAL with 100% accuracy whereas moderate true classification percentage was achieved for the two falls and the remaining activity SCH. Since the act of sitting involves producing a downward movement as is the case with falls, some samples of both falls were classified as being SCH instead of their correct types.

Another classification algorithm considered in this work was a two layer feed-forward Neural Network, samples of the sensor readings were given as inputs to train the network. After suitable training, the remaining samples were input to observe the classification performance.

As can be seen from Table 5, similar to the SVM, the Neural Network is able to correct identify the activities of STD and WAL. It achieves a classification accuracy of 95.83% for a sitting fall and 91.87% for a standing FOL fall. Another point of interest is the fact that the Neural Network does not confuse the falls as SCH as was the case in SVM thus providing a more robust result in the differentiation between falls and activities generally.

Table 4. Confusion matrix for classification using SVM

Total no: of samples		BSC	FOL	SCH	STD	WAL	Detection Percentages
48	BSC	42	2	3	1	0	87.50
49	FOL	3	42	4	0	0	85.71
41	SCH	1	1	35	4	0	85.366
77	STD	0	0	0	77	00	100
78	WAL	0	0	0	0	78	100

Table 5. Results of Neural Network

Total no: of samples		BSC	FOL	SCH	STD	WAL	Detection Percentages
48	BSC	46	1	1	0	0	95.83
49	FOL	2	45	0	0	0	91.836
41	SCH	1	2	38	0	0	92.68
77	STD	0	0	0	77	0	100
78	WAL	0	0	0	0	78	100

Table 6. Results of Neural Network

Fall/Activity	Detection Percentages using SVM	Detection Percentages using Neural Network
BSC	87.50	95.83
FOL	85.71	91.84
SCH	85.366	92.68
STD	100	100
WAL	100	100
Overall Accuracy	91.71	96.07

Comparing the two classification techniques (Table 6) when using AR modeling feature extraction, Neural Networks provide an overall classification accuracy of 96% whereas SVM results in a classification accuracy of 91.7% thus providing better performance for classification. It provides significantly better results when comparing the performance for the two falls and the sitting activity.

6. Conclusion

This work discusses the use of Auto-Regressive modeling for the use of fall detection in a non-restrictive setting. Using a dataset that does not enforce positional directives while performing falls and activities of daily living we have provided a comparison of two types of classification algorithms to address detection of falls using inertial sensors. The intent of keeping features extraction independent of data acquisition is a complicated task as no features which are dependent on values of specific axes can be considered. In our experiments we have observed that a two layer feed-forward Neural Network has provided better classification accuracy as compared to a SVM with a Gaussian Kernel.

Future work in this direction would be the use of deep learning to improve fall detection rates, moreover, the effectiveness of each sensor can also be explored.

7. References

1. Chaudhuri S, Thompson H, Demiris G. Fall detection devices and their use with older adults: a systematic review. Journal of Geriatric Physical Therapy. 2014; 37:178-96. https://doi.org/10.1519/JPT.0b013e3182abe779 PMid:24406708 PMCid:PMC4087103
2. World Health Organization: WHO Global Report on Falls Prevention in Older Age. Community Health (Bristol). 2007; 53.
3. Alwan M, Rajendran PJ, Kell S, Mack D, Dalal S, Wolfe M, Felder R. A Smart and Passive Floor-Vibration Based Fall Detector for Elderly. 2006 2nd International Conference Information Communication Technology. 2006; 1:3-7. https://doi.org/10.1109/ictta.2006.1684511
4. Mager B, Patwari N and Bocca M. Fall detection using RF sensor networks. London: IEEE Personal, Indoor and Mobile Radio Communications Conference (PIMRC 2013). 2013 Sept. 9. https://doi.org/10.1109/pimrc.2013.6666749

5. Avinash C Kak and Malcolm Slaney. IEEE Press: Principles of computerized tomographic imaging. 1988.
6. Rougier C, Meunier J, St.-Arnaud A, Rousseau J. Monocular 3D head tracking to detect falls of elderly people. Annual International Conference IEEE Engineering Medicine Biology Proceedings. 2006; 6384-87.
7. Bian Z-P, Hou J, Chau L-P, Magnenat-Thalmann N. Fall detection based on body part tracking using a depth camera. IEEE Journal Biomedical Health Informatics. 2015; 19:430-9. https://doi.org/10.1109/JBHI.2014.2319372 PMid:24771601
8. Stone E, Skubic M. Fall Detection in Homes of Older Adults Using the Microsoft Kinect. IEEE Journal Biomedical Health Informatics. 2014; 19:290-301. https://doi.org/10.1109/JBHI.2014.2312180 PMid:24733032
9. Zhang Z. Microsoft kinect sensor and its effect. IEEE Multimed. 2012; 19:4-10. https://doi.org/10.1109/MMUL.2012.24
10. Hansen TR, Eklund JM, Sprinkle J, Bajcsy R, Sastry S. Using smart sensors and a camera phone to detect and verify the fall of elderly persons. Citeseer: European Medicine, Biology and Engineering Conference. 2005; p. 2486.
11. Wu G, Xue S. Portable preimpact fall detector with inertial sensors. IEEE Transactions on Neural Systems and Rehabilitation Engineering. 2008; 16:178-83. https://doi.org/10.1109/TNSRE.2007.916282 PMid:18403286
12. Bourke AK, O'Brien JV, Lyons GM. Evaluation of a threshold-based tri-axial accelerometer fall detection algorithm. Gait Posture. 2007; 26;194-9. https://doi.org/10.1016/j.gaitpost.2006.09.012 PMid:17101272
13. Vavoulas G, Pediaditis M, Spanakis EG. The MobiFall Dataset-An Initial Evaluation of Fall Detection Algorithms Using Smartphones. 2013 IEEE 13th International Conference on Bioinformatics and Bioengineering (BIBE). 2013; 1-4.
14. Vavoulas G, Chatzaki C, Malliotakis T, Pediaditis M, Tsiknakis M. The MobiAct Dataset : Recognition of Activities of Daily Living using Smartphones. Proceedings of International Conference on Information and Communication Technologies for Ageing Well and e-Health (ICT4AWE 2016). 2016; 143-51.
15. Hsieh C-J, Chang K-W, Lin C-J, Keerthi SS, Sundararajan S. A Dual Coordinate Descent Method for Large-scale Linear SVM. Proceedings of 25th International Conference on Machine Learning - ICML '08. 2008; 408-15. https://doi.org/10.1145/1390156.1390208
16. Jiangpeng Dai, Xiaole Bai, Zhimin Yang, Zhaohui Shen, Dong Xuan: PerFallD: A pervasive fall detection system using mobile phones. 2010 8th IEEE, International Conference Pervasive Computing Communications Workshops, (PERCOM Work). 2010; 292-7.
17. Engin M. ECG beat classification using neuro-fuzzy network. Pattern Recognition Letters. 2004; 25:1715-22. https://doi.org/10.1016/j.patrec.2004.06.014
18. Patterson AHD, Patterson BYHD. Biometrika Trust: The Use of Autoregression in Fitting an Exponential Curve. The Use Of AUTOREGRESSION YX-1. 2014; 45:389-400.
19. Zeng M, Nguyen LT, Yu B, Mengshoel OJ, Zhu J, Wu P, Zhang J. Convolutional Neural Networks for Human Activity Recognition using Mobile Sensors. Proceedings of 6th International Conference Mobile Computer Applications Services. 2014; 6:197-205. https://doi.org/10.4108/icst.mobicase.2014.257786
20. Hanrahan Grady. CRC Press: Artificial neural networks in biological and environmental analysis. 2011.
21. Bishop C, Nasrabadi N. Pattern Recognition and Machine Learning. 2007.