Detecting Falls Using a Wearable Accelerometer Motion Sensor

Hoa Nguyen
Auckland University of Technology
Auckland, New Zealand
h.nguyen@aut.ac.nz

Farhaan Mirza
Auckland University of Technology
Auckland, New Zealand
farhaan.mirza@aut.ac.nz

M. Asif Naeem
Auckland University of Technology
Auckland, New Zealand
mnaeem@aut.ac.nz

Mirza Mansoor Baig
Auckland University of Technology
Auckland, New Zealand
mirza.baig@aut.ac.nz

ABSTRACT

This research aims to early detect falls based on the rapid acceleration changes using the threshold based approach, using a single accelerometer. We propose the Acceleration Change-based Falls Detection Algorithm (ACFDA). The ACFDA observes and detects the rapid change of acceleration in vertical axis and the average value of signal magnitude vector of acceleration to differentiate falls from other activities of daily life (ADL). Initial results demonstrates that our algorithm achieved 100% of sensitivity, 95.65% of specificity and 96.35% of accuracy when tested with a total of 44 intentional falls and 230 ADLs in 32 datasets. Future work will focus on developing other strategies to reduce false alarms for improving both specificity and accuracy of the algorithm while still maintaining 100% of sensitivity.

CCS CONCEPTS

• **Theory of computation** → **Design and analysis of algorithms**; • **Information systems** → *Information systems applications*; • **Human-centered computing** → *Ubiquitous and mobile computing*;

KEYWORDS

ACFDA, falls detection, accelerometer, algorithm, threshold

ACM Reference Format:
Hoa Nguyen, Farhaan Mirza, M. Asif Naeem, and Mirza Mansoor Baig. 2017. Detecting Falls Using a Wearable Accelerometer Motion Sensor. In *Proceedings of the 14th EAI International Conference on Mobile and Ubiquitous Systems: Computing, Networking and Services (MobiQuitous 2017),* Jennifer B. Sartor, Theo D'Hondt, and Wolfgang De Meuter (Eds.). ACM, New York, NY, USA, Article 4, 10 pages. https://doi.org/10.1145/3144457.3144484

1 INTRODUCTION

According to the United Nations (UN), the population all over the world are aging rapidly, with the increase of 48% in the number of people over 60 years old from 2000 to 2015, and this proportion is estimated to be around 55% and 133% in the period of 2015 - 2030 and 2015 - 2050, respectively [8]. This trend has led to the need of developing intelligent healthcare applications for older adults to support their daily living more independent and safer. Falls are one of the health problems that affect many older adults in the world. According to the World Health Organization report, about 28-35% people aged over 65 and 32-42% people aged over 70 experience falls each year [1]. Previous studies show that almost 50% of the old aged have a minor injury after falling, while about 25% of them have a more serious injury such as fracture [10]. Falls may cause injury related death to people aged over 79 [15] as well as hospitalization among older adults [4]. In a research conducted by Vellas et al., statistics show that 70% of the old aged fallers could not get up without support, although 47% of them experienced a non-injured fall [29]. This research also reports that the capability of admitting to hospital increases if the fallers remain helpless on the ground for a longer time. This long lie time may lead to hypothermia, dehydration, broncho-pneumonia and pressure sores [23], [27]. It is found to be a marker of weakness, illness and social isolation as well as to be related to high mortality rates among the old aged [19]. As indicated in [19], more than 20% of the fallers who had been on the floor for an hour or more admitted to hospital and their morbidity rates within 6 months were very high. Falls affect not only patients' physical health but also their mental health such as the fear of falling and social isolation among them [17] or the fear and loss of independence [6]. For this reason, falls detection systems are necessary to older adults, especially to those who live alone. These systems can monitor patients' daily activities, detect falls automatically and alert their family's members or caregivers if a fall is detected. Hence, the fallers can be supported timely and the consequence of the fall can be minimized. For example, the lying time on the floor of patients may be reduced [12].

In recent years, falls detection systems have drawn significant attention of researchers and technologists with the dramatic increase in the numbers of studies on this application. Falls detection systems operate based on discriminating between falls and Activities of Daily Living (ADL). In these systems, data which can be acceleration signals, images, pressure signals,... are collected by sensors and are processed and classified into fall events or ADLs [12]. Based on the devices used, falls detection systems can be classified into three groups including ambient devices based systems, camera based systems, and wearable devices based systems. In the

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than ACM must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org.
MobiQuitous 2017, November 7–10, 2017, Melbourne, VIC, Australia
© 2017 Association for Computing Machinery.
ACM ISBN 978-1-4503-5368-7/17/11…$15.00
https://doi.org/10.1145/3144457.3144484

first type, ambient devices such as pressure sensors, PIR sensors, Doppler radars, microphones are used to detect falls. These ambient sensors are cheap and non-invasive, however the major limitation of ambient device based systems is the effect of environmental factors on their accuracy [28]. The second type, using cameras, tracks users' movements and detect falls based on the inactive state of users for a long time. Although these systems are less invasive, the spatial coverage and privacy are the main disadvantages of them. Frequently, accelerometers and/or gyroscopes are used in the last type to monitor users' motion and differentiate falls from normal daily activities. Wearable devices are portable, cheap and easy to use, however users need to wear them all the time to avoid false alarms.

Wearable devices can be smartphones or other devices such as watches, belts, waists, etc. with embedded motion sensors for detecting posture and motion of the body. Most motion sensor-based systems use accelerometers, and some of them incorporate sensors such as gyroscopes or magnetometers. Information about the gait, balance, and position of users can be analyzed in order to identify falls. Wearable devices - based solution is known as the more preferable solution than others due to its advantages including high accuracy and mobility [21]. Many studies employ multiple accelerometers [3], [14] or a combination of accelerometers and gyroscopes [18] to observe motion patterns of the patients. In these studies, it is argued that the accuracy of activity classification will increase when the number of sensors increases. However, a large number of sensors may result to the patients' inconvenience.

Many studies use accelerometers which are independent devices or embedded in smartphones to collect changes in acceleration. A fall is detected when the Signal Magnitude Vector (SMV) of acceleration in all three dimensions (X, Y, and Z) exceeds a critical threshold [5], [31]. Other research relies on both threshold and detection of body orientation after a fall [13], [25] or develops machine learning algorithms to detect falls [21], [28].

In this paper, we proposed a falls detection algorithm based on data collected from a single accelerometer which is called Acceleration Change-based Falls Detection Algorithm (ACFDA). The **unique feature** of our algorithm is that it observes the rapid change between the maximum and minimum values of acceleration in vertical axis and the average value of SMV to identify falls from other movements. The tested results show that our proposed algorithm achieves *high sensitivity*, *specificity*, and *accuracy* without using any complex computations.

The rest of the paper is organized as follows. Section 2 summarizes some of the existing research on falls detection systems that used wearable sensing devices. Furthermore we discusses the two types of algorithms that are employed in many of these systems. Section 3 presents the processes of data collection and data analysis and proposed our falls detection algorithm ACFDA. Section 4 explains the pilot testing of our proposed algorithm including experimental results and discussion. The limitations of this work and future direction is proposed in Section 5.

2 RELATED WORK

Falls detection systems based on wearable devices use data from sensors that are worn by the users or integrated in the clothes. Most of the wearable sensors used are in the form of accelerometer devices [12]. The accuracy of the systems depends on the sensors used and the type of classifications [28]. This section explores current work on wearable devices - based falls detection systems.

2.1 Sensing Techniques

In falls detection applications that rely on wearable sensors, activities are detected using either motion sensors that are integrated in *smartphones* or *external motion sensors* which use accelerometers.

2.1.1 *Smartphone-based systems.*

Smartphones are widely used to detect falls in many systems due to their cost-effectiveness and availability. Smartphones integrate inexpensive MEMS sensors including accelerometer, gyroscope, compass, magnetometer, proximity light and pressure sensors. These sensors provide computational capabilities and allow different algorithms to be applied to increase the accuracy of the systems [28].

A falls detection using both accelerometers and smartphone is demonstrated in the study of Basili et al. [5]. The position of the person (sitting, standing or lying) can be estimated based on the orientation of the smartphone and the orientation of the accelerometer with respect to the vertical position. A possible fall is considered based on comparing the acceleration magnitude of the smartphone and the vertical acceleration component of the device with 5 threshold values.

Kau and Chen [16] proposed a falls accident detection and rescue system using a tri-axial accelerometer and an e-compass embedded in the smartphone for acquiring the posture of motion activities for older adults. When a fall is detected, a loud sound as a warning signal will be sent out. Simultaneously, the current position and important personal information of the patient will be transmitted to the coordination center via 3G network. To distinguish fall events between ADLs, four features were applied in the proposed algorithm including the reference points for the sampling of SMV sequence, the slowly varying waveform of SMV, and the pitch which is the angle between the Y-axis and the ground. In addition, authors proposed to use frequency component analysis in the classification process for higher accuracy. Experimentations were conducted 50 times for each of the nine different kinds of activities consisting of a fall down event, running, walking, sitting down, going upstairs, going downstairs, treading, jumping, and wavering the smartphone. The system's performance can achieve up to 92% of sensitivity and 99.75% of specificity.

In a research conducted by Yildirim et al. [31], authors designed a falls detection application using a smartphone. This application can run on any Android devices that have an accelerometer and ability to make calls and send SMS. A fall is detected using a single threshold for SMV. In order to limit false alarms, an extra screen is applied to ask if the user is OK or not after detecting a fall. The testing experiment is performed by 5 people doing five times for each of the following activities: falling, walking, sitting, jumping, lying, and climbing stairs. Results show that the system has missed 9 out of 25 falls, and there are 7 out of 25 jumping times which were identified as falls.

Reusing the sensing and computing capabilities of smartphones is a cost-effective and easy-to-deploy way of implementing falls detection systems [6]. However, the capability of the embedded sensors, especially with built-in accelerometer are still under discussion and the system's performance strongly relies on the quality of the embedded sensors. In addition, information about the manufacturer model or the features of the installed sensors in smartphones are not disclosed which makes it difficult to validate the accuracy of the systems. Moreover, the range of built-in accelerometers hardly reach 2g compare to a typical range between 6g and 16g of external accelerometers. This is one of the limitations of smartphone - based falls detection systems [6].

2.1.2 Accelerometer-based systems. Accelerometer is the most widely used device to detect falls out of wearable devices [20]. Acceleration data are collected during falls and ADLs using independent tri-axial accelerometers attached to different parts of the body. The number of published studies based on accelerometers are higher in comparison with smartphone - based systems [12].

Kangas et al. [14] proposed three fall detection algorithms based on data provided by a single accelerometer. To detect a fall, two or more of these following phases of the acceleration changes occurring during an accidental fall including *Start*, *Impact*, *Aftermath*, and *Posture* phase [22] are taken into account. These four phases can articulate whether a person is stable or experiences an accidental fall. In the first algorithm, a fall is detected if a possible impact occurs and follows a horizontal posture after that. While the second and the third algorithms detect start, impact, and posture phases. All the above algorithms are based on the detection of the posture phase which shows the different orientation of the subject's body after a fall. Data from a single 3-axis accelerometer can provide posture phase detection. However, the accurate estimation of orientation of the subject may not be obtained by these data, especially in cases users are trembling or rapidly moving the accelerometer [22].

In the study conducted by Ozdemir [21], a total of 2520 records of fall dataset was collected from 14 volunteers with six sensor units placed on different parts of the body including head, chest, waist, right wrist, right thigh and right ankle. The volunteers performed 36 sets of movements consisting of 16 ADL sets and 20 fall sets with five repetitions to determine the best sensor positioning for wearable fall detection devices. Applying this dataset on six machine learning techniques, namely the k-nearest neighbor (k-NN) classifier, Bayesian decision making (BDM), support vector machines (SVM), least squares method (LSM), dynamic time warping (DTW), and artificial neural networks (ANNs), the classification performances of six sensors are investigated. Results show that waist is found to be the most suitable position for sensor placement on the body with 99.96% of sensitivity by using k-NN classifier.

A fall detection system base on a tri-axial accelerometer MMA 7260 is designed in [24]. The system can detect the actual fall and the possible fall direction such as frontward falls, backward falls, and lateral falls. An actual fall is assumed when the acceleration value reaches to the positive or negative thresholds in X or Y directions. The minimum absolute value of thresholds is set to be 1.5g in cases of fast falls and medium falls. For testing, a total of 100 fall experiments are performed for five sets of threshold values. The algorithm achieves the correct detection rate of 95% for frontward falls, 98% for backward falls, 87% for left-side falls, and 88% for right-side falls. However, it is not easy to determine the direction of the falls if the falls having direction that does not align with front, back, left, and right directions occur.

2.2 Falls Detection Techniques

Falls detection systems operate based on the principle of distinguishing falls from other conventional movements which are called ADLs. Falls detection techniques can be classified into two categories: threshold-based approaches and pattern recognition methods [9]. Threshold-based algorithms compare one or several magnitudes captured by the motion sensors with certain thresholds to make decision about activities detection. Pattern recognition methods base on diverse classification techniques such as SVM [16], [21], [28], k-NN [21], [28], ANNs [21], [28], Naive Bayes classifier [28], decision trees [32], Hidden Markov Models (HMM), fuzzy logic. These methods comprise Artificial Intelligence, rule-based algorithms and machine learning-based algorithms [6]. Some of the threshold-based algorithms and machine learning-based algorithms that are applied widely in existing wearable devices-based falls detection systems are considered below.

2.2.1 Threshold-based Algorithms. Threshold-based algorithms are applied in many existing falls detection application using wearable sensors such as smartphones [5], [31], or external accelerometers and other sensors [13], [14], [24]. In [13], authors determined thresholds for total sum vector, dynamic sum vector, fast changes in acceleration signal, and vertical acceleration for falls detection algorithms using data gathered by a single accelerometer. A fall is detected by comparing one of the above four parameters with their defined thresholds and checking lying posture after falling. Results show that the algorithms can achieve high sensitivity and specificity up to 100%. However, the algorithms use both threshold comparison and posture detection.

Basili et al. [5] proposed a system consists of a smartphone and a 3-axis accelerometer to detect falls. In this system, authors determine a low threshold for the vertical acceleration component of the device and four thresholds for the acceleration magnitude of the smartphone including: a low threshold and a up threshold for falling state, a low threshold and a up threshold for lying state. Each activities (sitting, lying, standing up, going upstairs and downstairs) was performed 20 times, while falling was performed 41 times. Threshold values are set for three settings: a reference example (A), an optimization of specificity (B), and the best value of the sum of sensitivity, specificity and accuracy (C). Results show that in setting B, no wrong fall has been detected but the system has 8 missed falls. In converse, the system has detected all the real falls correctly and has some false positives in setting C.

Authors in [11] presented a falls detection algorithm based on a wearable sensor system comprised of a 3-axis accelerometer and a gyroscope. Critical thresholds for total sum acceleration vector and angular velocity are used for detecting a fall with optimal sensitivity and specificity. Experiments were performed on 27 young healthy subjects and 9 middle-age subjects with the sensors attached on the chest. Movements such as standing, walking, sitting down, standing up, stepping, and running and 4 different falls (forward, backward, right/left side) were conducted. Experimentations show that the proposed algorithm can detect falls with 96.3% sensitivity and 96.2% specificity.

In [25], authors develop a falls detection system using the Bosch BMI055 tri-axial accelerometer. The vector sum of acceleration data is compared with two thresholds to check the *Free Fall* and *Impact* states. A fall is detected when a free fall state is followed by a strong impact state and a horizontal posture. An experiment with ten healthy people was done to test the performance of this algorithm. The performed falls are backward, sideward and forward, and the performed ADLs are walking, sitting, jumping, going upstair and downstair. In this experiment, 14 falls out of 150 falls were identified as ADL and all 250 ADLs were classified correctly. The system achieved 91% of sensitivity and 100% of specificity. The false negatives mostly are caused by the simulation of falls, especially by simulating forward falls.

Threshold-based algorithms are simple to implement and have minimal computational work [12]. Falls detection systems using smartphones are mainly limited by computing and storage capabilities. Hence, threshold-based algorithms are preferred in these systems. A simple application run in smartphones can implement threshold comparison straightforwardly and in real-time [6]. However, fall detection focusing only on large acceleration can result in many false positives. For example, the average value of SMV during running (2.3-2.8g) overlaps with this value during falling (2.4-5.4g) [11]. To reduce false alarms, many works rely on detection of body orientation after falling . However these systems may be affected by activities with similar postures such as sleeping, reclining [11].

2.2.2 Machine learning-based Algorithms. In a study conducted by Vallabh et al. [28], the differentiation between falls and ADLs (including standing, walking, jogging, jumping, walking upstairs, walking downstairs and sitting on a chair) was performed using the MobiFall dataset. A total of 38 features were extracted from the dataset and were filtered to find the most useful features. Five features consisting of z median, x mean, y mean, x median, and skewness of SMV were used to train the classification models. Five different classification algorithms including Naives Bayes, k-NN, LSM, ANN, and SVM were implemented and evaluated. Results show that with an accuracy of 87.5%, sensitivity of 90.70% and specificity of 83.78%, k-NN is the best classifier. Compare to LSM and Naive Bayes, the k-NN, ANN, and SVM had the better accuracy and are viable options for implementation.

Yuan et al. [32] proposed a fall detection algorithm and an ADL classification algorithm for accelerometer-based systems. Fall detection algorithm sets thresholds for six states including *F0 - resets state*, *Activity*, *Inactivity*, *Free Fall*, *FIFO*, and *Trigger*. ADL classification algorithm is developed based on decision tree learning. Pilot tests with a device worn on person's left wrist was performed repeatedly 20 times. The results show that the system has one missed fall when walking and walking upstair, and has three missed falls when walking downstair. Three picking up and putting down objects are identified as false positives. The proposed algorithms are more power-efficient than conventional algorithms due to allowing to process accelerometer data completely locally. However, these algorithms need to be developed for eliminating false negatives.

Compare to threshold-based algorithms, machine learning based algorithms are more sophisticated, however if they lead to better detection is questionable. Nevertheless, the machine learning algorithms demand high mathematical skills and are computationally intensive [12] that may not respond in real-time [11]. Results from [4] show that all the five machine learning algorithms in their research (Logistic Regression, Naive Bayes, k-NN, Decision tree, and SVM) provided sensitivity and specificity of at least 90%, while the studied threshold-based algorithms (Kangas2Phase, Kangas3Phase, BourkeUFT, BourkeLFT, and Bourke4Phase) have sensitivity and specificity from 0 to 100%.

The effectiveness of machine learning algorithms versus threshold based is not the focus of this paper, but based on the literature review above we decided to apply threshold based approach as it offers more accuracy and simplicity. We also found the algorithms in the studies reviewed are not clearly documented, therefore we not only present the experimentation data, but a clear articulation of the algorithm is accompanied in the following sections.

3 DESIGN

The scope of this study is to determine the parameters and thresholds for falls detection, using motion data measured by an accelerometer.

3.1 Preliminary

In many falls detection algorithms based on threshold approach, falls are identified by comparing SMV with upper and/or lower thresholds [5], [11], [13], [25], and the thresholds are determined by experimentation in many studies [11]. Similarly, this study gathered motion data using a single tri-axial accelerometer and analyzed the collected data to investigate the difference between accelerations in falling and ADL. The results were used to develop our algorithm.

3.1.1 Data Collection. Intentional falls including forward, backward, left-side and right-side falls and ADLs including walking, sitting, standing and lying were performed by one subject (female, aged 36 years). Accelerations during these activities were measured with the accelerometer fixed on the chest of the person which is the same place as in [11], [16]. The position of the accelerometer was identical all the times when data were collected.

Data collection was performed in the two following steps:

Step 1: Collecting sample patterns. Accelerations were collected in this step for data analysis in order to identify the difference in acceleration between falling and ADL. Each activity was performed 20 times at different locations. Walking, standing, and sitting were performed separately, each event was recorded in a 60-second dataset. While each fall event was performed along with a lying state after that with a total of 20-second duration. The number of each activities in sample patterns is summarized in Table 1.

Table 1: Number of activities in sample patterns

Activities	Number
Walking	20
Standing	20
Sitting	20
Lying	20
Falling	20

Table 2: Number of activities in test patterns

Activities	Falling	Idle	Standing up	Walking
Number	44	126	56	48

Step 2: Collecting test patterns. In this step, data was collected from the same subject for testing the performance of the proposed falls detection algorithm. 32 datasets was recorded, each dataset consists of a mixed set of activities (walking, sitting, standing, falling, lying, standing up). A total of 44 intentional falls and 230 ADLs collected for testing is summarized in Table 2. These patterns are used for validation to determine the sensitivity, specificity, and accuracy of ACFDA.

3.1.2 Data Analysis. Data analysis was performed using MATLAB program to find the maximum, minimum, and average values of accelerations in each axes and in the SMV for each of the following activities: falling, walking, standing, sitting, and lying. SMV is calculated from sampled data as indicated in Eq.[1]:

$$SMV = \sqrt{x^2 + y^2 + z^2} \tag{1}$$

Where x, y, z are normalized accelerations in X, Y, and Z axes. Raw accelerations gathered by the accelerometer are normalized from *digital count* into *gravity unit* (g) as illustrated in Eq.[2].

$$a\_normalized = \frac{a}{1024}[g] \tag{2}$$

Where a is raw accelerations in each axis.

Algorithm 1 summarizes the data analysis process. Raw acceleration data collected in sample patterns are loaded from accelerometer to a computer (line 1) and are converted to gravity unit (line 2). SMV is calculated for each sample point (line 3). The maximum, minimum, and average values of accelerations in each axis (x, y, z) and in the SMV are identified (line 4).

Algorithm 1 Pseudocode code for Data Analysis

Input: Raw data (t, Ax, Ay, Az) from Accelerometer.
Output: max, min, average of x, y, z, SMV.
Method:

1: Receive raw data (t, Ax, Ay, Az) from Accelerometer.
2: Convert raw accelerations (Ax, Ay, Az) into normalized accelerations (x, y, z), Eq. [2].
3: Calculate SMV, Eq. [1].
4: Find max, min, average of x, y, z, SMV.
5: When End of data $\Rightarrow$ **Exit**.

Table 3 summarizes accelerations in falling patterns. All the values in this table are in gravity (g). Each row of this table illustrates the data analysis results of each falling pattern.

Accelerations of falling patterns from Table 3 are represented in the form of quartile box as shown in Figure 1(a). Similarly, accelerations of ADL patterns including walking, standing, sitting,

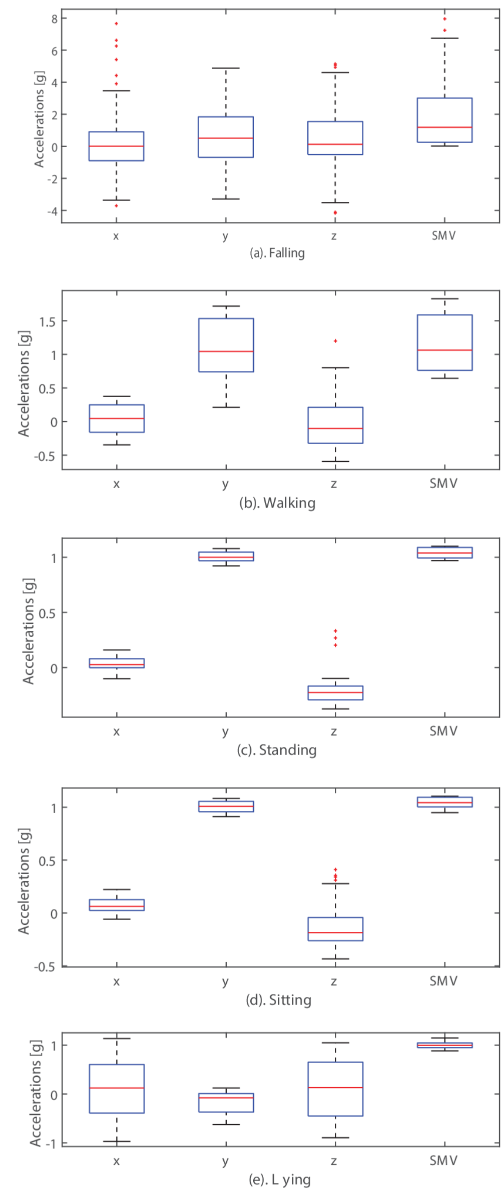

Figure 1: Accelerations in falling and ADL patterns

and lying are represented in Figure 1(b), 1(c), 1(d), and 1(e), respectively. The summarized acceleration tables for these patterns are not illustrated in this paper.

From Figure 1, it can be seen that SMV of acceleration in standing, sitting and lying patterns are nearly the same at around 1g. These activities are grouped into one state called *Idle*.

Additionally, from Figure 1, it can be seen that SMV in falling patterns ranges from 0.01g to 7.95g, a lot more fluctuation compared with the *Idle* state. In addition, the average value of SMV in falling patterns is above 1.1g, while this value in other activities is around 1g. However, the majority of falling patterns has SMV value between 0g and 3g which has some overlapping with this value

Table 3: Accelerations in falling patterns

Activities	x_max	x_average	x_min	y_max	y_average	y_min	z_max	z_average	z_min	SMV_max	SMV_average	SMV_min
Fall1	1.50	0.01	-1.57	1.56	0.44	-0.77	2.67	0.96	-0.47	2.84	1.31	0.24
Fall2	1.41	0.04	-1.13	1.48	0.52	-0.59	2.75	0.77	-0.41	2.80	1.19	0.14
Fall3	1.46	0.03	-1.09	1.76	0.43	-0.76	5.14	0.88	-0.34	5.63	1.24	0.22
Fall4	0.73	-0.02	-1.46	2.92	0.50	-0.71	4.93	0.81	-0.39	4.95	1.22	0.19
Fall5	1.32	0.07	-1.36	2.30	0.44	-0.93	3.72	0.86	-0.23	3.97	1.23	0.23
Fall6	1.13	-0.06	-2.30	3.97	0.32	-2.08	2.36	-0.10	-4.17	4.91	1.35	0.18
Fall7	0.65	0.04	-0.91	3.27	0.47	-1.49	1.28	-0.08	-3.35	4.02	1.23	0.17
Fall8	1.04	-0.17	-1.96	4.88	0.36	-1.99	2.64	-0.04	-3.05	5.11	1.30	0.15
Fall9	0.91	-0.01	-1.75	4.57	0.54	-1.90	2.86	0.00	-2.43	4.73	1.33	0.10
Fall10	0.82	0.01	-1.29	2.00	0.42	-1.70	2.21	0.20	-3.40	3.64	1.30	0.08
Fall11	0.90	-0.18	-2.45	3.63	0.56	-0.95	1.63	0.05	-1.54	3.99	1.14	0.08
Fall12	1.20	-0.18	-2.11	2.49	0.54	-1.11	1.52	-0.05	-1.70	2.95	1.18	0.10
Fall13	0.85	-0.15	-2.72	2.57	0.59	-1.58	1.54	0.03	-1.79	3.02	1.15	0.01
Fall14	0.95	-0.14	-3.36	2.48	0.58	-1.28	1.85	0.12	-0.99	3.53	1.11	0.12
Fall15	0.96	-0.11	-3.71	4.00	0.60	-1.97	1.79	0.13	-1.75	4.38	1.19	0.15
Fall16	6.61	0.31	-1.27	2.38	0.58	-1.13	1.39	0.18	-1.96	6.74	1.21	0.16
Fall17	7.66	0.42	-1.23	1.91	0.44	-1.04	1.56	0.21	-2.14	7.95	1.29	0.12
Fall18	6.25	0.36	-1.14	1.99	0.49	-1.64	2.53	0.12	-3.52	7.24	1.26	0.08
Fall19	5.41	0.44	-0.90	1.97	0.45	-1.91	1.77	0.14	-2.78	5.97	1.24	0.13
Fall20	4.41	0.41	-1.01	1.98	0.40	-4.27	1.73	0.09	-3.14	5.01	1.24	0.05

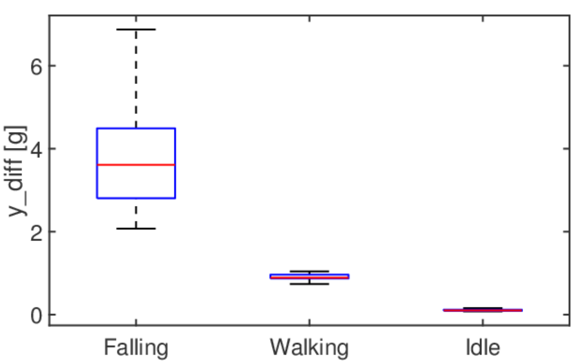

Figure 2: Difference between maximum and minimum values of acceleration in vertical

in walking patterns. On the other hand, the acceleration in y-axis (y-axis is the vertical direction corresponding with the position of the accelerometer) has more fluctuation compared with other activities.

The rapid change of acceleration in vertical axis (y_diff) in falling, walking, and *Idle* states was calculated using the difference between the maximum and minimum values in the period of time that activities occurred. These changes of falls and ADLs are shown in Figure 2. As being illustrated in this Figure, acceleration in vertical axis in falling activity has the most fluctuation which is usually larger than $2g$. While this value in walking activity and *Idle* state are around $1g$ and $0g$, respectively. There is no overlapping in this Figure, which helps us propose ACFDA.

3.2 Acceleration Change-based Falls Detection Algorithm (ACFDA)

In this paper, we proposed our algorithm based on defining thresholds for *the rapid change of acceleration in vertical axis* (y_diff) and *the average value of SMV* (*SMV_average*) which is summarized in the pseudo-code below. This algorithm allows to identify falls from normal daily activities including walking, standing, sitting, lying and standing up after falling. The algorithm was computed in MATLAB program.

Algorithm 2 Pseudocode code for ACFDA

Input: Raw data (t, Ax, Ay, Az) from Accelerometer.
Output: Falls detection results.
Parameters: window length (w); threshold for y_diff (th_y); threshold for $SMV\_average$ (th_S).
Method:

1: Initialize parameters: w = 2s; th_y = 2.0g; th_S = 1.1g.
2: Receive raw data (t, Ax, Ay, Az) from Accelerometer.
3: Convert raw accelerations (Ax, Ay, Az) into normalized accelerations (x, y, z), Eq.[2].
4: Analyze the normalized data in each 2s window to find: y_max; y_min; $y\_average$; y_diff; SMV (Eq.[1]); $SMV\_average$.
5: If ($y\_diff >= th_y$) and ($SMV\_average >= th_S$) then **Falling**.
6: Else **Normal activities**.
7: When End of data ⟹ **Exit**.

The first step (line 1) initializes parameters used in this program including window length, threshold for y_diff and threshold for

SMV_average. Raw data collected from the accelerometer in the format of .csv files are loaded into the program (line 2). Raw accelerations are divided by 1024 to obtain normalized accelerations [7] (line 3). The algorithm analyzes through these normalized data in windows with 2 second length and detect activities for each window. In the data analysis process (line 4), in each window, the maximum, minimum, average values of y-acceleration which are shown as *y_max, y_min, y_average* in the above pseudo-code are identified. Simultaneously, the rapid change of acceleration in y axis in each window (*y_diff*) is computed by subtracting *y_max* and *y_min*. SMV is calculated as illustrated in equation (1) and *SMV_average* is identified. In the activity detection process (line 5), two comparisons are performed to detect falls. If (*y_diff* >= th_y) and (*SMV_average* >= th_S) then it is assumed that a fall has occurred, otherwise the activities are identified as normal activities. The size of the window can range from 1 to 7.5 second to contain at least one cycle of a single activity [30]. In this experimentation, a window length of 2s was chosen (line 1) for capturing the largest change in y-acceleration when a fall occurs. The two thresholds are initialized (line 1) based on our results obtained in 3.1.2.

4 EXPERIMENTS

To collect motion data for developing and evaluating our proposed ACFDA, experiments were performed on one healthy subject (female, 36 years old). The subject simulated 4 different fall tests (frontward, backward, left-side, and right-side fall) and performed ADLs including walking, standing, sitting, lying, and standing up after falling. The subject involved in collecting sample data and test data is the same. Motion data were collected by a single tri-axial accelerometer fixed on the chest of the subject when performing the above activities.

4.1 Set up

The device used to collect motion data in this work is the accelerometer model X8M-3 [7]. X8M-3 includes a tri-axial 14-bit accelerometer with a range of $+/-8g$ and the resolution of 0.001 *g/count*. This device enables users to select sampling rate in the range of 6, 12, 25, 50, 100, and 200 *Hz*. Acceleration is collected in X, Y, and Z axes and stored real-time in 2GB internal flash memory. The collected data can be transferred to Windows or Linux operating system via USB interface without special software required. Data is written to files in text format, each data line includes a time entry (t), the raw accelerometer sensor readings from the X, Y, and Z axes (*Ax, Ay, Az*). These raw data is signed digital *counts* and are converted to g by dividing the value by 1024 [7]. The accelerometer was configured with the sampling frequency of 50 *Hz*, similar to the chosen sampling frequency in [24]. With the fixed position of the accelerometer, the measured data *Ax, Ay, Az* correspond to accelerations in the three following axes: X (lateral: left - right), Y (vertical: up - down), and Z (direction: front - back).

Data analysis and falls detection algorithms are written in MATLAB (MATLAB and Statistics Toolbox Release 2016b, The MathWorks, Inc., Natick, Massachusetts, United States) to receive the collected data from X8M-3 device, display and analyze accelerations as well as display falls detection results. All the data and results are stored for later analyses.

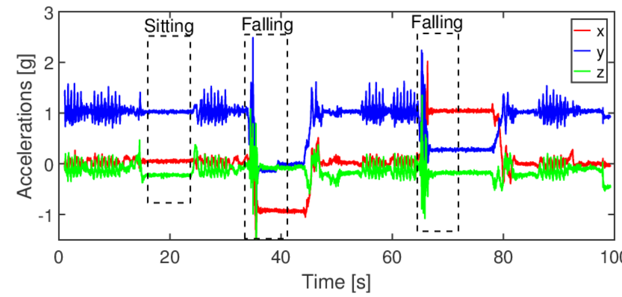

Figure 3: An example of testing pattern

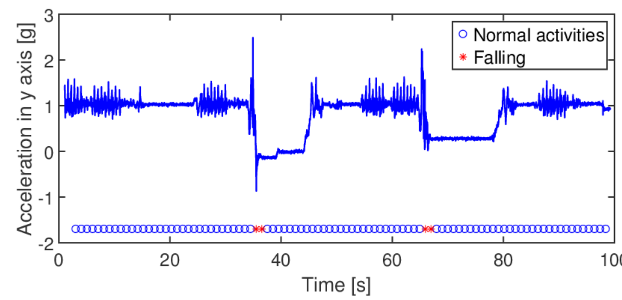

Figure 4: Falls detection result

4.2 Performance Evaluation

To test our algorithm, test patterns were applied as inputs for falls detecting program. A total of 32 test patterns were tested. The number of each activities in the test patterns was summarized in Table 2.

An example of testing pattern is illustrated in Figure 3. In this pattern, data were collected when user performed the following activities: walking (10s), standing (5s), sitting (10s), walking (10s), left-side falling and lying (10s), standing up and standing (10s), walking (10s), right-side falling and lying (10s), standing up and standing (10s), walking (10s), and standing (5s). Data are represented in this Figure are normalized accelerations in three axes. The result of applying our proposed algorithm with th_y = 2.0g and th_S = 1.1g on this testing pattern is illustrated in Figure 4. In this pattern, two falls are detected correctly, and all other activities are identified as ADLs.

In general most falls detecting algorithms can produce the following four possible outcomes:

- True Positive (TP): a fall is detected properly.

- False Positive (FP): a fall is detected when no fall has occurred. This outcome also is known as false alarm.

- True Negative (TN): no fall is detected when no fall has occurred.

- False Negative (FN): no fall is detected when a fall has occurred. This case is also called missed fall.

Based on these possible outcomes, the performance of the algorithm is represented including sensitivity, specificity, and accuracy which are given by [2]:

Table 4: Performance of the ACFDA

th_y (g)	TP	TN	FP	FN	Sensitivity	Specificity	Accuracy
2.0	41	226	4	3	93.18%	98.26%	97.45%
1.6	43	225	5	1	97.73%	97.83%	97.81%
1.3	44	220	10	0	100%	95.65%	96.35%

$$Sensitivity = \frac{TP}{TP + FN} * 100 \tag{3}$$

$$Specificity = \frac{TN}{FP + TN} * 100 \tag{4}$$

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} * 100 \tag{5}$$

Sensitivity is the ratio between truly identified falls and all falls which defines how successfully the algorithm detects falls. Specificity is the proportion of the algorithm to correctly identify ADLs and indicates how successfully the algorithm detects ADLs. Accuracy indicates how well the algorithms identify both falls and ADLs. It is derived from the rate between truly give decisions and all decisions.

Table 4 shows the performance of the ACFDA with different thresholds of th_y and th_S = 1.1g.

When th_y is set to 2g, ACFDA can detect 41 out of 44 falls, it cannot detect 3 falls, 4 standing up states after falling are detected as falls. In this case, the algorithm achieves 93,18% of sensitivity, 98.26% of specificity and 97.45% of accuracy. However, false negative need to be eliminated because it may be dangerous for users if a fall occurs but the system cannot detect it. A simplest way to improve sensitivity of the algorithm is to decrease th_y.

When th_y is decreased, the number of falls which are detected increases. It results in the increase in sensitivity. This is because when a high threshold is set, some weaker falls may be classified as safe activities. However, when a low threshold is set, fall positives may increase because strong daily activities may be classified as falls. As we can see from Table 4, when th_y is selected as 1.6g, the number of false negative decreases from 3 to 1 comparing to th_y = 2.0g, while the number of false positive increases from 4 to 5. The sensitivity and accuracy of the algorithm are improved, although the specificity reduces slightly. When th_y is chosen as 1.3g, all falls in the test patterns are detected correctly, however the number of false positive increase to 10, resulting to the reduction in both specificity and accuracy of the algorithm. In this experimentation, all the false positive cases are the result of the *standing up* states after falling.

An algorithm having high scores for all sensitivity, specificity and accuracy is desirable. However, in term of the key purpose of falls detection, it can be pointed out that the success of the algorithm mostly depends on the frequency of false negative because it is the most dangerous and unwanted case [21]. False negative is a serious mistake for the algorithm and for a reliability of the system, hence it is expected to be 0. Falls detection systems must achieve very high score in sensitivity and accept the trade-off between sensitivity and specificity. False positive is a lesser concern, however it needs to be avoided as well to prevent confusion and unnecessary escalation [21].

4.3 Discussion

This study was conducted to investigate the accelerations collected from a body attached accelerometer when performing intentional falls and activities of daily living. The purpose of this study is to determine the important parameters and their threshold values for distinguishing falls from ADLs. Our algorithm considers the fast change of acceleration in vertical axis (y_diff) and the average value of SMV (SMV_average). It achieved up to 100% of sensitivity and 95.65% of specificity, based on a single accelerometer attached at the chest. Although previous studies have achieved some significant results, accuracy of the strategies that applied thresholds for SMV is still below desired levels. For example, the system designed by Yildirim et al. [31] has 9 false negatives out of 25 falls and 7 false positives out of 25 jumpings. However, their work mainly focused on designing and realization of the fall detection system rather than on developing falls detection algorithms. On the other hand, despite of using complex and sophisticated computation, falls detection algorithms based on machine learning approach mentioned in literature review still cannot detect some falls. For instance, authors in [32] used five features related to all x, y, z, and SMV to train five classification models. Their best classifier achieves 90.70% of sensitivity and both specificity and accuracy are under 90%. Compared to previous works, our ACFDA has all the three performance ratio including sensitivity, specificity and accuracy above 90% for three chosen thresholds for *y_diff*.

In this work, from the experimental data collected, a subset of data (sample patterns) is analyzed to identify the thresholds for *y_diff* and *SMV_average*. Determining the optimal thresholds is difficult due to improper selection of low thresholds may increase false positives, while improper selection of high thresholds may lead to the rise in false negatives [11]. Results from Table 4 illustrate that determining th_y = 1.6g for *y_diff* is the best choice for optimizing both sensitivity and specificity. In this case, ACFDA achieves 97.73% of sensitivity, 97.83% of specificity, and 97.81% of accuracy which are much higher than prior results reported in literature. In order to optimize sensitivity up to 100%, th_y = 1.3g is chosen, hence all false negatives are eliminated. However, in some circumstances, other daily activities such as going upstair or downstair may have *y_diff* around this low th_y (data of going upstair and going downstair are not presented in this paper). The purpose of adding lower threshold as 1.1g for *SMV_average* in our proposed algorithm is to limit these incorrect classifications. In this case, ACFDA still identified 10 *standing up* states after falling as falls. These false positives can be eliminated when applying other strategies after detecting falls such as sending message to patients for checking the patients are fine or not as in [31] or checking whether posture phase is horizontal (which means a real fall) or vertical (not a real fall) as in [13]. Alternatively, other devices such as cameras can be used to track patients' movement after detecting a fall to identify is it a real fall or not. Therefore, the number of datasets for sample patterns as well as the number of subjects who take part in data collection phase should be increased for setting threshold values more properly.

This research focused on differentiate falls from slow motion activities. However, daily living activities consist of various activities which have slow or fast changes in accelerations. Falls may not be differentiated from fast motion activities such as running or jumping based on determining lower thresholds for fast change of acceleration in vertical axis and the average value of SMV. In order to identify falls from these strong activities, ACFDA need to be improved. For example, the average value of acceleration in vertical axis ($y\_average$) can be checked. From Table 3, we can see that $y\_average$ values of all the falling patterns (the light cyan column) are lower than $0.6g$. In the mean while, these values of other activities such as walking, standing, running, and jumping are approximate $1g$ (data of running and jumping states are not shown in this paper). In addition, ACFDA mainly focuses on identify falls from ADLs. The scope of this study does not focus on distinguishing among these states: lying, sitting, and standing. Identifying lying state correctly may support falls detection systems to detect falls more accurately because a fall usually occurs along with a lying state after that. From Figure 1, it is clearly seen that $y\_average$ in sitting and standing states is around $1g$, while $y\_average$ of lying state is much lower (around $0g$). Therefore, determining upper threshold or lower threshold for $y\_average$ can be used for differentiating lying state from sitting and standing states. Furthermore, the changes of accelerations in X, Y, and Z axes could be analyzed to identify the orientation of falling activity including frontward fall, backward fall, left-side fall, and right-side fall. This analysis may be useful for distinguishing lying between other idle states as well.

5 CONCLUSION AND FUTURE WORK

Falls detection is an important application in the area of ambient assisted living, especially in older adult care. Falls detection systems based on wearable sensors have drawn significant attention of researchers recently, mostly making prediction based on collecting data from both accelerometer and gyroscope.

This study introduced an algorithm called ACDFA for falls detection using a single accelerometer that makes ACFDA less complex than many prior studies. Using two threshold comparisons, ACFDA can detect falls with higher accuracy compared to previous related work. The proposed algorithm is a simple threshold method, hence ACFDA can be applied in falls detection systems that do not require robust computational platforms.

Likewise most of the previous studies, our thresholds and algorithm was developed based on collecting motion data from a young and healthy subjects. Despite the possible difference of actual data collected from the older adults, it is predicted that unexpected falls would experience higher changes in accelerations than intentional falls [11]. Therefore, applying ACFDA in real-world scenarios may still be effective. However, this algorithm need to be improved to enhance specificity while still remaining high sensitivity. Our further work will focus on applying other strategies after detecting falls to eliminate all false alarms.

In addition, in future work, we propose the validation of our algorithm using data collected from other sources such as built-in accelerometers in smartphones and the combination of external accelerometer and smartphone. Although the 8XM-3 mini accelerometer used in this work is convenient due to its small size, light weight, and long-life battery (up to 4 days working continuously) [7], collecting accelerations from other devices is useful to help us further develop our algorithm. Using multiple sensors to collect data for evaluating the proposed algorithm is necessary when taking into account the sensing heterogeneity for activity recognition [26]. Additionally, in our experiment, the sensor was attached to the chest of the subject. The sensor position will be tested in different places such as chest, waist in our further research to determine the optimum placement of sensors, targeted at optimizing falls detection accuracy and enhancing patients' convenience when using this device. Simultaneously, the number of subjects who take part in data collection process should be increased to investigate the variability in ADLs across users. Furthermore, our future studies may focus on testing and comparing machine learning algorithms with threshold-based algorithms over prolonged natural usage. Despite so many features being considered and complex algorithms being used, the accuracy gains have not surpassed standard threshold-based algorithms.

REFERENCES

[1] World Health Organization. Ageing and Life Course Unit. 2008. *WHO global report on falls prevention in older age.* World Health Organization.

[2] Bruno Aguiar, Tiago Rocha, Joana Silva, and Ines Sousa. 2014. Accelerometer-based fall detection for smartphones. In *Medical Measurements and Applications (MeMeA), 2014 IEEE International Symposium on.* IEEE, 1–6.

[3] Louis Atallah, Benny Lo, Rachel King, and Guang-Zhong Yang. 2010. Sensor placement for activity detection using wearable accelerometers. In *Body Sensor Networks (BSN), 2010 International Conference on.* IEEE, 24–29.

[4] Omar Aziz, Magnus Musngi, Edward J Park, Greg Mori, and Stephen N Robinovitch. 2017. A comparison of accuracy of fall detection algorithms (threshold-based vs. machine learning) using waist-mounted tri-axial accelerometer signals from a comprehensive set of falls and non-fall trials. *Medical & biological engineering & computing* 55, 1 (2017), 45–55.

[5] Luca Basili, Giuseppe DeMaso-Gentile, Cristiano Scavongelli, Simone Orcioni, Stefano Pirani, and Massimo Conti. 2016. Fall Detection Using Wearable Accelerometers and Smartphone. In *Mobile Networks for Biometric Data Analysis.* Springer, 299–311.

[6] Eduardo Casilari, Rafael Luque, and María-José Morón. 2015. Analysis of android device-based solutions for fall detection. *Sensors* 15, 8 (2015), 17827–17894.

[7] Gulf Coast Data Concepts. 2012. *Accelerometer/magnetometer data logger X8M-3mini.* http://www.gcdataconcepts.com/x8m-3mini.html.

[8] Department of Economic and Social Affairs in Population Division of the United Nations. 2015. World Population Ageing 2015. (2015). www.un.org/en/development/desa/population/.../pdf/ageing/WPA2015_Report.pdf.

[9] Mohammad Ashfak Habib, Mas S Mohktar, Shahrul Bahyah Kamaruzzaman, Kheng Seang Lim, Tan Maw Pin, and Fatimah Ibrahim. 2014. Smartphone-based solutions for fall detection and prevention: challenges and open issues. *Sensors* 14, 4 (2014), 7181–7208.

[10] Matt Herman, Elaine Gallagher, and Victoria Janice Scott. 2006. *The Evolution of Seniors' Falls Prevention in British Columbia: Working strategically and collectively to reduce the burden and impact of falls and fall-related injury among seniors.* British Columbia, Ministry of Health.

[11] Quoc T Huynh, Uyen D Nguyen, Lucia B Irazabal, Nazanin Ghassemian, and Binh Q Tran. 2015. Optimization of an accelerometer and gyroscope-based fall detection algorithm. *Journal of Sensors* 2015 (2015).

[12] Raul Igual, Carlos Medrano, and Inmaculada Plaza. 2013. Challenges, issues and trends in fall detection systems. *Biomedical engineering online* 12, 1 (2013), 66.

[13] Maarit Kangas, Antti Konttila, Ilkka Winblad, and Timo Jamsa. 2007. Determination of simple thresholds for accelerometry-based parameters for fall detection. In *Engineering in Medicine and Biology Society, 2007. EMBS 2007. 29th Annual International Conference of the IEEE.* IEEE, 1367–1370.

[14] Maarit Kangas, Irene Vikman, Jimmie Wiklander, Per Lindgren, Lars Nyberg, and Timo Jämsä. 2009. Sensitivity and specificity of fall detection in people aged 40 years and over. *Gait & posture* 29, 4 (2009), 571–574.

[15] Pekka Kannus, Harri Sievänen, Mika Palvanen, Teppo Järvinen, and Jari Parkkari. 2005. Prevention of falls and consequent injuries in elderly people. *The Lancet* 366, 9500 (2005), 1885–1893.

[16] Lih-Jen Kau and Chih-Sheng Chen. 2015. A smart phone-based pocket fall accident detection, positioning, and rescue system. *IEEE journal of biomedical and health informatics* 19, 1 (2015), 44–56.

[17] Mary B King and Mary E Tinetti. 1995. Falls in Community-Dwelling Older Persons. *Journal of the American Geriatrics Society* 43, 10 (1995), 1146–1154.
[18] Qiang Li, John A Stankovic, Mark A Hanson, Adam T Barth, John Lach, and Gang Zhou. 2009. Accurate, fast fall detection using gyroscopes and accelerometer-derived posture information. In *Wearable and Implantable Body Sensor Networks, 2009. BSN 2009. Sixth International Workshop on*. IEEE, 138–143.
[19] Stephen R Lord, Catherine Sherrington, Hylton B Menz, and Jacqueline CT Close. 2007. *Falls in older people: risk factors and strategies for prevention*. Cambridge University Press.
[20] Yoosuf Nizam, Mohd Norzali Haji Mohd, and M Mahadi Abdul Jamil. 2016. A Study on Human Fall Detection Systems: Daily Activity Classification and Sensing Techniques. *International Journal of Integrated Engineering* 8, 1 (2016).
[21] Ahmet Turan Özdemir. 2016. An analysis on sensor locations of the human body for wearable fall detection devices: Principles and practice. *Sensors* 16, 8 (2016), 1161.
[22] Paola Pierleoni, Alberto Belli, Lorenzo Palma, Marco Pellegrini, Luca Pernini, and Simone Valenti. 2015. A high reliability wearable device for elderly fall detection. *IEEE Sensors Journal* 15, 8 (2015), 4544–4553.
[23] Laurence Z Rubenstein and Karen R Josephson. 2002. The epidemiology of falls and syncope. *Clinics in geriatric medicine* 18, 2 (2002), 141–158.
[24] Jeong T Ryu and Byung H Moon. 2016. Fall Detection Algorithm based on Peaks of Voltage Measurements from the Accelerometer. *Indian Journal of Science and Technology* 9, 45 (2016).
[25] Dennis Sprute, Aljoscha Pörtner, Alexander Weinitschke, and Matthias König. 2015. Smart Fall: Accelerometer-Based Fall Detection in a Smart Home Environment. In *International Conference on Smart Homes and Health Telematics*. Springer, 194–205.
[26] Allan Stisen, Henrik Blunck, Sourav Bhattacharya, Thor Siiger Prentow, Mikkel Baun Kjærgaard, Anind Dey, Tobias Sonne, and Mads Møller Jensen. 2015. Smart devices are different: Assessing and mitigating mobile sensing heterogeneities for activity recognition. In *Proceedings of the 13th ACM Conference on Embedded Networked Sensor Systems*. ACM, 127–140.
[27] Mary E Tinetti, Wen-Liang Liu, and Elizabeth B Claus. 1993. Predictors and prognosis of inability to get up after falls among elderly persons. *Jama* 269, 1 (1993), 65–70.
[28] Pranesh Vallabh, Reza Malekian, Ning Ye, and Dijana Capeska Bogatinoska. 2016. Fall detection using machine learning algorithms. In *Software, Telecommunications and Computer Networks (SoftCOM), 2016 24th International Conference on*. IEEE, 1–9.
[29] B Vellas, F Cayla, H Bocquet, F De Pemille, and JL Albarede. 1987. Prospective study of restriction of acitivty in old people after falls. *Age and Ageing* 16, 3 (1987), 189–193.
[30] Johan Wannenburg and Reza Malekian. 2016. Physical activity recognition from smartphone accelerometer data for user context awareness sensing. *IEEE Transactions on Systems, Man, and Cybernetics: Systems* (2016).
[31] Kerim Yildirim, Gokhan Ucar, Tahir Keskin, and Adnan Kavak. 2016. Fall Detection Using Smartphone-Based Application. *International Journal of Applied Mathematics, Electronics and Computers* 4, 4 (2016), 140–144.
[32] Jian Yuan, Kok Kiong Tan, Tong Heng Lee, and Gerald Choon Huat Koh. 2015. Power-efficient interrupt-driven algorithms for fall detection and classification of activities of daily living. *IEEE Sensors Journal* 15, 3 (2015), 1377–1387.