Hindawi Publishing Corporation
International Journal of Distributed Sensor Networks
Volume 2016, Article ID 2308183, 10 pages
http://dx.doi.org/10.1155/2016/2308183

Research Article

A Smart Device Enabled System for Autonomous Fall Detection and Alert

Jian He,[1,2] Chen Hu,[1] and Xiaoyi Wang[1,2]

[1]*School of Software Engineering, Beijing University of Technology, Beijing 100124, China*
[2]*Beijing Engineering Research Center for IoT Software and Systems, Beijing 100124, China*

Correspondence should be addressed to Jian He; jianhee@bjut.edu.cn

Received 31 July 2015; Revised 11 December 2015; Accepted 28 December 2015

Academic Editor: Yuanzhu Chen

The activity model based on 3D acceleration and gyroscope is created in this paper, and the difference between the activities of daily living (ADLs) and falls is analyzed at first. Meanwhile, the *k*NN algorithm and sliding window are introduced to develop a smart device enabled system for fall detection and alert, which is composed of a wearable motion sensor board and a smart phone. The motion sensor board integrated with triaxial accelerometer, gyroscope, and Bluetooth is attached to a custom vest worn by the elderly to capture the reluctant acceleration and angular velocity of ADLs in real time. The stream data via Bluetooth is then sent to a smart phone, which runs a program based on the *k*NN algorithm and sliding window to analyze the stream data and detect falls in the background. At last, the experiment shows that the system identifies simulated falls from ADLs with a high accuracy of 97.7%, while sensitivity and specificity are 94% and 99%, respectively. Besides, the smart phone can issue an alarm and notify caregivers to provide timely and accurate help for the elderly, as soon as a fall is detected.

1. Introduction

There are about 30% living people over the age of 65 who fall at least one time each year in the USA, and the prevalence of fall in the elderly is about 20% in China [1]. Falls and fall induced injuries account for over 80% of all injury-related hospital admissions among people over 65 [2]. Consequently, falls affect tens of millions of the elderly throughout the world. For example, falls among the elderly cost the National Health Service more than £4.6 million per day according to a report by the Centre for Social Justice UK [3]. Researches showed that the risk of hospitalization could be reduced by 26% and death by over 80% after fall event detection followed by immediate notification to caregivers [4]. Therefore, the high incidences of falls, combined with their associated costs, make it imperative to develop a reliable and effective fall detection solution.

Over the last decade, a variety of different methods were developed to automatically detect falls. They are categorized into three different classes depending on the deployed sensor technology, namely, vision-based sensors [5], ambient sensors [6], and wearable devices [7]. For example, Yu's team [8] developed a vision-based fall detection method by applying background subtraction to extract the foreground human body and postprocessing to improve the result, and information is fed into a directed acyclic graph SVM for posture recognition in order to detect a fall. Yazar et al. introduced vibration and PIR sensors and deployed winner-takes-all decision algorithm to detect fall [9]. However, both vision-based and ambient sensors have a constrained monitoring area and require installation, adjustment, and maintenance which can result in higher costs. Recently, technological advancements in the fields of electrical, mechanical, and computer engineering, particularly involving microelectromechanical systems (MEMS), have resulted in smaller and cheaper inertial sensors. An inertial sensor (such as 3D accelerometer, 3D gyroscope, and 3D compass) is as tiny as $5 * 5$ millimeters and is as cheap as one US dollar [7]. Therefore, it is widely used to develop wearable devices which allow the measurement of physical activity under real-life environment. This includes indoor and outdoor activities as well as recordings in very private areas like the bathroom or the toilet [10]. Meanwhile, smart phone integrated with inertia sensors is more and more popular; many works have been

done for the fall detection on the smartphone. For example, Bai et al. proposed a system based on a triaxis accelerometer embedded in a smart phone with global positioning system (GPS) function to detect falls [11]. But such systems face the relatively high energy consumption of current smart phones, and it is inconvenient for the elderly to take smart phone at any time.

This paper presents a smart device enabled fall detection solution by using a smart phone and a custom vest integrated with triaxial accelerometer and gyroscope. The system incorporates an array of features, such as sending alerts, shortest message service (SMS), and global positioning system (GPS) location for easy alerting and monitoring. In short, the system takes advantage of wearable device and smart phone and can provide the elderly with an unobtrusive fall detection.

The rest of this paper is organized as follows. Section 2 introduces the available technology for automatic fall detection based on inertial sensors. The methodology to deploy the system is discussed in Section 3. Section 4 presents the implementation of the system. The simulated experiment and its analysis are discussed in Section 5. The conclusion and future research are proposed in Section 6.

2. Related Work Based on Inertial Sensors

Different approaches have been explored to solve the fall detection problem by using inertial sensors. The majority of these approaches can be divided in two main types: threshold-based and machine learning (or data mining) [12]. Both types are based on features extracted from the recorded signals.

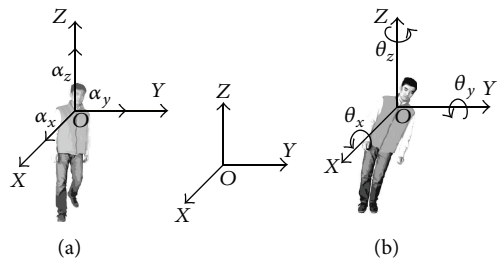
Threshold-based methods for fall detection use single or multiple thresholds on the extracted features. Bourke et al. [13] presented an approach to detect falls, which is based on an assumption that acceleration in falls is sharper than those in ADLs. Purwar et al. [14] used a triaxial accelerometer to set thresholds of acceleration and orientation of trunk through experiments to detect falls and achieved an accuracy of 81%. Lindemann et al. [15] integrated a triaxial accelerometer into a hearing aid device and used thresholds for acceleration and velocity to judge whether a fall had occurred. Noury et al. [16] developed a sensor with two orthogonally oriented accelerometers and used this sensor to monitor the inclination and inclination speed to detect falls. Though body orientation can improve the fall detection accuracy. Using one single device can only monitor the body orientation, and sufficient posture information cannot be collected using this method. Wang et al. [17] applied triaxial accelerometer and wireless sensor network to develop an enhanced fall detection system for the elderly monitoring. The main problem was that the use of only acceleration for fall detection led to many false positives. For instance, sitting down quickly produced similar vertical acceleration data. As a result, more and more researchers study technology of combining triaxial accelerometer with gyroscope to detect fall events accurately. Li et al. [18] proposed a system where two accelerometers are placed on the abdomen and the right thigh, and the data stream is segmented into one second window. The system could reduce both false alarms by deriving the posture information from both gyroscopes and accelerometers. Gjoreski et al. [19] introduced RAReFall which measures the difference between maximum value and minimum value within one-second window; if the difference is larger than 1 g and the maximum value is at the back of minimum value, then a fall is detected.

Machine learning techniques use automatic methods starting from the extracted features and try to differentiate between a fall and ADLs [20]. Ojetola et al. [3] introduce two sensor motes (each has one accelerometer and one gyroscope) that are wore on chest and right thigh to differentiate ADLs and fall. In the system, raw data is first processed by mean filter and lower resampling; then vector magnitude of acceleration and angular velocity are used as features to train a C4.5 Decision Tree model. However, the accuracy and timeliness were not mentioned. Zhang et al. [21] presented a fall detection method based on one-class support vector machine where a triaxial accelerometer is used to capture the movement data of human. This method needs specific activity patterns and computation, which is not appropriate for real-time and comfort fall detection. Tong et al. [22] used hidden Markov model (HMM) and triaxial accelerometer to detect and predict falls through analyzing the features of human motion series during fall processes. The experiment results showed that this method could predict falls in 200~400 ms before the impact and also accurately distinguish falls from other daily activities. However, the HMM λ and thresholds of the system were set based on the data samples of young people's simulated activities; the mathematical model and thresholds should be trained and reset based on the large real-world samples of the elderly. Dinh and Struck transform the acceleration data from Cartesian coordinates to spherical coordinates and develop the algorithm based on a fuzzy logic and a neural network to detect falls [23]. Gjoreski et al. [24] study a combination of body-worn inertial and location sensors, and the Random Forest classifier is introduced to detect falls. However, hybrid sensor approaches or location information increases energy consumption and data storage, which thereby limits recording duration and increases cost, size, and so forth.

Since wearable device based on inertial sensors is limited to its computing power, storage, and energy consumption, it is not suitable for running complex algorithm. Nevertheless, the smart phone has strong computing and communicating capability. So an innovative technology which takes advantage of wearable device and smart phone is introduced to provide the elderly with an unobtrusive fall detection.

3. Method and System Setup

Researches [25] studied acceleration of falls and activities of daily living (ADLs) from the waist, wrist, and head. The research data show that the upper trunk, which is below the neck and above the waist, is the most suitable feature region for distinguishing falls from other movements using acceleration. Meanwhile, in order to reduce the inconvenience caused by wearable device, the motion sensor board is put on the top of the custom vest to capture the activities of the individual in this paper.



(a) (b)

Figure 1: Definition of coordinate systems. (a) The sensitivity axes of the accelerometer. (b) The sensitivity axes of the angular velocity.

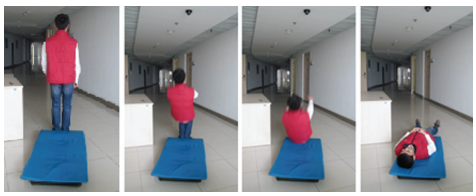


Figure 2: Bw-Fall.

3.1. Activity Model. In the process of human motion, the acceleration and the deflection angle vary in real time. So the upper trunk Cartesian coordinate system *oxyz*, whose origin is close to the neck of human body, is paralleled with the geodetic coordinate system *OXYZ* as shown in Figure 1. Accelerations along *x*-, *y*-, and *z*-axis are denoted as α_x, α_y, and α_z, respectively. The resultant acceleration can be calculated in

$$\alpha = \sqrt{\alpha_x^2 + \alpha_y^2 + \alpha_z^2}. \quad (1)$$

The resultant angular velocity (ω) can be calculated in (2). ω_x, ω_y, and ω_z are the angular velocity along the *x*-, *y*-, and *z*-axis. In general falls, ω_z is very small:

$$\omega = \sqrt{\omega_x^2 + \omega_y^2 + \omega_z^2}. \quad (2)$$

Falls are usually characterized by rapid acceleration and angular velocity. In order to find out the differences between ADLs and falls, 3 typical subcategories of ADLs, walking-turning-walking (W-T-W), sitting down-standing up (Sd-Su), squatting-standing (Sq-Su), are analyzed and compared with two types of falls: Bw-Fall means backward falling without recovery in 2 seconds (as shown in Figure 2), and Sd-Fall means falling either to the left or to the right side (as shown in Figure 3).

Figure 4 shows the resultant acceleration and angular velocity curves from each kind of motion process. Both the resultant acceleration and angular velocity are normalized treatment so as to simplify the expression. The horizontal axis is for time in unit of 0.1 s, while the longitudinal axis is for the resultant acceleration or angular velocity.

As can be seen from Figure 4, falls are usually characterized by rapid acceleration and great angular velocity, and they could be identified from ADLs as long as proper features are extracted.

Algorithm Sliding-Window
Input: Sensor data stream
Output: Type(label) of a slide instance
(1) label =
(2) $S_{\text{width}} = 20$, // set the width of sliding window
(3) **for** ($S_{\text{ref}} = 0$; size($S_{\text{ref}} + S_{\text{width}}$) $\geq S_{\text{width}}$; t++)
(4) label = kNN(D_{train}, $S_{\text{ref}} + S_{\text{width}}$, k)
(5) **end for**
(6) return label;

Algorithm 1: Sliding window algorithm pseudocode.

3.2. Data Processing and kNN Algorithm. In the process of human motion, the reluctant acceleration and angular velocity vary real-timely and then make up stream data. It faces great challenges to classify stream data because of its infinite length. Hence, sliding window which just takes the last seen N elements of the stream into account is introduced to maintain similarity queries over stream data.

Figure 5 illustrates the conventions that new data elements are coming from the right and the elements at the left are ones already seen. The sliding window covers a time period of $T_S \times n$, where T_S is the same sampling period. Each element of sensor data stream has an arrival time, which increments by one at each arrival, with the leftmost element considered to have arrived at time 1, since the duration of fall is less than 2 seconds, and the sample period is 0.1 second. So n is set to 2, and the width of the sliding window is 20.

For an illustration of this notation, consider the situation presented in Figure 5. The start time of the sliding window is 17, the current time instant is 36, and the last seen element of the stream data is e_{36}. Each element e_i consists of the resultant acceleration and angular velocity collected by sensors at time i.

Based on such sliding window, two kinds of features are selected to classify falls from ADLs. The first one (namely, α) is the set of 20 reluctant accelerations, and the other one (namely, ω) is the set of 20 reluctant angular velocities. Meanwhile kNN algorithm is used as a classification model. Algorithm 1 represents the program code for how the sliding window slides through the data stream. D_{train} is the training dataset for fall patterns.

kNN algorithm is introduced to measure the difference or similarity between instances according to a distance function.

(a) (b)

Figure 3: Sd-Fall. (a) Right side fall. (b) Left side fall.

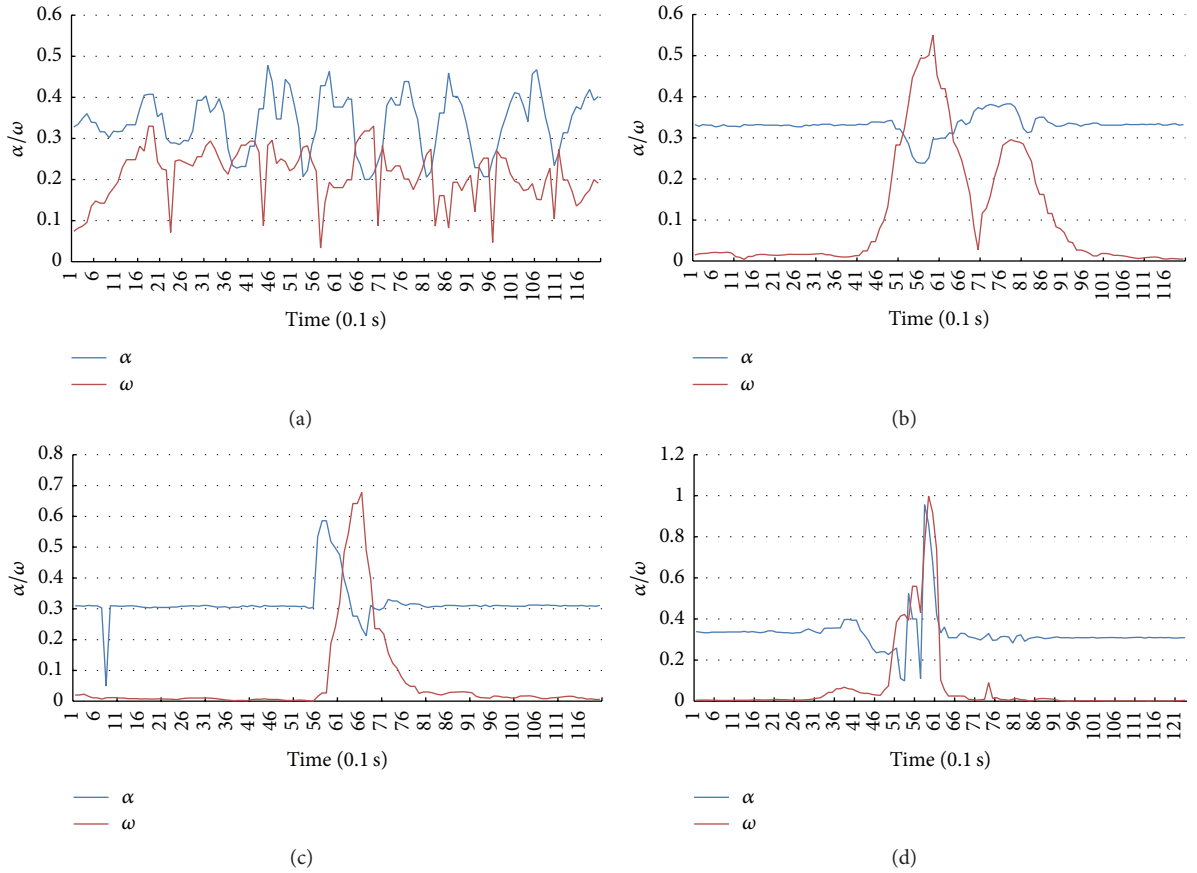


Figure 4: Data examples (the resultant of acceleration and angular velocity) of ADLs and fall. (a) Process of W-T-W. (b) Process of Sd-Su. (c) Process of Sq-Su. (d) Process of Bw-Fall.

Given a test instance x, its k closest neighbors, $y_1, \ldots, y_k$, are calculated, and a vote is conducted to assign the most common class to x. That is, the class of x, denoted by $c(x)$, is determined as follows [26]:

$$c(x) = \arg\max \sum_{n=1}^{k} \delta(c, c(y_i)), \quad c \in C, \tag{3}$$

where $c(y_i)$ is the class of y_i and δ is a function that $\delta(u, v) = 1$ if $u = v$.

Since there are two kinds of features (namely, α and ω) used for classifying, Euclidean distance defined in formula (4) is selected as the distance function. Among formula (4), $D(x, t)$ is the Euclidean distance, α_x is a test instance, ω_t is a training instance, and both of them are 2-dimensional real vector:

$$D(x,t) = \sqrt{(\alpha_{x1} - \alpha_{t1})^2 + (\omega_{x1} - \omega_{t1})^2 + \cdots + (\alpha_{x20} - \alpha_{t20})^2 + (\omega_{x20} - \omega_{t20})^2}. \tag{4}$$

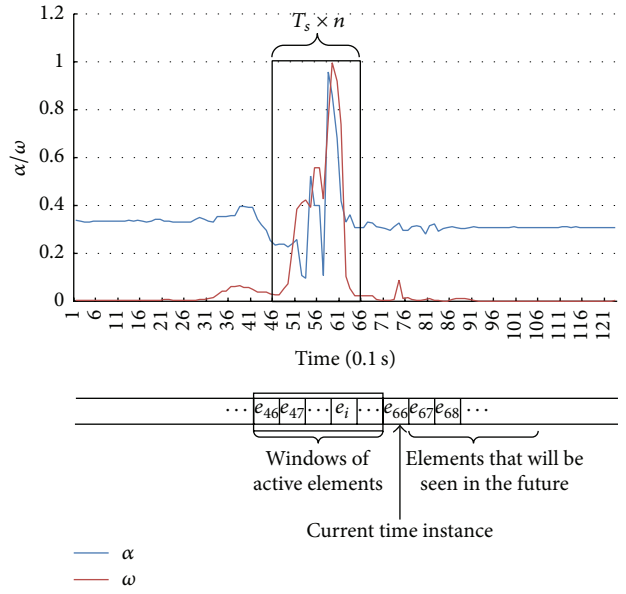


FIGURE 5: Illustration for the notation and the conventions of sliding window.

4. Implementation

Since most smart devices (such as smart phone and pad) are also integrated with the Bluetooth module and have strong computing capability, a smart phone integrated with Bluetooth is introduced to receive the stream data from sensors, and program based on the above technologies is developed to detect fall and issue alarm.

4.1. System Architecture. Figure 6 shows the architecture of the system. The system mainly consists of a custom vest and a smart phone running the fall detection program. First the mainboard reads the individual accelerations and angular velocities from the accelerometer and gyroscope. Then it calculates the resultant acceleration and angular velocity and sends the data to the smart phone via the Bluetooth. After getting the data from the mainboard, the fall detection program judges whether the individual is falling or not based on the kNN algorithm presented in Section 2. The phone can make a call or send a message with GPS position to a healthcare center or family member as soon as it detects a fall. This then provides a timely warning that a fall has occurred.

4.2. Data Acquisition. The sensor board measures 24 mm × 38 mm × 9 mm (width × length × thickness), which is suitable for use in a vest. It consists of a high-performance, low-power microcontroller and a class 2 Bluetooth module. The Bluetooth module has a range of 10 m and a default transmission rate of 115 k baud. The microcontroller reads the data from the accelerometer and gyroscope and calculates the resultant acceleration and angular velocity and transmits them to the smart phone. The triaxial accelerometer has a range of ±16 g. The triaxial gyroscope has a full-scale range of ±2000°/sec.

Ten healthy individuals (5 males and 5 females, aged from 20 to 45) are asked to perform the intentional falls and ADLs both indoors and outdoors so as to get the training dataset. There are three kinds of ADLs (namely, W-T-W, Sd-Su, and Sq-Su) and fall. Each one includes 5 sets of resultant acceleration and angular velocity. As a result, there are a total of 200 sets of training samples in the training dataset.

4.3. Software Design. The software includes the on-chip program inserted into the custom vest and the fall detection program app downloaded onto the smart phone. The key steps of the on-chip program are as follows:

(A) Initialize the triaxial accelerator and gyroscope and set the frequency for viewing the triaxial acceleration and angular velocity and the baud rate for Bluetooth.

(B) Monitor the accelerations and angular velocities from triaxial accelerometer sensor and gyroscope in interval of 0.1 s.

(C) Calculate the resultant acceleration and angular velocities according to the data from triaxial accelerator and gyroscope.

(D) Send the resultant acceleration and angular velocities to the smart phone via Bluetooth.

After getting the training dataset, the kNN classification algorithm can be easily adapted for fall detection program. According to the instance of a sliding window from the input stream, the similarity between the instance and training sample in the training dataset is calculated using Euclidean distance function. If the similarity score of all NNs (from 1 to k) of the instance is voted to fall label, then it is classified as a fall pattern immediately. Otherwise, it is not classified as a fall pattern. The pseudocode for the adapted kNN algorithm is presented in Algorithm 2.

5. Experiment

Since it is very dangerous for the elderly to test falls, there is not any experiment on the elderly over 50 years old. Fifteen healthy individuals, including 10 males and 5 females, aged from 20 to 45 years, are asked to perform the simulated falls and normal ADLs indoors and outdoors. The average height and mass of volunteers are 172.3 cm and 64.5 kg, respectively. According to the fall simulation protocol [27], fall simulation is conducted onto a 15 cm thick spongy cushion (hardness = 4 kPa, pressure to compass a piece of foam by 35% of its original height) to reduce the impact. Participants stand at a distance of 1.5 times the lengths of their foot apart from the spongy cushion and are instructed to do Sd-Fall (or Bw-Fall) like a frail old person, and there are no warm-up trials to familiarize with spongy cushion. In order to evaluate the fall detection system, the sensitivity and specificity are introduced.

5.1. Experiment Results. The experiment includes 100 simulated falls, 100 W-T-W, 100 Sd-Su, and 100 Sq-Su. The experiment results are shown in Table 1. Most samples are

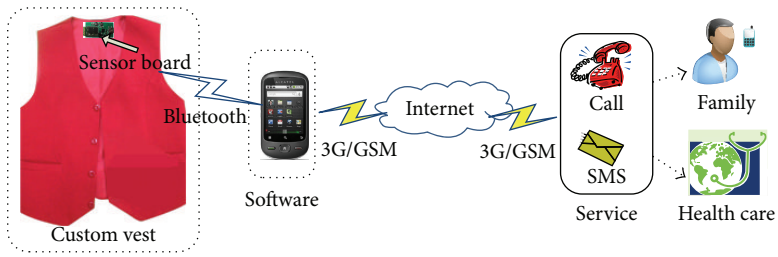


Figure 6: The system architecture.

Algorithm k-Nearest Neighbour
Input: $D_{\text{train}}\{((\alpha 1_1, \omega 1_1), \ldots, (\alpha 1_{20}, \omega 1_{20}), C_1), \ldots,$
$((\alpha n_1, \omega n_1), \ldots, (\alpha n_{20}, \omega n_{20}), C_n)\}$ // training data set
$S = ((\alpha s_1, \omega s_1), \ldots, (\alpha s_{20}, \omega s_{20}))$ // sliding window instances // to be classified
k // number of nearest neighbour
Output: Label $= \{(\alpha s_1, \omega s_1), \ldots, (\alpha s_{20}, \omega s_{20}), C_i\}$ // labelled test set
(1) for $(i = 0; n; i{+}{+})$ $Ni = 0$;
(2) for each labelled instance
$((\alpha i_1, \omega i_1), \ldots, (\alpha i_{20}, \omega i_{20}), C_i)$ do
(3) calculate $\text{sim}_i = \text{sim}(((\alpha i_1, \omega i_1), \ldots, (\alpha i_{20}, \omega i_{20})), ((\alpha s_1, \omega s_1), \ldots, (\alpha s_{20}, \omega s_{20})))$;
(4) Order sim_i from low to highest, $(i = 1, \ldots, N)$
(5) end for
(6) Select the k nearest instances to $sj : D^k_{sj}$
(7) for each D^k_{sj} do
(8) if $D^k_{sj} = C_i$ then $N_i{+}{+}$
(9) end for
(10) if $\max(N_1, \ldots, N_n) = N_i$ then
(11) Label $= \{(\alpha s_1, \omega s_1), \ldots, (\alpha s_{20}, \omega s_{20}), C_i\}$
(12) return Label

Algorithm 2: Pseudocode for the kNN classifier algorithm.

Table 1: Experiment results.

Tests	Total	Correct	Wrong	Accuracy
W-T-W	100	100	0	100%
Sd-Su	100	99	1	99%
Sq-Su	100	98	2	98%
Fall	100	94	6	94%

detected successfully; only a few samples are undetected. Table 1 shows the statistic for the test samples. It can be calculated that the accuracy is 97.7%, while the sensitivity and specificity are 94% and 99%, respectively. It proves that the algorithm coupled with accelerometers and gyroscopes reduces both false positives and false negatives, while improving fall detection accuracy.

Compared with the thresholds method using accelerations or gyroscopes at several single time points, the technology in this paper is an effective method for human fall detection. Most thresholds methods use the results of sensing information at uncontinuous time points to detect falls; thus some misdetection may be caused by the incomplete information in some experiments. For example, Wang et al. [17] achieved sensitivity of 91% and specificity of 92%. In this paper, the new method analyzes the steam data of accelerations and angular velocities during the whole course of human fall process in a 2 s sliding window, so more complete sensing information is used to study the features of human fall process. As a result, the experiment shows better results.

Report shows that approximately 3% of all fallers lie for more than 20 min without external support, and 80% of the fallers aged 90 years or older are unable to get up by themselves [27]. Hence, an autonomous notification to caregiver after detecting fall will be greatly helpful for the elderly by reducing the time between the fall and the arrival of medical attention. The kNN-based program running on smart phone takes advantage of several smart phone components (such as XML file, phone, and GPS) to provide a convenient and efficient alerting service. For example, Figure 7 shows the interface where user can set the interval time for autonomous notification to caregiver after detecting fall and can configure different warning methods on the smart phone. All of the configurations are stored into an XML file. Each of the detected falls triggers an alarm. If the user could not stop the alarm in the interval time, a call notifying caregiver is made, or an emergency message with GPS location is immediately sent to caregivers, so as to provide a timely and accurate help.

Table 2: General comparison of different learning algorithms.

Algorithm	Correctness (%)	Sensitivity (%)	Specificity (%)	Times (s)
kNN (K = 3)	97.8548	93.8	99.1	<0.01
kNN (K = 7)	97.8548	93.8	99.1	<0.01
kNN (K = 7)	97.6898	93.2	99.1	<0.01
Native Bayes	97.5248	95.2	98.3	0.01
Bayes Net	96.2046	91.8	97.6	0.04
ANN	97.5248	93.8	98.7	0.44
Decision Tree (J48)	96.5347	91.1	98.3	0.08
Bagging	96.3696	91.8	97.8	0.03
Ripper	97.0297	93.8	98	0.03

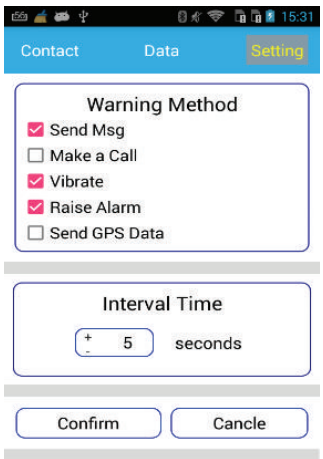


Figure 7: Option menu to configure warning messages.

Figure 8: An alarm message with GPS location.

Figure 8 shows an example about an alerting message with GPS location when Mr. He fell down and could not stop the alarm in the interval time.

5.2. Comparison of Different Learning Algorithms. The Weka which integrates with various machine learning algorithms for data mining is introduced to compare kNN with other learning algorithms according to the same training dataset and experimental data. A Lenovo ThinkCenter m6200t with an i5 CPU and 4 G memory is selected to run Weka. Table 2 shows the comparison of different learning algorithms. It can be seen that kNN algorithm has the most correctness, and its running time is much less than 0.01 seconds. The sensitivity of kNN while k equals 7 is less than others (i.e., k equals 3 or 5), and they have the same specificity. Besides, native Bayes algorithm has the most sensitivity (namely, 95.2%), but its specificity is less than kNN, and its running time is about 0.01 seconds. The correctness of ANN is about 97.5%; it takes more than 0.44 seconds to run ANN algorithm.

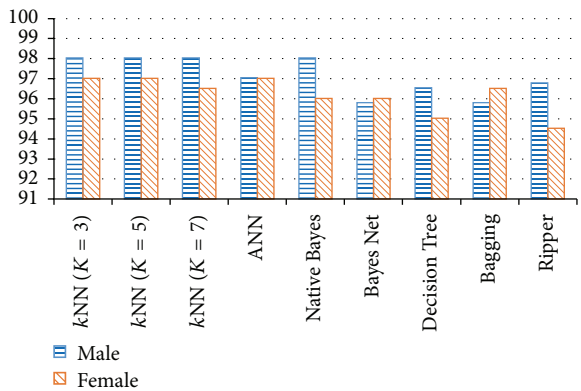


Figure 9: Correctness comparison of different learning algorithms between male and female.

Figure 9 shows the correctness comparison of different learning algorithms between male and female. It can be seen that the male's correctness is usually higher than female's correctness except the Bayes Net and Bagging algorithms. Figure 10 shows the correctness comparison of different learning algorithms in different ages. It can be seen that kNN algorithms have higher accuracy in those aged 21–25 and 36–40. ANN algorithm has higher accuracy in those aged 21–25, 26–30, and 41–45 than in those aged 31–35 and 41–45. There is not any clearly accurate difference on Decision Tree algorithm among different ages. Since kNN is the most efficient learning algorithm with highest accuracy, it is quite suitable for running on smart phone.

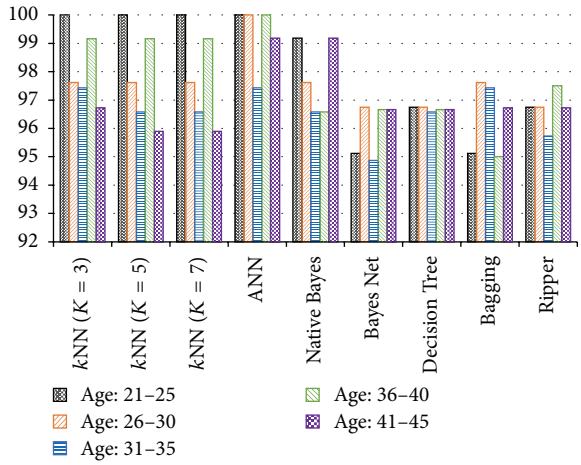


Figure 10: Correctness comparison of different learning algorithms in different ages.

5.3. Discussion. Because of ethical constraints, a spongy cushion is used to avoid injuries to the participants during the simulated fall. While many (injurious) falls occur on hard materials (e.g., tiles), the spongy cushion can absorb the impact. As a result, the acceleration values of the simulated falls could not reflect a real-world fall. Besides, though there is not any warm-up trials for anticipants to familiarize with spongy cushion, the anticipants know that they will fall. This leads to anticipation that the anticipants may change postural control and response mechanisms. Just as Bagalà et al. outlined, algorithms calculated from fall simulations in healthy young subjects lack the necessary accuracy requirements for real-world fall detection [28].

Due to access problems to the elderly and other difficulties (e.g., cost and adherence), the number of recorded, documented, and published real-world fall data of older people is minimal [29]. Bourke et al. made a systematic review of a total number of 96 articles on fall detection with body-worn sensors published between 1998 and 2012. It showed that less than 7% of studies have used fall data recorded from elderly people in real life, and simulated fall data were used in 90 (93.8%) studies. However, recently the FARSEEING is a European collaborating project, and one goal of the project is to generate a large metadatabase of real-world fall signals [30]. We plan to verify our system as soon as we can access the database of real-world falls.

In Schwickert et al.'s review [29], most of the sensors were placed at waist, chest, thorax, and trunk. In order to reduce the inconvenience for anticipants caused by wearable device and protect the motion sensor board from being broken, it is put on the top of the custom vest. Figure 11 shows that this results in higher impact signals than of those which the motion sensor board is putting at waist.

Since our fall detection system aims to detect fall and issue alarm, we do not divide the fall process into multiple phases, such as prefall phase, fall phase, and recovery phase. The system will just issue an alarm as soon as it detects fall, and a call to notify caregiver is made, or an emergency message with GPS location is sent to caregiver, as long as the user does not stop the alarm in the interval time.

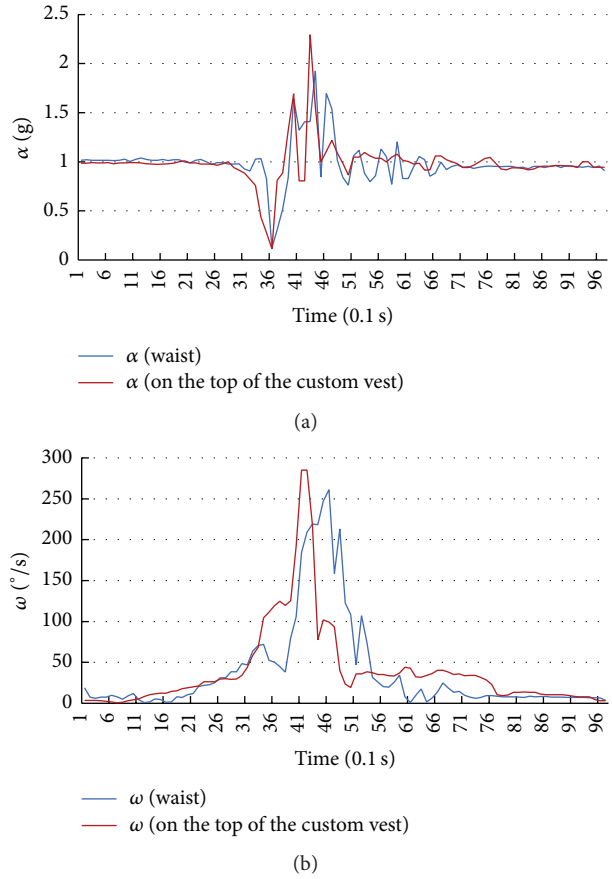


Figure 11: The comparison between the impact signal from the motion sensor put on the top of vest and of those put at waist. (a) The comparison on resultant acceleration. (b) The comparison on resultant angular velocity.

6. Conclusion and Future Work

Taking into account the results and analysis provided above, it can be concluded that the proposed system is able to detect simulated falls with sufficient accuracy and can provide timely help for the elderly.

In China, there is not any available data of real-world fall as well nowadays. Based on the encouraging results achieved, we will cooperate with community geracomium, supply our motion sensor board and software to the elderly without compensation, so as to collect data of the daily activities, and harvest the database of real-world falls. After getting the database of real-world falls, our fall detection system will be verified and improved. Besides, we will study different important aspects of a fall event according to the multiphase model of fall supposed by Klenk et al. [27]. For example, we will study algorithm to predict fall according to the research of the prefall activity.

Conflict of Interests

The authors declare that there is no conflict of interests regarding the publication of this paper.

Acknowledgments

This work was supported by the Beijing Natural Science Foundation under Grant no. 4102005, was partly supported by the National Nature Science Foundation of China (no. 61040039), and was supported by Beijing Science and Technology Innovation Platform Program (no. PXM2015014204500211).

References

[1] M. Gao and Y. Song, "Meta-analysis of the prevalence of fall in elderly in China," *Beijing Medical Journal*, vol. 36, no. 10, pp. 796–798, 2014.

[2] P. Kannus, J. Parkkari, S. Koskinen et al., "Fall-induced injuries and deaths among older adults," *The Journal of the American Medical Association*, vol. 281, no. 20, pp. 1895–1899, 1999.

[3] O. Ojetola, E. I. Gaura, and J. Brusey, "Fall detection with wearable sensors—safe (smart fall detection) ," in *Proceedings of the 7th IEEE International Conference on Intelligent Environments (IE '11)*, pp. 318–321, Nottingham, UK, July 2011.

[4] N. Noury, P. Rumeau, A. K. Bourke, G. ÓLaighin, and J. E. Lundy, "A proposal for the classification and evaluation of fall detectors," *IRBM*, vol. 29, no. 6, pp. 340–349, 2008.

[5] D. Anderson, J. M. Keller, M. Skubic, X. Chen, and Z. He, "Recognizing falls from silhouettes," in *Proceedings of the 28th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBS '06)*, pp. 6388–6391, New York, NY, USA, September 2006.

[6] Y. Zigel, D. Litvak, and I. Gannot, "A method for automatic fall detection of elderly people using floor vibrations and sound—proof of concept on human mimicking doll falls," *IEEE Transactions on Biomedical Engineering*, vol. 56, no. 12, pp. 2858–2867, 2009.

[7] A. Buke, F. Gaoli, W. Yongcai, S. Lei, and Y. Zhiqi, "Healthcare algorithms by wearable inertial sensors: a survey," *China Communications*, vol. 29, no. 4, pp. 7–15, 2015.

[8] M. Yu, A. Rhuma, S. M. Naqvi, L. Wang, and J. Chambers, "A posture recognition-based fall detection system for monitoring an elderly person in a smart home environment," *IEEE Transactions on Information Technology in Biomedicine*, vol. 16, no. 6, pp. 1274–1286, 2012.

[9] A. Yazar, F. Erden, and A. E. Cetin, "Multi-sensor ambient assisted living system for fall detection," in *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP '14)*, pp. 1–3, May 2014.

[10] C. Becker, L. Schwickert, S. Mellone et al., "Proposal for a multiphase fall model based on real-world fall recordings with body-fixed sensors," *Zeitschrift fur Gerontologie und Geriatrie*, vol. 45, no. 8, pp. 707–715, 2012.

[11] Y.-W. Bai, S.-C. Wu, and C.-L. Tsai, "Design and implementation of a fall monitor system by using a 3-axis accelerometer in a smart phone," *IEEE Transactions on Consumer Electronics*, vol. 58, no. 4, pp. 1269–1275, 2012.

[12] N. Pannurat, S. Thiemjarus, and E. Nantajeewarawat, "Automatic fall monitoring: a review," *Sensors*, vol. 14, no. 7, pp. 12900–12936, 2014.

[13] A. K. Bourke, P. van de Ven, M. Gamble et al., "Evaluation of waist-mounted tri-axial accelerometer based fall-detection algorithms during scripted and continuous unscripted activities," *Journal of Biomechanics*, vol. 43, no. 15, pp. 3051–3057, 2010.

[14] A. Purwar, D. U. Jeong, and W. Y. Chung, "Activity monitoring from real-time triaxial accelerometer data using sensor network," in *Proceedings of the IEEE International Conference on Control, Automation and Systems (ICCAS '07)*, pp. 2402–2406, IEEE, Seoul, The Republic of Korea, October 2007.

[15] U. Lindemann, A. Hock, M. Stuber, W. Keck, and C. Becker, "Evaluation of a fall detector based on accelerometers: a pilot study," *Medical and Biological Engineering and Computing*, vol. 43, no. 5, pp. 548–551, 2005.

[16] N. Noury, P. Barralon, G. Virone, P. Boissy, M. Hamel, and P. Rumeau, "A smart sensor based on rules and its evaluation in daily routines," in *Proceedings of the 25th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBS '03)*, vol. 4, pp. 3286–3289, IEEE, Cancún, Mexico, September 2003.

[17] J. Wang, Z. Zhang, B. Li, S. Lee, and R. Sherratt, "An enhanced fall detection system for elderly person monitoring using consumer home networks," *IEEE Transactions on Consumer Electronics*, vol. 60, no. 1, pp. 23–29, 2014.

[18] Q. Li, J. A. Stankovic, M. A. Hanson, A. T. Barth, J. Lach, and G. Zhou, "Accurate, fast fall detection using gyroscopes and accelerometer-derived posture information," in *Proceedings of the 6th International Workshop on Wearable and Implantable Body Sensor Networks (BSN '09)*, pp. 138–143, Berkeley, Calif, USA, June 2009.

[19] H. Gjoreski, S. Kozina, M. Gams, and M. Lustrek, "RAReFall—real-time activity recognition and fall detection system," in *Proceedings of the IEEE International Conference on Pervasive Computing and Communication Workshops (PERCOM WORKSHOPS '14)*, pp. 145–147, IEEE, Budapest, Hungary, March 2014.

[20] A. T. Özdemir and B. Barshan, "Detecting falls with wearable sensors using machine learning techniques," *Sensors*, vol. 14, no. 6, pp. 10691–10708, 2014.

[21] T. Zhang, J. Wang, L. Xu, and P. Liu, "Fall detection by wearable sensor and one-class SVM algorithm," in *Intelligent Computing in Signal Processing and Pattern Recognition*, vol. 345 of *Lecture Notes in Computer Science*, pp. 858–886, Springer, Berlin, Germany, 2006.

[22] L. Tong, Q. Song, Y. Ge, and M. Liu, "HMM-based human fall detection and prediction method using tri-axial accelerometer," *IEEE Sensors Journal*, vol. 13, no. 5, pp. 1849–1856, 2013.

[23] C. Dinh and M. Struck, "A new real-time fall detection approach using fuzzy logic and a neural network," in *Proceedings of the 6th International Workshop on Wearable Micro & Nano Technologies for Personalized Health*, pp. 57–60, Oslo, Norway, June 2009.

[24] H. Gjoreski, M. Luštrek, and M. Gams, "Context-based fall detection using inertial and location sensors," in *Ambient Intelligence*, vol. 7683 of *Lecture Notes in Computer Science*, pp. 1–16, Springer, Berlin, Germany, 2012.

[25] A. K. Bourke and G. M. Lyons, "A threshold-based fall-detection algorithm using a bi-axial gyroscope sensor," *Medical Engineering and Physics*, vol. 30, no. 1, pp. 84–90, 2008.

[26] S. Z. Erdogan and T. T. Bilgin, "A data mining approach for fall detection by using k-nearest neighbour algorithm on wireless

sensor network data," *IET Communications*, vol. 6, no. 18, pp. 3281–3287, 2012.

[27] J. Klenk, C. Becker, F. Lieken et al., "Comparison of acceleration signals of simulated and real-world backward falls," *Medical Engineering and Physics*, vol. 33, no. 3, pp. 368–373, 2011.

[28] F. Bagalà, C. Becker, A. Cappello et al., "Evaluation of accelerometer-based fall detection algorithms on real-world falls," *PLoS ONE*, vol. 7, no. 5, Article ID e37062, 2012.

[29] L. Schwickert, C. Becker, U. Lindemann et al., "Fall detection with body-worn sensors: a systematic review," *Zeitschrift für Gerontologie und Geriatrie*, vol. 46, no. 8, pp. 706–719, 2013.

[30] A. K. Bourke, J. Klenk, L. Schwickert et al., "Temporal and kinematic variables for real-world falls harvested from lumbar sensors in the elderly population," in *Proceedings of the 37th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC '15)*, pp. 5183–5186, IEEE, Milan, Italy, August 2015.