

## Research Article

# GBDT-Based Fall Detection with Comprehensive Data from Posture Sensor and Human Skeleton Extraction

Wen-Yu Cai <sup>1,2</sup>, Jia-Hao Guo,<sup>1</sup> Mei-Yan Zhang <sup>3</sup>, Zhi-Xiang Ruan,<sup>1</sup> Xue-Chen Zheng,<sup>1</sup> and Shuai-Shuai Lv <sup>1</sup>

<sup>1</sup>College of Electronics and Information, Hangzhou Dianzi University, Hangzhou 310018, China

<sup>2</sup>Zhejiang Provincial Key Lab of Equipment Electronics, Hangzhou Dianzi University, Hangzhou 310018, China

<sup>3</sup>School of Electrical Engineering, Zhejiang University of Water Resources and Electric Power, Hangzhou 310018, China

Correspondence should be addressed to Mei-Yan Zhang; [meiyan19831109@163.com](mailto:meiyan19831109@163.com)

Received 11 March 2020; Revised 26 May 2020; Accepted 2 June 2020; Published 25 June 2020

Academic Editor: Fabrizio Taffoni

Copyright © 2020 Wen-Yu Cai et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Since fall is happening with increasing frequency, it has been a major public health problem in an aging society. There are considerable demands to distinguish fall down events of seniors with the characteristics of accurate detection and real-time alarm. However, some daily activities are erroneously signaled as falls and there are too many false alarms in actual application. In order to resolve this problem, this paper designs and implements a comprehensive fall detection framework on the basis of inertial posture sensors and surveillance cameras. In the proposed system framework, data sources representing behavior characteristics to indicate potential fall are derived from wearable triaxial accelerometers and monitoring videos of surveillance cameras. Moreover, the NB-IoT based communication mode is adopted to transmit wearable sensory data to the Internet for subsequent analysis. Furthermore, a Gradient Boosting Decision Tree (GBDT) classifier-based fall detection algorithm (GBDT-FD in short) with comprehensive data fusion of posture sensor and human video skeleton is proposed to improve detection accuracy. Experimental results verify the good performance of the proposed GBDT-FD algorithm compared to six kinds of existing fall detection algorithms, including SVM-based fall detection, NN-based fall detection, etc. Finally, we implement the proposed integrated systems including wearable posture sensors and monitoring software on the Cloud Server.

## 1. Introduction

An increased aging population in the world is forcing rapid rises in healthcare requirements [1]. Everyone knows that older people have poor balance ability and slow response ability. Falls are a major cause of injury for the elderly and a huge obstacle in the independent living of the seniors. Once the elderly falls down alone at home without help, the injured elderly may be lying on the ground for several hours or even days. More seriously, it is very likely to extended injury and be life-threatening if he did not get treatment timely. Therefore, timely fall incident detection and medical assistance for the elderly are intuitively important. However, due to different application scenarios and various body activities, satisfactory and reliable fall detection results are too hard to guarantee [2].

Some related fall detection algorithms have been proposed in the literature. Broadley et al. [3] review the latest reported

systems on activity monitoring of humans based on wearable sensors and issues to be addressed to tackle the challenges. As far as we know, there are three main categories of fall detection technologies: fall detection using wearable sensors [4, 5], fall detection using environmental sensors [6, 7], and video-based fall detection [8, 9]. Although there are some other methods such as radar-based fall detection [10], they are more complicated compared to the above three methods.

Wearable sensor based fall detection methods mainly depend on sensory data gathered from wearable accelerometer and gyroscope. It is generally agreed that the use of wearable sensors has played a quite important role in monitoring the physiological parameters of a person to minimize any malfunctioning happening in the body. In recent years, the advancement of sensing technologies, embedded systems, wireless communication technologies, nanotechnologies, and miniaturization makes it possible to

develop smart wearable sensors to monitor activities of human beings continuously. Nag et al. [11] provide a review on some of the significant research work done on wearable flexible sensors. Chen et al. [12] propose a novel intelligent fall detection method, named as ESAEs-OCCCH, which uses acceleration data from a wrist-worn smart watch. ESAEs-OCCCH is first adopted for unsupervised feature extraction to overcome the disadvantages of artificial feature extraction. Yacchirema et al. [13] propose an innovative IoT (Internet of Thing) based online system for detecting falls of the aged. Sensory readings are processed and analyzed using a decision tree based Big Data model running on a Smart IoT Gateway [14]. Although these wearable sensors have high sensitivity and good real-time characteristics, higher detection accuracy cannot typically be achieved due to the interference from diverse activities of hand or wrist. Hence, it is easy to cause misjudgment and missed detection of fall actions relying only on wearable sensory data.

Secondly, a few scholars apply environmental sensors to detect falls. Li et al. [15] propose a phase transform (SRP-PHAT) method which can locate the original source of a certain voice. In terms of sound classification phase, they apply the Mel-Frequency Cepstral Coefficients (MFCC) features with a Nearest Neighbor (NN) approach to improve fall detection performance. However, expensive acoustic devices have high requirements on the environment, and it is not feasible to promote accurate detection with certain ambient noise. Adnan et al. [16] adopt acoustic Local Ternary Patterns (acoustic-LTPs) to detect fall events by analyzing environmental noise. Acoustic features are extracted from the separated source components using the proposed acoustic-LTPs scheme. Subsequently, fall events would be identified with SVM based classifier. However, it will cause noise during the audio signal acquisition, which could lead to low accuracy and frequent false alarms.

Thirdly, vision-based fall detection typically uses image processing techniques to construct a human body model to detect fall. In general, video-based fall detection systems have shown some potential and reliability in detecting falls in public places. Due to the popularity of video surveillance, vision-based fall detection methods have already become one research hotspot. The boundary extraction method is used to obtain the aspect ratio of the human body and then to judge falls. Sase and Bhandari [17] used contour-based template matching to distinguish human and nonhuman and then judged human fall according to the distance between the top and center of the external rectangle of the human body. Shen et al. [18] propose a fall detection method using the Deeper Cut model to extract human key points, and it is implemented using Raspberry Pi platform. Vision-based fall detection can use relatively cheap cameras to quantify and judge various activities; nevertheless, it requires complex handling methods to construct a human body model and it is unsuitable for real-time detection mode.

In addition, some results have suggested that a single detection model from individual DataSet could easily lead to false detection. Recently, there are already a few methods on the basis of wearable sensors and surveillance cameras to classify body activities. Kepski and Kwolek [19] apply a

Kinect camera and a device consisting of an accelerometer and a gyroscope, and then a fuzzy inference system is used to separate fall from daily activities. Hondori et al. [20] present a detection system that helps monitor various dining activities of poststroke patients using a Kinect camera and an accelerometer. Nizam et al. [21] propose a novel approach that uses a depth sensor and employs a unique procedure that identifies the fall risk levels to adapt the algorithm for different people with their physical strength to withstand falls. Bogdan et al. [22] present a low-cost system for reliable fall detection with a very low false alarm ratio on the basis of accelerometric data and depth maps. The single drawback of the above methods is that Kinect camera is not cheap since a considerable computational power is needed to execute image processing algorithms. He et al. [23] propose a method to integrate the information of video images, sound, infrared, pulse, and other information into the elderly care system. However, it is not very realistic to detect fall accidents with so many sensors.

To overcome these shortcomings, we use an ordinary camera and accelerometer as a data source in this paper so as to improve the practicability of the detection system. Furthermore, a novel data fusion based fall detection online system and one GBDT based detection algorithm is provided in detail. Data fusion of human activity features obtained by posture sensor and surveillance cameras plays a significant role in the recognition of abnormal activities. In addition, the proposed platform uses NB-IoT communication and Ethernet to transmit manifold data to the Cloud Server for further analysis. The platform is able to effectively monitor the daily life of the elderly. When an unexpected fall incident occurs, the proposed system will send an alarm signal to inform the family relatives or other related guardians. In conclusion, the proposed system would meet the requirements of high sensitivity and precision. As a result, necessary assistance could be provided in times with high coverage communication technology, so it is suitable for application in the elderly care system.

The rest of the paper is organized as follows. Section 2 describes the whole online fall detection platform framework and comprehensive data source. In Section 3, GBDT based fall detection algorithm using comprehensive data from an accelerometer based posture sensor and human skeleton extraction is presented in detail. Comparable experimental results and actual operating interface are described in Section 4. Finally, a conclusion is drawn in Section 5.

## 2. System Framework and Comprehensive Data Source

Our complete framework of the fall detection system for seniors is illustrated in Figure 1. Each user is equipped with a kind of self-made MEMS (Micro Electro Mechanical Systems) based wearable sensor with hardware block diagram in Figure 2, which uses triaxial acceleration and angular velocity sensor to capture the body posture. Besides, NB-IoT (Narrow Band Internet of Things) [24] communication mode is used to transmit sensory data to the Cloud Server. With the development of Internet of Things technology, the

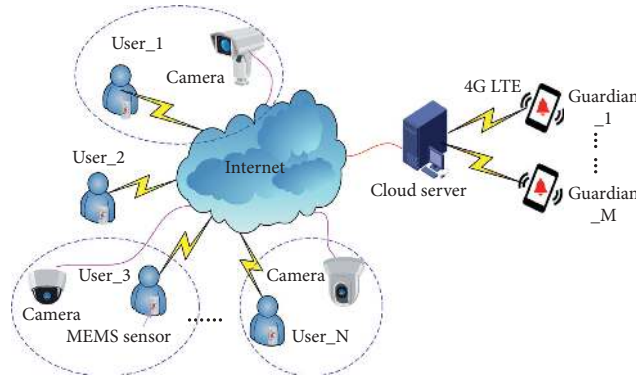


FIGURE 1: Framework of the proposed platform.

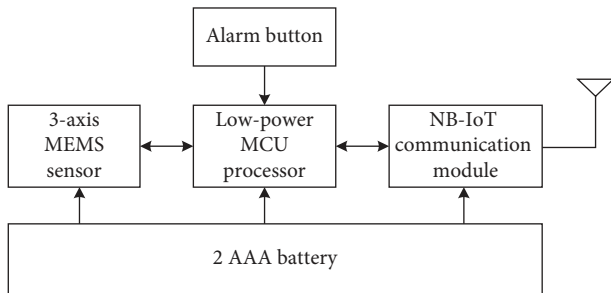


FIGURE 2: The block diagram of self-made posture sensor.

health care field is also affected deeply. As we know, NB-IoT is an emerging technology with many good features such as wide coverage, multiple connections, low speed, low power consumption, etc. In our IoT based health monitoring system, various detection devices are connected together for data exchange, so as to deliver warnings to medical staff or guardians in time when the elderly fall. Moreover, it is also supposed that each user is covered by at least one surveillance camera and so we can monitor and record each user's activity. Today, almost all surveillance cameras have the ability to transmit video sequences to the Internet through Ethernet or wireless networks. Therefore, the Cloud Server could obtain both attitude data and video data and store it at the local database for further analysis. The fall detection algorithm is running on the Cloud Server with high performance. Once fall events are detected, the Cloud Server will send an alarm signal to the specific guardian through 4G LTE (Long Term Evolution) communication technology. As a result, each user can get instant help and timely treatment in case of any abnormality with our proposed framework.

The principle of the fall detection process is demonstrated in Figure 3. In this system framework, real-time acceleration data from posture sensors are transmitted by NB-IoT communication mode to the Cloud Server, and then form a data collection named as Acceleration DataSet (ADS). Surveillance cameras are used to collect human activities' video, and key point coordinates of human body are obtained through OpenPose software [25] processing to form Video DataSet (VDS). Video DataSet and Acceleration DataSet together make up the so-called Merged DataSet (MDS). After processing by sliding window strategy [26],

one ensemble learning method named Gradient Boost Decision Tree (GBDT) [27] is applied for self-learning with MDS, so as to classify fall and other normal activities in a robust way.

Since human activity frequency generally does not exceed 20 Hz [28], the acceleration acquisition frequency is set to 30 Hz so as to process data more accurately, and video acquisition frequency is adjusted to 30 fps (frames per second) after software processing. The entire self-made DataSet comprises six kinds of human activities involving fall, walk, sit, squat, lie down, and jump.

Due to the diversity and complexity of fall accidents, it is hard to identify the way and direction of fall events. To overcome this shortcoming, we use the sum vector of triaxial acceleration value to measure the human activity. Let  $a_x$ ,  $a_y$ ,  $a_z$  denote acceleration value in three dimensions, respectively;  $A_{3\text{-axis}}$  represent the actual value of triaxial acceleration, which can be calculated with the following equation:

$$A_{3\text{-axis}} = \sqrt{a_x^2 + a_y^2 + a_z^2}. \quad (1)$$

The measured sensory data are illustrated in Figure 4 to compare the acceleration variation curve of falls and that of other normal activities, including squat, lie down, jump, walk, and sit down.

The other fall characteristics are key skeleton coordinates of the human body as shown in Figure 5. The rectangular coordinate system is established in Figure 5, and the horizontal and vertical coordinates of  $N = 18$  key points would be obtained, respectively.  $(X_{sk-i}, Y_{sk-i})$  ( $i = 1, 2, \dots, N$ ) denotes the respective coordinates of each key point. Hence, the collection of key skeleton coordinates is as follows:

$$(X_{sk}, Y_{sk}) = \{(X_{sk-1}, Y_{sk-1}), (X_{sk-2}, Y_{sk-2}), \dots, (X_{sk-18}, Y_{sk-18})\}. \quad (2)$$

However, the total 18 key points extracted using OpenPose software are not all effective and necessary for identifying fall events. Redundant data will only increase computation complexity and even introduce unnecessary noises. Therefore, the maximum and minimum  $x$ -coordinate and  $y$ -coordinate of each key skeleton point  $X_{sk\text{-max}}$ ,  $X_{sk\text{-min}}$ ,  $Y_{sk\text{-max}}$  and  $Y_{sk\text{-min}}$  are picked out, respectively, and

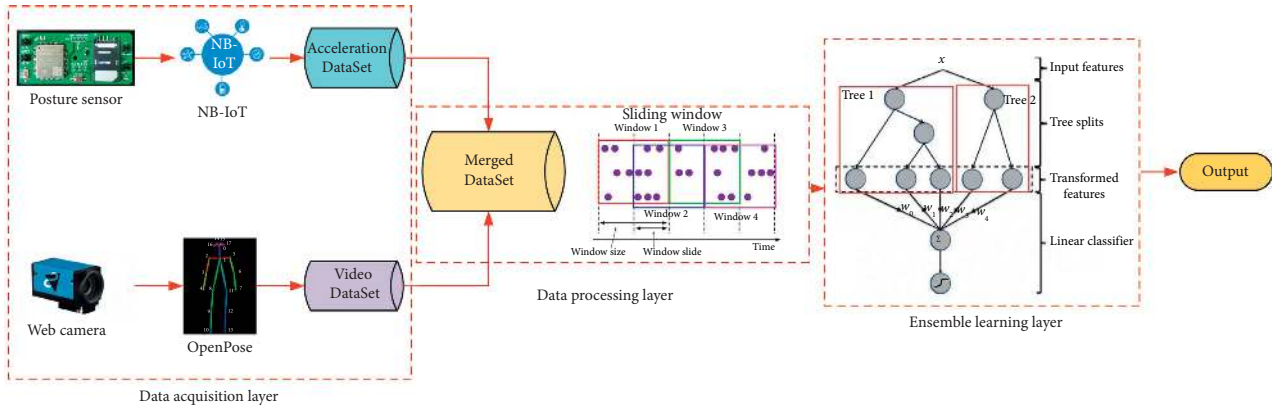


FIGURE 3: The principle of GBDT-FD algorithm.

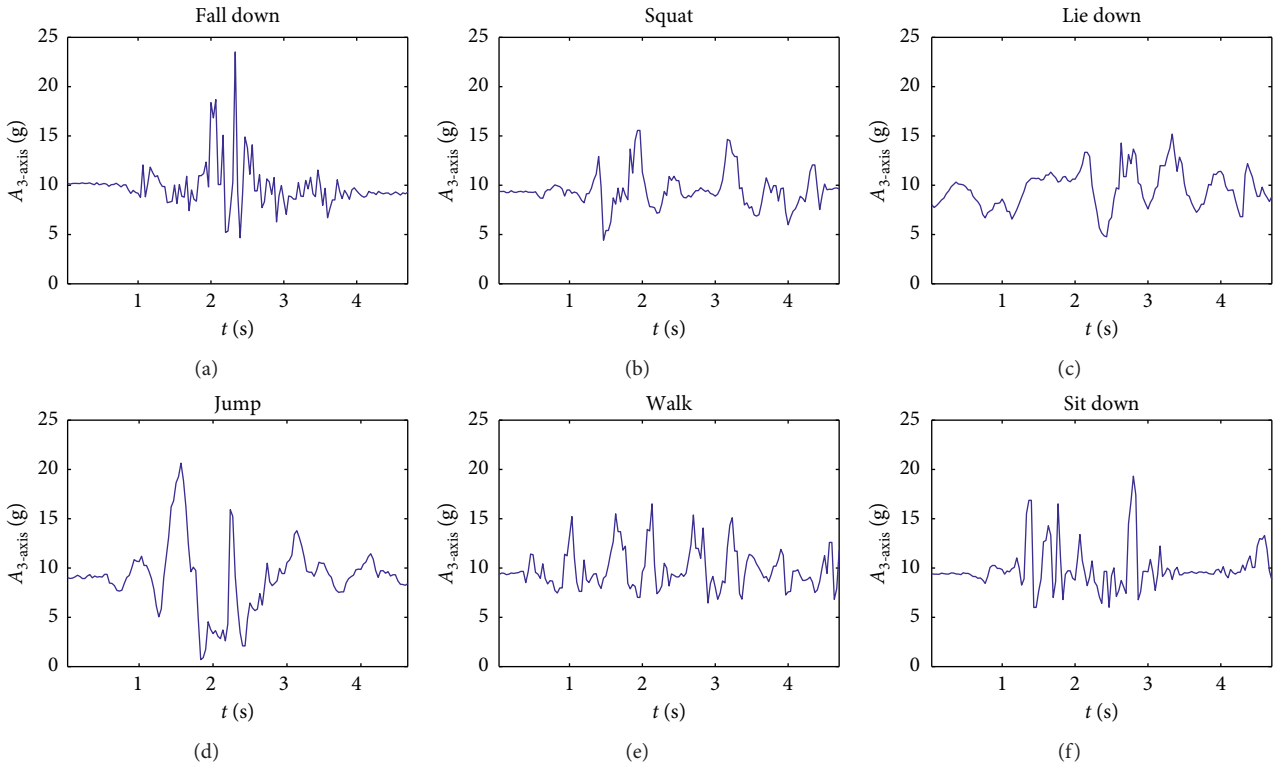


FIGURE 4:  $A_{3-axis}$  variation with different human activities.

then a body circumscribed rectangle is introduced to represent the contour of the human body. The behavior change of the human body can be identified only by paying attention to the changes of length and width of body rectangle, so as to reduce calculation complexity significantly compared to that using all the key skeleton points.

Figure 6 illustrates three different scenarios of standing, losing balance, and falling down completely. The colored rectangle denotes the outline of the human body, and it is very clear that the length and width of the body rectangle have changed a lot. We regard the aspect ratio  $R$  between body width and length as a feature from Video DataSet, which is calculated as follows:

$$R = \frac{X_{sk-max} - X_{sk-min}}{Y_{sk-max} - Y_{sk-min}} \quad (3)$$

Due to complex surroundings in different monitoring scenarios, sometimes there is no guarantee that OpenPose software can obtain 18 key points completely. In this case, the median of the previous occurrence of this point and the next occurrence of this point will be used instead. Figure 7 compares the aspect ratio  $R$  between falls and other normal activities.

Since the fall process is a continuous dynamic event in the time dimension, it could not be identified by acceleration data or video data in a moment. In order to search and detect

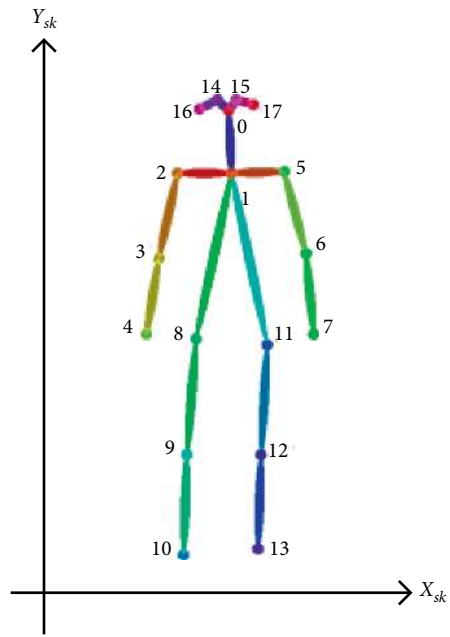


FIGURE 5: Skeleton scattergram of human body.

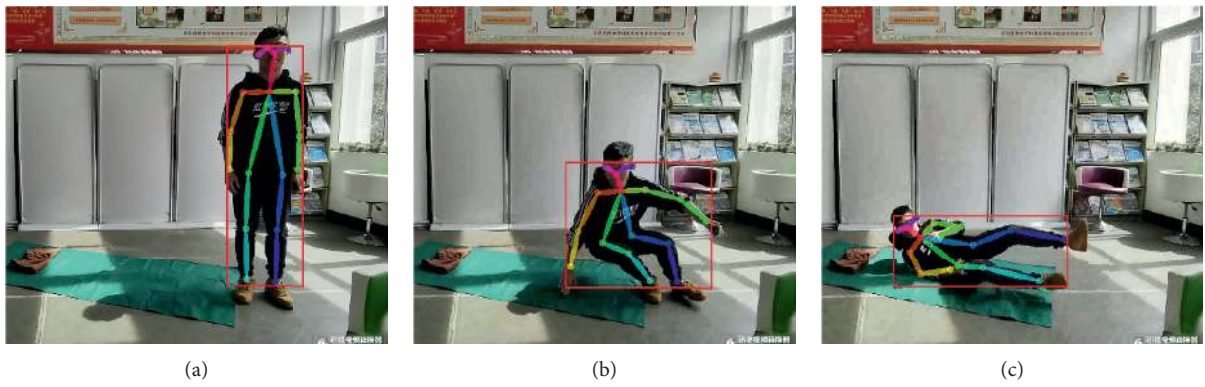


FIGURE 6: Three different scenarios during a fall process. (a) Standing. (b) Losing balance. (c) Falling down.

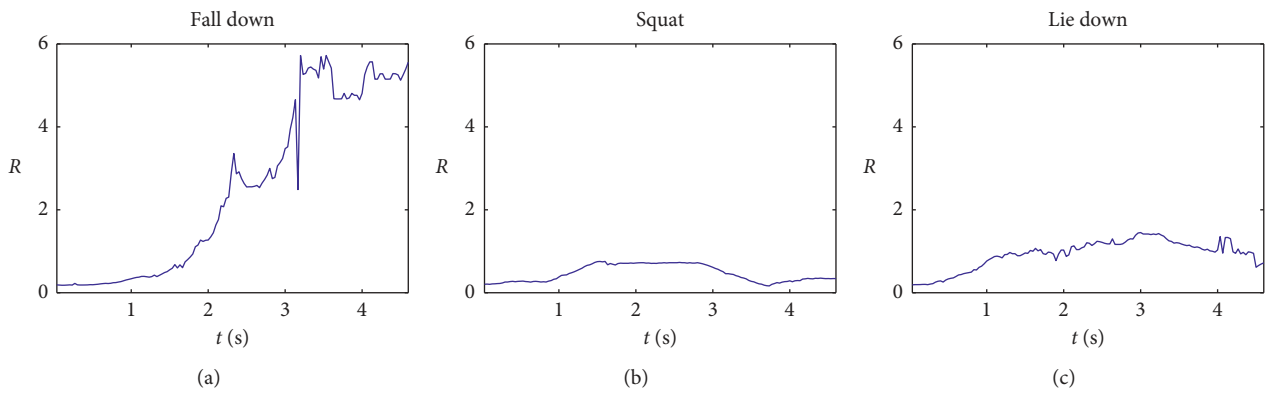


FIGURE 7: Continued.



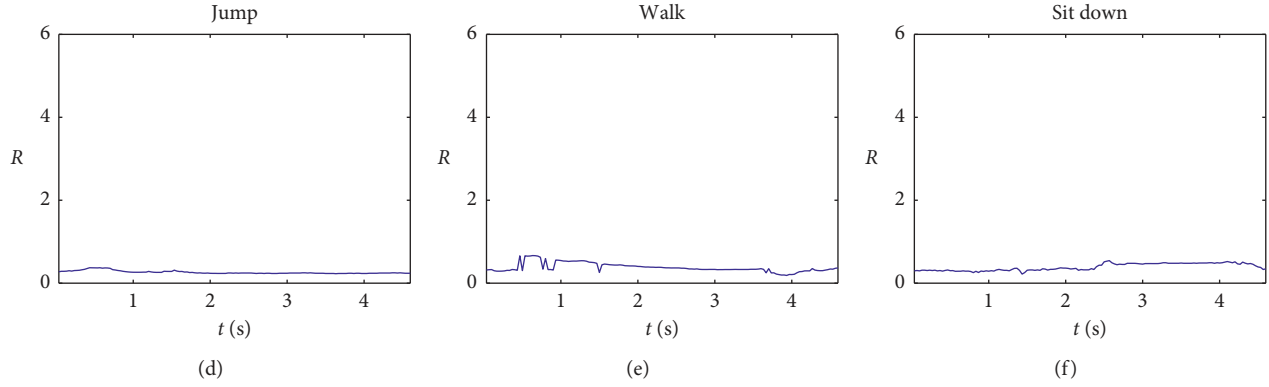


FIGURE 7: Aspect ratio  $R$  variation with different human activities.

the complete process of the falling event, we use a sliding window method to intercept the derived data flow as Figure 8, where  $A$  and  $R$  denote the sum vector of triaxial acceleration value and the aspect ratio of the human body, respectively. Following the study in Ref. [29], the complete fall process ranges from 0.3 s to 0.4 s, so we set width value and step value to 0.5 s and 0.1 s, respectively. In our GBDT-FD algorithm, the data acquisition frequency is 30 Hz, so there are  $W = 15$  pieces of data in a 0.5 s sliding window, and the step value between each two data sequences is  $S = 3$  pieces of data.

After obtaining  $A_{3\text{-axis}}$  and  $R$ , it is necessary to extract the statistical characteristic so as to carry out the classification process. When fall incident occurs, the acceleration and posture of the human body will exceed normal range and

change rapidly. In conclusion, we should only pay attention to the overall size and change rate of  $A_{3\text{-axis}}$  and  $R$  within each sliding window. For both Acceleration DataSet and Video DataSet, 7 characteristic values including mean, standard deviation (std), maximum value (max), minimum value (min), average change rate ( $d$ ), number of mean crossings (MCV), and root mean square (RMS) of each sliding window are calculated, respectively. MCV means the number of samples above the mean value in a set of data, so too large or too small MCV means this dataset change too dramatically. As a result, the above 14-dimensional statistical characteristics are listed in Table 1.

We extracted 14-dimensional statistical characteristics from each sliding window to constitute the Merged DataSet, which is stored as in the following matrix:

$$\begin{bmatrix} A_{\text{mean}1}, A_{\text{std}1}, A_{\text{max}1}, A_{\text{min}1}, A_{d1}, A_{\text{MCV}1}, A_{\text{RMS}1}, R_{\text{mean}1}, R_{\text{std}1}, R_{\text{max}1}, R_{\text{min}1}, R_{d1}, R_{\text{MCV}1}, R_{\text{RMS}1} \\ A_{\text{mean}2}, A_{\text{std}2}, A_{\text{max}2}, A_{\text{min}2}, A_{d2}, A_{\text{MCV}2}, A_{\text{RMS}2}, R_{\text{mean}2}, R_{\text{std}2}, R_{\text{max}2}, R_{\text{min}2}, R_{d2}, R_{\text{MCV}2}, R_{\text{RMS}2} \\ A_{\text{mean}3}, A_{\text{std}3}, A_{\text{max}3}, A_{\text{min}3}, A_{d3}, A_{\text{MCV}3}, A_{\text{RMS}3}, R_{\text{mean}3}, R_{\text{std}3}, R_{\text{max}3}, R_{\text{min}3}, R_{d3}, R_{\text{MCV}3}, R_{\text{RMS}3} \\ \dots \\ A_{\text{mean}i}, A_{\text{std}i}, A_{\text{max}i}, A_{\text{min}i}, A_{di}, A_{\text{MCV}i}, A_{\text{RMS}i}, R_{\text{mean}i}, R_{\text{std}i}, R_{\text{max}i}, R_{\text{min}i}, R_{di}, R_{\text{MCV}i}, R_{\text{RMS}i} \end{bmatrix}, \quad (4)$$

where subscript  $i$  denotes that this characteristic sequence is extracted from the  $i$ -th sliding window.

### 3. GBDT-Based Fall Detection Algorithm

**3.1. Ensemble Learning-Based Fall Detection.** Ensemble learning [30] is a machine learning method that combines multiple weak learners into a strong one. Several individual learners with complementary results are trained concurrently, and the results from each individual learner are merged into the final prediction result by a certain combination strategy, so as to achieve the effect of extensive learning and effective classification. If each individual learner is the same type in ensemble learning, these individual learners are called base learners. The advantage of ensemble learning lies in its strong robustness, which

enables it to reduce the interference of noisy data effectively. Moreover, normalization is not required in the data preprocessing stage. Gradient boosting is a machine learning technique for regression problems, which produces a prediction model in the form of an ensemble of weak prediction models. GBDT (Gradient Boosting Decision Tree) is a boosting method using Classification and Regression Tree (CART) [31] as its base learner which is illustrated in Figure 9. GBDT uses decision tree as the weak prediction model in gradient boosting so it has high accuracy.

The described residual is the difference between the correct result and the actual result. In GBDT-FD algorithm, we use the logarithmic loss function to define residuals as follows:

$$L(y, f(X)) = \log(1 + \exp(-yf(X))). \quad (5)$$

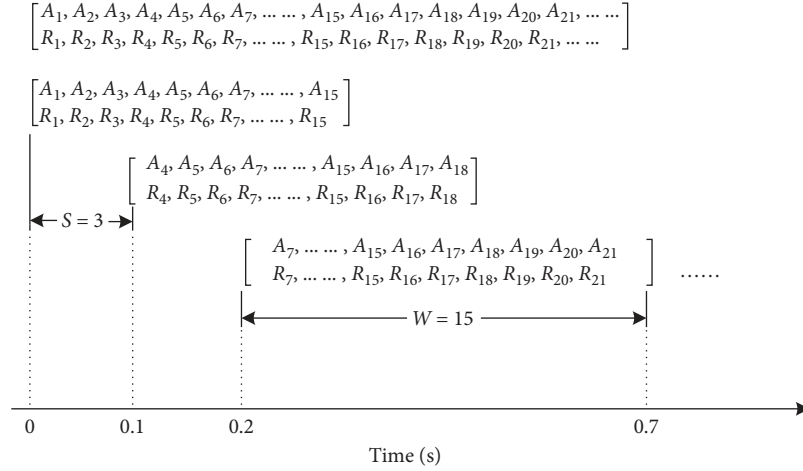


FIGURE 8: Sliding window.

TABLE 1: Statistical characteristics.

Acceleration data		Skeleton data	
Features	Symbol	Features	Symbol
Mean value	$A_{\text{mean}}$	Mean value	$R_{\text{mean}}$
Standard deviation value	$A_{\text{std}}$	Standard deviation value	$R_{\text{std}}$
Maximum value	$A_{\text{max}}$	Maximum value	$R_{\text{max}}$
Minimum value	$A_{\text{min}}$	Minimum value	$R_{\text{min}}$
Average change value	$A_d$	Average change value	$R_d$
Mean crossings value	$A_{\text{MCV}}$	Mean crossings value	$R_{\text{MCV}}$
Root mean square value	$A_{\text{RMS}}$	Root mean square value	$R_{\text{RMS}}$

$$X = (A_{\text{mean}}, A_{\text{std}}, A_{\text{max}}, A_{\text{min}}, A_d, A_N, A_{\text{RMS}}, R_{\text{mean}}, R_{\text{std}}, R_{\text{max}}, R_{\text{min}}, R_d, R_N, R_{\text{RMS}}). \quad (7)$$

The optimal parameters of the GBDT model-based fall detection algorithm are determined by the evaluation method with  $k$ -fold cross-validation, which means that the whole DataSet is divided into  $k$  mutually exclusive subsets with approximately equal size. One subset is taken as the test set and the rest as the training set while training GBDT model. In this way, the model can be trained and tested for  $k$  times to evaluate the performance of the model more objectively.

GBDT classifier is regarded as an additional model composed of CART [31] illustrated in equation (8). Classification is realized by continuously decreasing the residuals generated in the training process as follows:

$$f(X) = \sum_{n=1}^N h(X; \theta_n), \quad (8)$$

where  $f(X)$  denotes the overall model,  $h(X; \theta_n)$  denotes the  $n$ -th decision tree,  $\theta_n$  denote parameters of the  $n$ -th decision tree,  $N$  denotes the number of decision trees.

The purpose of GBDT is to make loss function reduce as fast as possible and preferably fall along its gradient direction. At each round, the negative gradient of the log-likelihood loss function is used to fit the new CART, which could accelerate the reduction and convergence of loss function as soon as possible and finally speed up the training process. GBDT-FD will train CART for fall detection through several iterations.

The following describes the main steps of the GBDT-FD algorithm.

**3.1.1. Initialization.** Firstly, GBDT-FD algorithm chooses one feature from  $X$  as the CART node and then selects an appropriate eigenvalue as the segmentation point. For example,  $A_{\text{mean}}$  is taken as the root node. In general,  $A_{\text{mean}}$

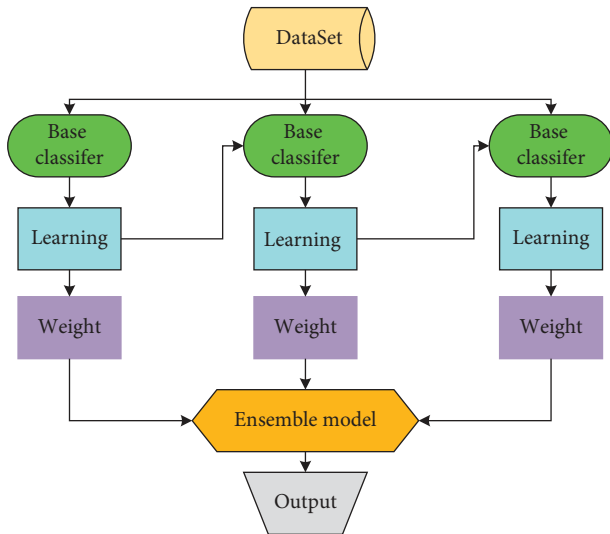


FIGURE 9: Ensemble learning.

The 14-dimensional Merged DataSet MDS defined in Section 2 is as follows:

$$\text{MDS} = \{(X_1, y_1), (X_2, y_2), \dots, (X_m, y_m)\}, \quad (6)$$

where  $X$  denotes 14-dimensional data, which is shown as follows:  $y$  denotes labels,  $y = 1$  means fall, and  $y = -1$  means normal activity:

value of a fall is usually much larger than that of normal activity. Finally, the Merged DataSet is divided into two categories with the eigenvalue boundary of preset segmentation. It can be roughly considered that these two categories represent falls and normal activities, respectively, but the loss function will be very large. GBDT-FD will iterate through all combinations of features and segmentation points to minimize the loss function of CART. Therefore, initial CART has only one root node as in the following equation and  $M$  denotes the number of samples:

$$f_0(X) = \arg \min \sum_{m=1}^M L(y_m, c). \quad (9)$$

**3.1.2. Iterations.** Suppose the number of iterations is  $T$ , and  $e_{tm}$  denotes the negative gradient error of  $m$ -th sample in the  $t$ -round iteration; hence,

$$e_{tm} = - \left[ \frac{\partial L(y_m, f(X_m))}{\partial f(x_m)} \right]_{f(X)=f_{t-1}(X)} = \frac{y_m}{1 + \exp(y_m f(x_m))}. \quad (10)$$

GBDT-FD will train the CART of this iteration according to  $(X_m, e_{tm})(i = 1, 2, \dots, M)$ . Leaf node region  $Re_{tj}(j = 1, 2, \dots, J)$  will be obtained by  $J$  denoting the number of regression leaf nodes. Using linear search, estimate the value of the leaf node region and minimize the loss function:

$$c_{tj} = \arg \min \sum_{X_m \in Re_{tj}} \log(1 + \exp(-y_m (f_{t-1}(X_m) + c))). \quad (11)$$

Since equation (11) is difficult to optimize, we use approximation as follows:

$$c_{tj} = \frac{\sum_{X_m \in Re_{tj}} e_{tm}}{\sum_{X_m \in Re_{tj}} |e_{tm}| (1 - |e_{tm}|)}. \quad (12)$$

Then a new learner generated by this iteration process can be expressed as follows:

$$f_t(X) = f_{t-1}(X) + \sum_{j=1}^J c_{tj} I, \quad X_m \in Re_{tj}. \quad (13)$$

However, in order to avoid overfitting, learning rate  $\nu$  ranging in  $(0, 1]$  should be considered when updating learners, so the evolution rule of a new learner is as follows:

$$f_t(X) = f_{t-1}(X) + \nu \cdot \sum_{j=1}^J c_{tj} I, \quad X_m \in Re_{tj}. \quad (14)$$

In general, much more iterations mean smaller loss function and better performance. However, the loss function will decrease less obviously as the number of iterations increases. In order to save training time and computational resources, GBDT-FD sets training termination conditions:

The training process will stop if,

$$\Delta L(y, f(X)) < \varepsilon, \quad (15)$$

where  $\Delta L(y, f(X))$  denotes the decrease of the loss function, and  $\varepsilon$  is the training termination threshold set by the GBDT-FD algorithm.

**3.1.3. Classification.** GBDT-FD classification process is as follows:

- (1) Input unclassified test dataset
- (2) The trained CART will start from the root node and compare with the input data based on features and segmentation points
- (3) After the comparison process is completed, CART will output the category of each input data
- (4) Test datasets are classified into two categories: fall and nonfall

**3.2. Performance Metrics.** As we know, for the dichotomous problem, the final classifying results can be divided into four categories: True Positive samples (TP), False Negative samples (FN), False Positive samples (FP), and True Negative samples (TN) on the basis of different combinations of real category and predicted category, which are summarized in Table 2.

Since fall detection belongs to a typical dichotomous problem (fall and nonfall), five integrated metrics are used to verify performance usually [2].

- (1) Precision: the ratio between real falling samples and predicted falling samples:

$$I_{\text{pre}} = \frac{TP}{TP + FP}. \quad (16)$$

- (2) Sensitivity: the ratio between samples correctly identified as falls and real falling samples:

$$I_{\text{sens}} = \frac{TP}{TP + FN}. \quad (17)$$

- (3) Specificity: the ratio between samples correctly identified as nonfalls and real nonfalling samples:

$$I_{\text{spec}} = \frac{TN}{TN + FP}. \quad (18)$$

- (4) Accuracy: the ratio between samples correctly identified and the total sample set:

$$I_{\text{acc}} = \frac{TP + TN}{TP + FN + TN + FP}. \quad (19)$$

- (5)  $F$ -score: weighted harmonic mean of precision and sensitivity:

$$I_{\beta\text{-score}} = \frac{(\beta^2 + 1) \cdot I_{\text{pre}} \cdot I_{\text{sens}}}{\beta^2 \cdot I_{\text{pre}} + I_{\text{sens}}}, \quad (20)$$



TABLE 2: Different combinations of real category and predicted category.

Real category	Predicted category	
	Fall	Non-fall
Fall	TP	FN
Non-fall	FP	TN

where  $\beta$  is a nonnegative parameter, and different  $\beta$  indicates different weights between precision and sensitivity. Apparently, there are three main cases:

- (1)  $\beta = 1$ , denoting the weight of  $I_{pre}$  and  $I_{sens}$  is equal
- (2)  $\beta > 1$ , denoting the weight of  $I_{sens}$  is greater than that of  $I_{pre}$
- (3)  $\beta < 1$ , denoting the weight of  $I_{pre}$  is greater than that of  $I_{sens}$

According to the difference of  $\beta$ ,  $I_1$ -score,  $I_{0.5}$ -score, and  $I_2$ -score are commonly used in statistical occasions. In terms of fall detection, we should pay attention to the sensitivity of the detection model so as to reduce health hazards effectively, and moreover, ensure the wounded could get immediate medical assistance when fall occurs. Meanwhile, wrong fall alarm may lead to a waste of communication resource, so it is necessary to pay attention to the precision of the model, but it is not as important as the sensitivity metric.

## 4. Experimental Results

**4.1. Experimental Setup.** The experimental scene at our school is illustrated in Figure 10. Ten graduate students (6 male and 4 female) with average age 26 are volunteered for this research testing. Each user's real-time acceleration data during daily activities is gathered by our wearable posture sensor, and monitoring video is recorded using a surveillance camera. In order to imitate the real behavior of the aged, we also tie sandbags on our feet, so there is amount of falling down slowly situations in our training data set. With this scenario, we got more than 150 videos with each lasting for 1–1.5 minutes. Finally, more than 6000 activities records are used to verify our GBDT based algorithm, 20% of them are used for training, and the rest 80% is used for testing. For safety's sake, the experimental environment is padded completely and tight. The acceleration data and video data are combined to form the merged data to store in Cloud Server and finally input to GBDT based fall down classifier.

Firstly, how to determine optimal parameters for GBDT based fall detection algorithm is discussed. We treat  $I_2$ -score and accuracy as main evaluation criteria and  $I_1$ -score and  $I_{0.5}$ -score as the reference when training GBDT based detection dichotomy model.

In the whole GBDT model, the iteration times have an important influence on the quality of the model. When the iteration times increase from 20 to 300, the  $I_2$ -score and accuracy value of the GBDT model will increase. However, it is clear from Figure 11 when the iteration times exceed 110,  $I_2$ -score and accuracy of GBDT model no longer increase



FIGURE 10: Experimental scene.

significantly. But we can find from Figure 11 that the time required for the training model is still approximately linear, so the optimal value of iteration times is set to 110 in this paper.

Some main parameters of GBDT-FD are shown in Table 3.

**4.2. Experimental Results.** The original sensory data of body activities in a typical fall case are illustrated in Table 4 and Figure 12. Table 4 displays a complete fall process with 6 key frames. Figure 12 demonstrates the detailed change law of the  $A_{3-axis}$  during this process. It is clear that both types of acceleration sensory data and human posture perception data can reflect the fall process on many occasions.

Unfortunately, the previous situation is not always guaranteed. In some cases, only a single type of sensory data is difficult to classify falls and normal behaviors accurately. When a user lies down, the aspect ratio  $R$  will change a lot as given in Figure 13(a), so it will be recognized as fall activity using separated video data. But it is a misjudgment clearly and we can verify this from  $A_{3-axis}$  variation curve. Nevertheless, the fall classification result of this case should be right using the accelerometric data, which is displayed in Figure 13(b).

In either case, when a user jumps up, it is possible to identify this kind of activity as fall because the acceleration value changes extremely violently. But it is not a fall through video data from Figure 14 because there is no great change of aspect ratio  $R$ . To summarize, these two special examples verify that more accurate fall detection results could be achieved with data fusion using the Merged DataSet, and not from a single dataset. Hence, this discovery is also the main purpose and significance of this article.

Secondly, we compare the detection accuracy influence of each feature. GBDT can output the relative importance of each feature to model training so as to help understand the influence of each feature on fall down detection. As shown in Figure 15, the sum of importance ratio of features in Video DataSet (VDS) is 50.4%, and that in Acceleration DataSet (ADS) is 49.6%. This indicates that both datasets play an important role in classification, so we choose all these characteristics as our source data in GBDT-FD.

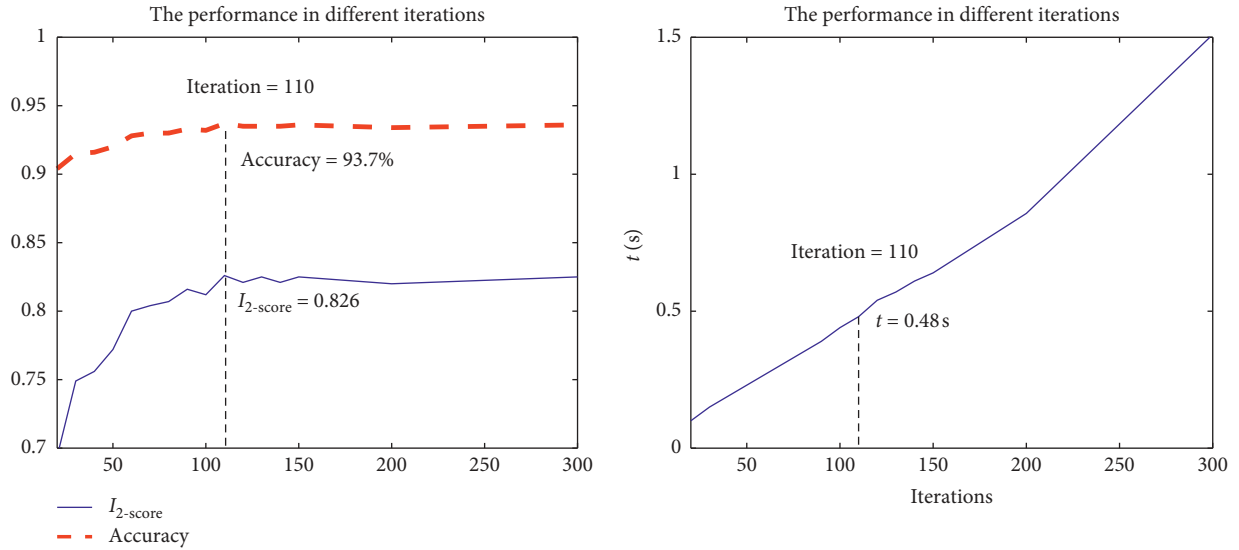


FIGURE 11: The influence of iteration times of GBDT-FD.

TABLE 3: GBDT-FD parameters.

Parameter	Symbol	Optimal value
Learning rate	$\nu$	0.35
Iteration rounds	$T$	110
Cross validation	$k$	5
Total sample number	$M$	5160
Terminal condition	$\epsilon$	$10^{-4}$

TABLE 4: Fall process.






Frame no.	Scene
1	
20	
30	
49	

TABLE 4: Continued.

Frame no.	Scene
65	
85	

Furthermore, we compare the fall judgment results of support vector machine (SVM) based Fall Detection (SVM-FD) [32], Naive Bayes (NB) based Fall Detection (NB-FD) [33], Decision Tree (DT) based Fall Detection (DT-FD) [34], K-Nearest Neighbor (KNN) based Fall Detection (KNN-FD) [35], Neural Network (NN) based Fall Detection (NN-FD) [36], Random Forest (RF) based Fall Detection (RF-FD) [37], and GBDT based Fall Detection (GDBT-FD) with the Merged DataSet, acceleration DataSet, and skeleton DataSet, respectively. The results in Tables 5–7 explain the

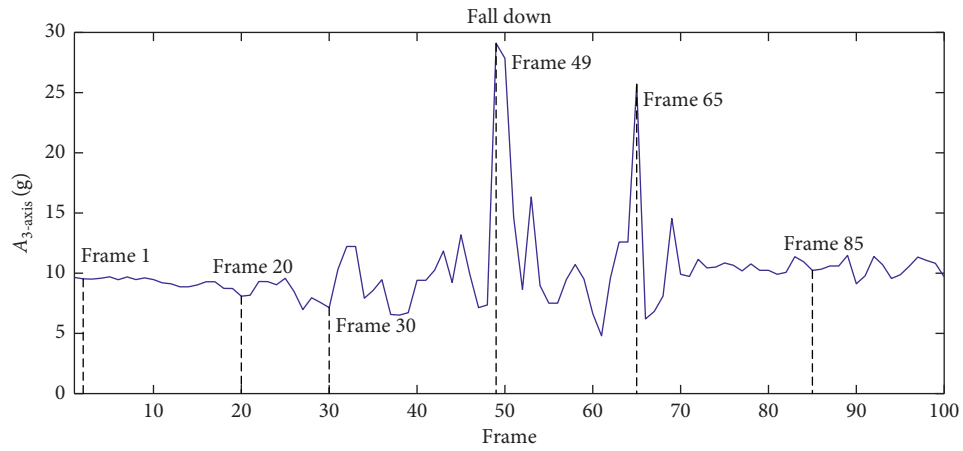
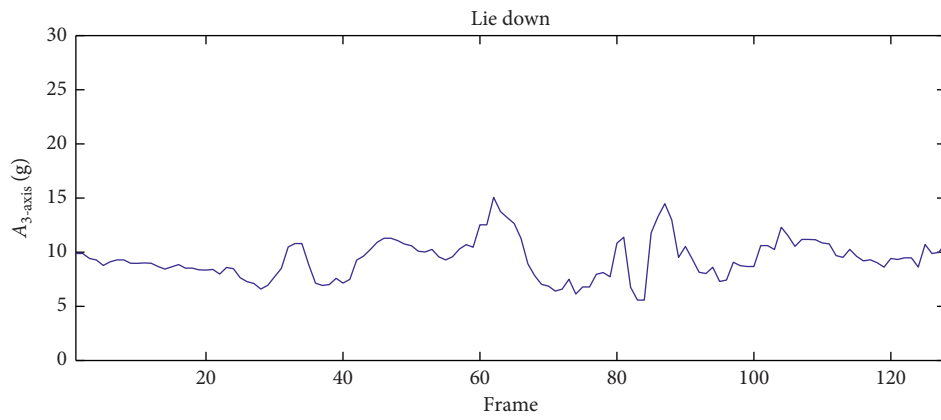


FIGURE 12: A<sub>3</sub>-axis variation when falling down.



(a)



(b)

FIGURE 13: Sensory data when lying down. (a) Video skeleton data when lying down. (b) A<sub>3</sub>-axis variation when lying down.



(a)

FIGURE 14: Continued.

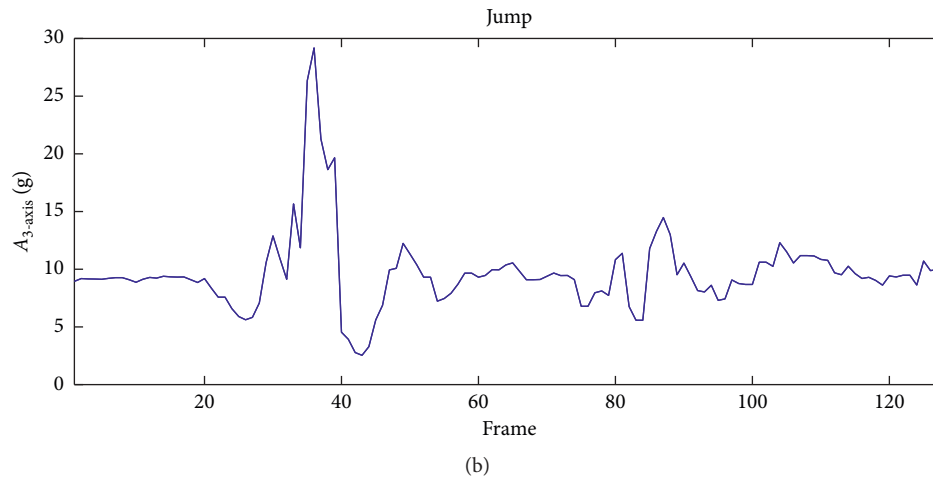


FIGURE 14: Sensory data when jumping up. (a) Video skeleton data when jumping up. (b) A3-axis variation when jumping up.

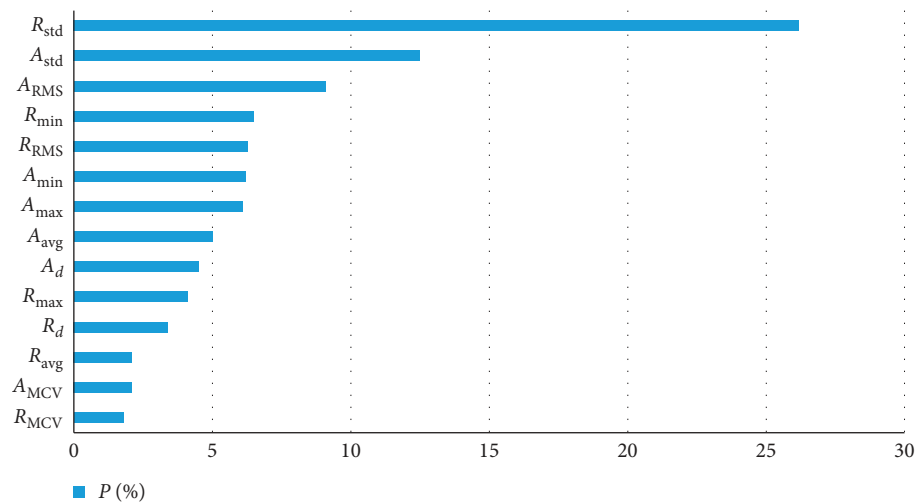


FIGURE 15: Feature importance in GBDT-FD algorithm.

comparable detection results of these algorithms with three kinds of DataSet, respectively.

It is clear from Table 5 that  $I_2$ -score and accuracy of GBDT-FD are, respectively, 0.878 and 95%. The results of GBDT based fall detection algorithm outperform that of SVM-FD, NB-FD, DT-FD, KNN-FD, NN-FD, and RF-FD, which indicates that GBDT-FD can identify fall events accurately. NN-FD and RF-FD are slightly worse than GBDT-FD. Due to the simplicity of the Naive Bayes model, the performance of the NB-FD algorithm is not ideal too. The Accuracy of SVM-FD and KNN-FD is not much worse than that of GBDT-FD, but there is a big gap between them in  $I_2$ -score.

From detection results in Tables 6 and 7 with independent DataSet, it is clear that GBDT-FD has good generalization ability and can handle various types of DataSet. Moreover, the recognition results with Video DataSet are slightly better than that of Acceleration DataSet. This is because the characteristics of video skeleton data are more directly perceived than that of acceleration. However, the

performance is still not good as that with Merged DataSet since the Merged DataSet expands the data dimensions, and more features are used to make the model much easier to train. The results verify that using a posture sensor and video skeleton fusion will be more accurate than the traditional individual detection method. However, the other methods have poor generalization ability due to the defects of algorithms. Therefore, the detection results are not satisfactory.

Finally, the most important index  $I_2$ -score of each algorithm using three different DataSet is compared in Figure 16. The performance advantages of the proposed GBDT-FD algorithm are obvious. From the aspect of system implementation, the Web interface is shown in Figure 17. There are a few users who have already connected and transmitted sensory data to the Cloud Server. GBDT-FD algorithm is executed on the Server. At present, the well-designed fall detection online platform is already in trial operation in nursing homes, and so it is with a good result in application and worth popularization.

TABLE 5: Detection results of each algorithm with merged DataSet.

Model	Merged DataSet			Accuracy (%)
	$I_{0.5}$ -score	$I_1$ -score	$I_2$ -score	
SVM-FD	0.842	0.793	0.749	91.6
NB-FD	0.630	0.579	0.535	83.5
DT-FD	0.806	0.805	0.789	90.9
KNN-FD	0.785	0.751	0.72	89.7
NN-FD	0.846	0.835	0.825	92.8
RF-FD	0.846	0.831	0.818	92.7
<b>GBDT-FD</b>	<b>0.878</b>	<b>0.886</b>	<b>0.878</b>	<b>95</b>

TABLE 6: Detection results of each algorithm with Acceleration DataSet.

Model	Acceleration DataSet			Accuracy (%)
	$I_{0.5}$ -score	$I_1$ -score	$I_2$ -score	
SVM-FD	0.707	0.600	0.516	85.8
NB-FD	0.619	0.557	0.506	83.1
DT-FD	0.659	0.613	0.602	83.8
KNN-FD	0.707	0.656	0.613	86.3
NN-FD	0.685	0.670	0.655	85.8
RF-FD	0.741	0.656	0.588	87.1
<b>GBDT-FD</b>	<b>0.777</b>	<b>0.737</b>	<b>0.696</b>	<b>89.2</b>

TABLE 7: Detection results of each algorithm with Video DataSet.

Model	Video DataSet			Accuracy (%)
	$I_{0.5}$ -score	$I_1$ -score	$I_2$ -score	
SVM-FD	0.728	0.639	0.574	86.6
NB-FD	0.608	0.587	0.567	82.6
DT-FD	0.678	0.677	0.631	84.6
KNN-FD	0.692	0.658	0.628	85.9
NN-FD	0.707	0.692	0.677	86.7
RF-FD	0.739	0.695	0.656	87.6
<b>GBDT-FD</b>	<b>0.754</b>	<b>0.728</b>	<b>0.706</b>	<b>88.6</b>

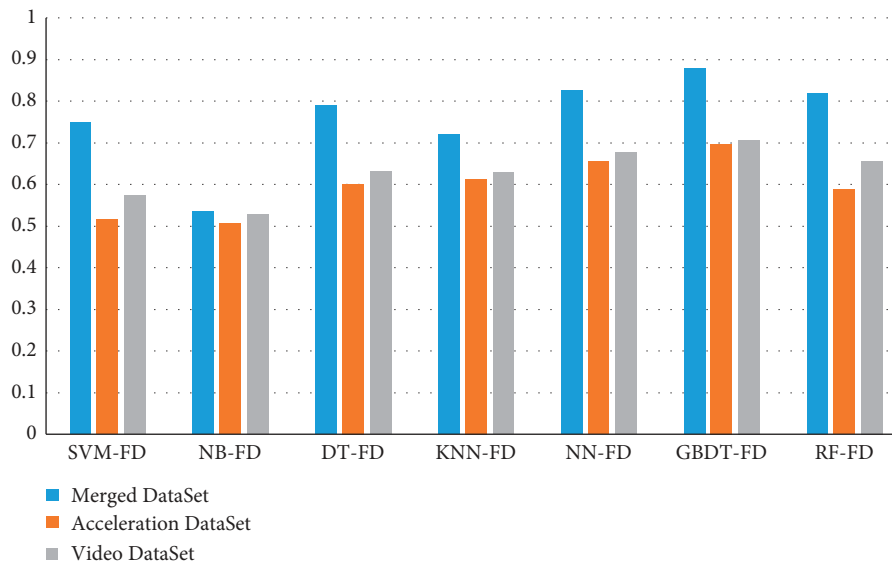


FIGURE 16:  $I_2$ -score comparison of each algorithm with different DataSets.



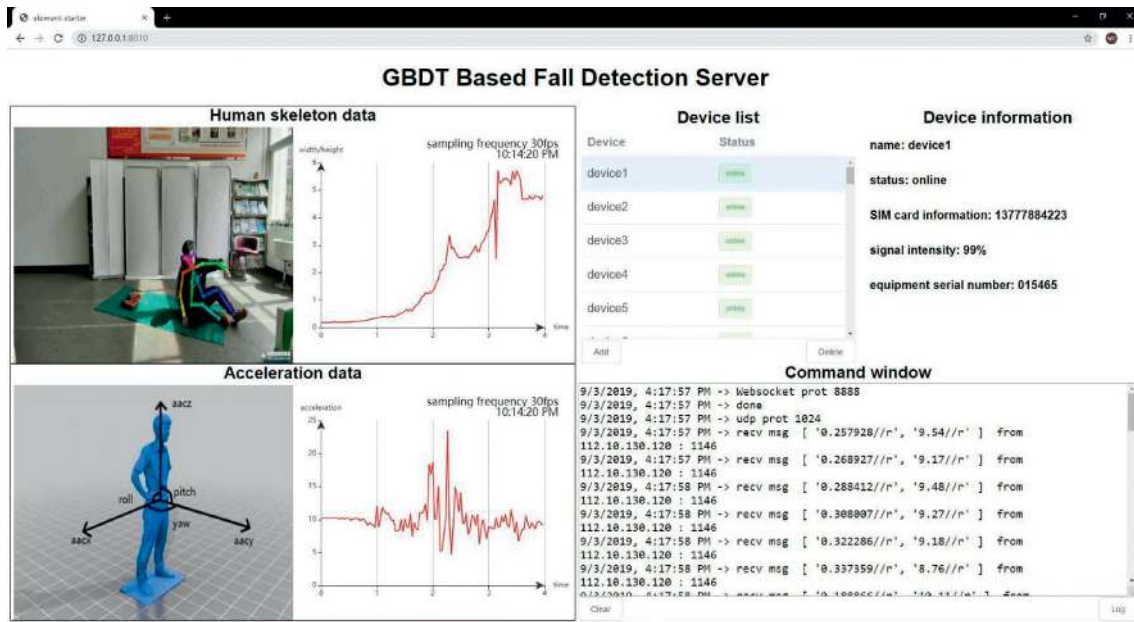


FIGURE 17: Web interface on the cloud server.

## 5. Conclusions

In this paper, we propose one kind of comprehensive framework of the fall detection system using inertial triaxial acceleration sensors and monitoring cameras to detect fall accidents. The wearable triaxial accelerometer is used to detect the body's posture, and a monitoring camera is used to extract key points of human skeletons information. The fall detection is operated on the basis of fusion based data including accelerometric data and human skeleton key points. In order to reduce false positives of falling incidents, GBDT classifier based fall detection algorithm is investigated in depth. The good performance of the proposed GBDT-FD algorithm is compared with SVM-FD, NB-FD, DT-FD, KNN-FD, NN-FD, and RF-FD in terms of  $I_{0.5\text{-score}}$ ,  $I_{1\text{-score}}$ ,  $I_{2\text{-score}}$ , and accuracy, so as to verify the performance improvements of GBDT-FD. In our future work, multiview human skeleton extraction will be adopted, and detection speed will be improved so as to enhance the real-time performance of our system.

## Data Availability

The data supporting the results of this study are available from the corresponding author upon request.

## Conflicts of Interest

The authors declare that they have no conflicts of interest.

## Acknowledgments

This research was partially supported by the National Natural Science Foundation of China (Grant Nos. 61801431 and 61871163), Fundamental Research Funds for the Provincial Universities of Zhejiang (GK199900299012),

Zhejiang Provincial Key Lab of Equipment Electronics, and National Innovation and Entrepreneurship Training Program for College Students.

## References

- [1] S. C. Mukhopadhyay and S. Chandra, "Wearable sensors for human activity monitoring: a review," *IEEE Sensors Journal*, vol. 15, no. 3, pp. 1321–1330, 2015.
- [2] L. S. Hu, S. Z. Wang, Y. Q. Chen et al., "Fall detection algorithms based on wearable device: a review," *Journal of Zhejiang University (Engineering Science)*, vol. 52, no. 9, pp. 92–103, 2018.
- [3] R. Broadley, J. Klenk, S. Thies, L. Kenney, and M. Granat, "Methods for the real-world evaluation of fall detection technology: a scoping review," *Sensors*, vol. 18, no. 7, p. 2060, 2018.
- [4] E. Boutellaa, O. Kerdjidi, and K. Ghanem, "Covariance matrix based fall detection from multiple wearable sensors," *Journal of Biomedical Informatics*, vol. 94, pp. 103–189, 2019.
- [5] J. K. Lee and E. J. Park, "3D spinal motion analysis during staircase walking using an ambulatory inertial and magnetic sensing system," *Medical & Biological Engineering & Computing*, vol. 49, no. 7, pp. 755–764, 2011.
- [6] D. Litvak, Y. Zigel, and I. Gannot, "Fall detection of elderly through floor vibrations and sound," in *Proceedings of the 2008 Annual International Conference of the IEEE Engineering in Medicine and Biology Society*, IEEE, Vancouver, Canada, pp. 4632–4635, August 2008.
- [7] 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.
- [8] A. Edgcomb and F. Vahid, "Automated fall detection on privacy-enhanced video," in *Proceedings of the 2012 Annual International Conference of the IEEE Engineering in Medicine*

- and Biology Society, IEEE, San Diego, CA, USA, pp. 252–255, August 2012.
- [9] B. Z. Dai, D. Q. Yang, L. G. Ai et al., “A novel video-surveillance-based algorithm of fall detection,” in *Proceedings of the 2018 International Congress on Image and Signal Processing, BioMedical Engineering and Informatics (CISP-BMEI)*, pp. 1–6, Beijing, China, January 2018.
  - [10] L. Liu, M. Popescu, M. Skubic et al., “Automatic fall detection based on Doppler radar motion signature,” in *Proceedings of the 5th International Conference on Pervasive Computing Technologies for Healthcare (PervasiveHealth) and Workshops*, pp. 222–225, Dublin, Ireland, 2011.
  - [11] A. Nag, S. C. Mukhopadhyay, and J. Kosel, “Wearable flexible sensors: a review,” *IEEE Sensors Journal*, vol. 17, no. 13, pp. 3949–3960, 2017.
  - [12] L. Chen, R. Li, H. Zhang, L. Tian, and N. Chen, “Intelligent fall detection method based on accelerometer data from a wrist-worn smart watch,” *Measurement*, vol. 140, pp. 215–226, 2019.
  - [13] S. A. Waheed and P. S. A. Khader, “A novel approach for smart and cost effective IoT based elderly fall detection system using Pi camera,” in *Proceedings of the 2017 IEEE International Conference on Computational Intelligence and Computing Research (ICIC)*, pp. 1–4, Beijing, China, September 2017.
  - [14] D. Yacchirema, J. S. de Puga, and C. M. Palau, “Fall detection system for elderly people using IoT and big data,” *Procedia Computer Science*, vol. 130, pp. 603–610, 2018.
  - [15] Y. Li, K. C. Ho, and M. Popescu, “A microphone array system for automatic fall detection. IEEE Transactions on Bio-Medical Engineering,” vol. 59, no. 5, pp. 1291–1301, 2012.
  - [16] S. M. Adnan, A. Irtaza, S. Aziz, M. O. Ullah, A. Javed, and M. T. Mahmood, “Fall detection through acoustic local ternary patterns,” *Applied Acoustics*, vol. 140, pp. 296–300, 2018.
  - [17] P. S. Sase and S. H. Bhandari, “Human fall detection using depth videos,” in *Proceedings of the 2018 5th International Conference on Signal Processing and Integrated Networks (SPIN)*, pp. 546–549, New Delhi, India, February 2018.
  - [18] L. X. Shen, Q. Y. Zhang, X. C. Guo et al., “Fall detection system based on deep learning and image processing in cloud environment,” in *Proceedings of the 2018 Conference on Complex, Intelligent, and Software Intensive Systems*, Matsue, Japan, July 2018.
  - [19] M. Kepski and B. Kwolek, “Fall detection on embedded platform using Kinect and wireless accelerometer,” *Computers Helping People with Special Needs*, pp. 407–414, Springer, Berlin, Germany, 2012.
  - [20] H. M. Hondori, M. Khademi, and C. V. Lopes, “Monitoring intake gestures using sensor fusion (microsoft Kinect and inertial sensors) for smart home tele-rehab setting,” in *Proceedings of the 2012 IEEE Conference on Healthcare Innovation*, Houston, TX, USA, 2012.
  - [21] Y. Nizam, M. N. H. Mohd, and M. M. A. Jamil, “Development of a user-adaptable human fall detection based on fall risk levels using depth sensor,” *Sensors*, vol. 18, no. 2260, pp. 1–14, 2018.
  - [22] B. Kwolek and M. Kepski, “Human fall detection on embedded platform using depth maps and wireless accelerometer,” *Computer Methods and Programs in Biomedicine*, vol. 117, no. 3, pp. 489–501, 2014.
  - [23] Y. He, F. Yang, Y. Yu, and C. Grebogi, “An elderly care system based on multiple information fusion,” *Journal of Healthcare Engineering*, vol. 2018, Article ID 8102597, 13 pages, 2018.
  - [24] Y. P. E. Wang, X. Lin, A. Adhikary et al., “A primer on 3GPP narrowband internet of things (NB-IoT),” *IEEE Communications Magazine*, vol. 55, no. 3, 2016.
  - [25] Z. Cao, G. Hidalgo, T. Simon et al., “OpenPose: realtime multi-person 2D pose estimation using part affinity fields,” 2019, <https://arxiv.org/abs/1812.08008>.
  - [26] M. Datar, A. Gionis, P. Indyk et al., “Maintaining stream statistics over sliding windows (extended abstract),” in *Proceedings of the 13th ACM-SIAM Symposium on Discrete Algorithms*, San Francisco, CA, USA, 2002.
  - [27] J. H. Friedman, “Greedy function approximation: a gradient boosting machine,” *The Annals of Statistics*, vol. 29, no. 5, pp. 1189–1232, 2001.
  - [28] S. S. Khan and J. Hoey, “Review of fall detection techniques: a data availability perspective,” *Medical Engineering & Physics*, vol. 39, pp. 12–22, 2017.
  - [29] L. Liu, T. C. Lu, Y. Y. Lv et al., “MEMS tri-axial accelerometer based fall detection,” *Chinese Journal of Sensors and Actuators*, vol. 4, pp. 570–574, 2014.
  - [30] R. Polikar, “Ensemble learning,” *Scholarpedia*, vol. 4, no. 1, pp. 1–34, 2009.
  - [31] B. S. Everitt, *Classification and Regression Trees—Encyclopedia of Statistics in Behavioral Science*, John Wiley & Sons, Hoboken, NJ, USA, 2005.
  - [32] M. A. Hearst, S. T. Dumais, E. Osuna, J. Platt, and B. Scholkopf, “Support vector machines,” *IEEE Intelligent Systems and Their Applications*, vol. 13, no. 4, pp. 18–28, 1998.
  - [33] M.-L. Zhang, J. M. Peña, and V. Robles, “Feature selection for multi-label naive Bayes classification,” *Information Sciences*, vol. 179, no. 19, pp. 3218–3229, 2009.
  - [34] J. R. Quinlan, “Induction of decision trees,” *Machine Learning*, vol. 1, no. 1, pp. 81–106, 1986.
  - [35] K. Q. Weinberger and L. K. Saul, “Distance metric learning for large margin nearest neighbor classification,” *Journal of Machine Learning Research*, vol. 10, no. 1, pp. 207–244, 2009.
  - [36] L. K. Hansen and P. Salamon, “Neural network ensembles,” *IEEE Transactions on Pattern Analysis & Machine Intelligence*, vol. 12, no. 10, pp. 993–1001, 2002.
  - [37] V. Svetnik, “Random forest: a classification and regression tool for compound classification and QSAR modeling,” *Journal of Chemical Information & Computer Sciences*, vol. 43, no. 6, pp. 1947–1958, 2002.