

## Research Article

# Preimpact Fall Detection for Elderly Based on Fractional Domain

Ning Liu , Dedi Zhang, Zhong Su, and Tianrun Wang

Beijing Key Laboratory of High Dynamic Navigation Technology, Beijing Information Science & Technological University, Beijing 100101, China

Correspondence should be addressed to Ning Liu; [ning.liu@bistu.edu.cn](mailto:ning.liu@bistu.edu.cn)

Received 12 December 2020; Revised 7 February 2021; Accepted 10 February 2021; Published 26 February 2021

Academic Editor: Ali Ahmadian

Copyright © 2021 Ning Liu 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.

The aging population has become a growing worldwide problem. Every year, deaths and injuries caused by elderly people's falls bring huge social costs. To reduce the rate of injury and death caused by falls among the elderly and the following social cost, the elderly must be monitored. In this context, falls detecting has become a hotspot for many research institutions and enterprises at home and abroad. This paper proposes an algorithm framework to prealarm the fall based on fractional domain, using inertial data sensor as motion data collection devices, preprocessing the data by axis synthesis and mean filtering, and using fractional-order Fourier transform to convert the collected data from time domain to fractional domain. Based on the above, a multilayer dichotomy classifier is designed, and each node parameter selection method is given, which constructed a preimpact fall detection system with excellent performance. The experiment result demonstrates that the algorithm proposed in this paper can guarantee better warning effect and classification accuracy with fewer features.

## 1. Introduction

According to a report of the World Health Organization (WHO) in 2018, about 400000 people die each year from fall around the world; fall has become the second leading cause of unintentional injury death and is also the number one cause of injury and death among the elderly in the community. The people needed for treatment of serious falls are 37.3 million per year [1], the fall in the low- and middle-income countries deaths accounted for 80% of the fall in the global death toll, and the proportion of people over 65 is the largest [2]. According to the 2018 Statistical Bulletin Social Service Development released by Ministry of Civil Affairs (MOCA) published, by the end of 2018, there are 249.49 million elderly people at or above the age of 60, accounting for 17.9 percent of the total population, of which 166.58 million are aged 65 or above, accounting for 1.9 percent of the total population. According to statistics, at least 20 million elderly fall 25 million times a year, directly resulting in medical costs more than 5 billion yuan. Every year in China, the social cost caused by elderly falls is estimated to be 16 billion~80 billion; this cost will dramatically increase along with the aggravation of the aging population. To bring

down the injury and death rate of elderly fall and the resulting social cost, the elderly must be detected [3], protected, or intervened for falls, and the premise of protection or intervention is the accurate monitoring of falls. In this context, fall monitoring has become a research hotspot for many scientific research institutions and enterprises at home and abroad.

Currently, research directions for fall monitoring are mainly divided into two aspects: postfall identification, namely, fall detection [4], and prefall detection [5–11], namely, preimpact fall detection [2].

Among them, fall detection is an effective fall protection strategy, in which the main idea is to automatically detect the occurrence of falls, so as to timely send out alarm and help information, save the time of rescue arrival, and thus greatly reduce the disability rate and death rate caused by falls. Although this method can accurately find the occurrence of falls, it still has its inherent limitations; that is, the detection takes place after the fall and the injury caused by the fall has already occurred, so the fall detection cannot fundamentally avoid the casualties caused by the fall.

Different from fall detection, the main idea of preimpact fall detection is to detect the occurrence of falls in the process

from the occurrence of falls to the occurrence of an impact. Therefore, preimpact fall detection can not only provide timely fall alarm and help and shorten the time of rescue arrival, but also break through the inherent limitations of fall detection, and work in cooperation with fall protection or intervention equipment to intervene or protect the person who falls, thereby radically avoiding or reducing the casualties caused by falls.

Several solutions for fall detection have been presented in the literature in recent years. In [12], an accelerometer installed at the waist detects the vertical acceleration of the inertial frame near the center of gravity of the body. The threshold method is used to distinguish the ongoing and unrecoverable fall motion from the nonfall motion. The advance time is 70 ms to 375 ms. In [10], acceleration, angular velocity, and body inclination are measured by inertial sensors, and the threshold method is used for fall detection. The lead time is  $401.9 \pm 46.9$  ms, the sensitivity is 100%, and the specificity is 100%. Similarly, the authors of [11] use inertial sensors to measure the vertical velocity and displacement, and the threshold method is used for fall warning. The sensitivity is 93.6% and the specificity is 95.6%. Besides, in [13], the authors record 3D movements of the whole body, measure the linear acceleration of all body parts, and classify walking from accidental disturbances through artificial neural networks. The average detection time is  $351 \pm 123$  ms, and the accuracy is 95.4%. They also proposed that only by observing the feet and hands, the appropriate performance in terms of average detection time and accuracy can be obtained. The authors of [14] used the waist inertial sensor to measure acceleration and angular velocity and used the SVM algorithm based on hidden Markov model (HMM) for fall warning. The accuracy is 94.91%, the sensitivity is 97.22%, and the specificity is 93.75%.

The present study tests the following hypotheses:

Hypothesis 1. Testers with different ages and physical fitness have an impact on the test results.

Hypothesis 2. Preimpact fall detection algorithm based on fractional domain has higher classification performance and accuracy.

The present study tests have the following limitations:

Limitation 1. The data source for this test is young people aged 20–25. It is not representative in terms of age.

Limitation 2. The algorithms selected in this article use MATLAB for simulation.

Limitation 3. The algorithm proposed in this paper is only suitable for fall detection and is not universal.

This article mainly focuses on the research on efficient classification and early warning methods for the elderly to fall. The testers imitate the elderly walking, jogging, tripping, fainting, slipping while standing, and choosing wearables Inertial data sensor that includes a three-axis accelerometer and a three-axis gyroscope as data acquisition devices [15], preprocessing the data using the average filtering method and the fractional Fourier transform, selecting the

classification algorithm that is easy to implement on the embedded system [16] and the appropriate motion feature as group standard, constructing a multilayer dichotomy classifier, and finally performing experimental verification.

The rest of this article is arranged as follows. Section 2 is the theoretical basis; it introduces the method of inertial data acquisition and the principle of fractional Fourier transform and then applies it to the domain transformation of data to highlight different motions at different transformation orders. And Section 3 analyzes and compares fall detection algorithms based on the time domain threshold method, machine learning, and fractional domain. The experiments carried out are contained in Section 4. Finally, Section 5 describes the conclusions.

## 2. Theoretical Basis

*2.1. Inertial Data Collection.* The experimental data were collected through 15 wearable inertial sensors integrating triaxial accelerometers and three-axis gyroscope. The inertial sensors are shown in Figure 1, and the map of wearables on the body is shown in Figure 2, among which the No. 12 sensor and the No. 8 sensor were not worn in the experiment. The main parameters of the sensor are shown in Table 1.

In this paper, falls are classified into tripping, slipping, and fainting under the state of walking or jogging [17]. By imitating the fall of the elderly in the above common motion state, the tester uses the above data acquisition device to collect the inertial data during the moving process. There were three participants in the experiment; all were males. The experimental scenario is shown in Figure 3.

*2.2. Domain Transformation.* For human motion data, it is usually analyzed from the perspective of time domain or frequency domain. At present, most of the documents directly extract the corresponding features from the time domain for classification or judgment. Considering that human movement also contains certain frequency information, the accuracy of classification can be improved to some extent by introducing frequency-related features. Therefore, by introducing fractional Fourier transform, this paper extracts features from the perspective of the fraction domain and makes it contain both time domain and frequency information [18].

*2.2.1. Fractional Fourier Transform Theory.* In the field of signal processing, the traditional Fourier transform is a mature and widely used mathematical tool. In 1980, V. Namias proposed the concept of fractional Fourier transform (FRFT) in a purely mathematical way, from the perspective of the characteristic value and characteristic function [19]. Subsequently, some researchers proposed the concept of fractional Fourier transforms from the perspective of optics. It can be proved that these definitions are completely equivalent. Because fractional-order Fourier transform can be realized by simple optical equipment, it is first applied in optical signal processing. Until recent years,



FIGURE 1: Mini wireless IMU devices.

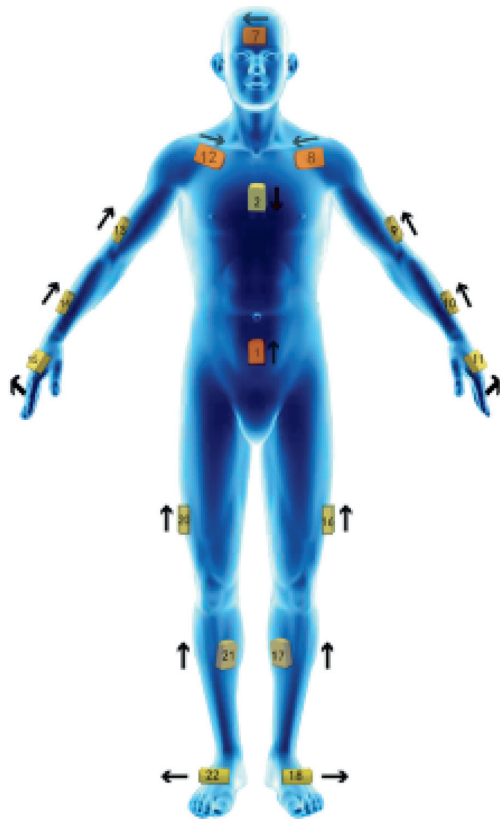


FIGURE 2: The wearing-on diagram of inertial data acquisition devices.

scholars at home and abroad have found several fast algorithms of fractional Fourier transform, which has attracted much attention in the application of signal processing and other fields.

Fractional Fourier transform is the signal representation method in the fractional Fourier domain which, formed by the coordinate axis, rotates counterclockwise around the origin at any angle in the time-frequency plane. It is a time-frequency analysis method and a generalized Fourier transform. The fractional Fourier transform has many properties that the traditional Fourier transform does not have and can be widely used in scientific research and engineering technology.

The fractional Fourier transform is defined as follows [19].

Generally, the  $p$ -order of fractional Fourier transform of function  $f(u)$  can be represented as follows:  $f_p(u)$  or  $F^p f(u)$ ; among them,  $F^p f(u)$  can be explained by the operator  $F^p$  effects on function  $f(u)$ , and its results are located on  $u$  domain.

The basic definition of fractional Fourier transform is

$$f_p(u) = \int_{-\infty}^{+\infty} K_p(u, t) f(t) dt. \quad (1)$$

Among them,  $K_p(u, t) = \{A_\alpha \exp[j\pi(u^2 \cot \alpha - 2ut \csc \alpha + t^2 \cot \alpha)], \alpha \neq n\pi \delta(u - t), \alpha = 2n\pi \delta(u + t), \alpha = (2n + 1)\pi$  is the kernel function of the fractional Fourier transform,  $A_\alpha = \exp[-j\pi \operatorname{sgn}(\sin \alpha)/4 + j\alpha/2]/|\sin \alpha|^{1/2}$ ,  $\alpha = p\pi/2$ , and  $n$  is an integer.

The following can be derived from the above:

$$f_\alpha(u) = A_\alpha T_t(u) \int_{-\infty}^{+\infty} T_s(u - x) [T_t(x) f(x)] dx. \quad (2)$$

Among them,  $T_t(x) = \exp(-j\pi t x^2)$ ,  $t = \tan(\alpha/2)$ ,  $s = -\csc(\alpha)$ .

Notice  $F^{4n}$  and  $F^{4n \pm 2}$  are, respectively, equal to the unit operator  $\tau$  and parity operator  $P$ . For  $p = 1$ , we have  $\alpha = \pi/2$ ,  $A_\alpha = 1$ , and  $f_1(u) = \int_{-\infty}^{+\infty} e^{-j2\pi ut} f(t) dt$ .

Therefore,  $f_1(u)$  is the ordinary Fourier transform of  $f(u)$ , and the function of zero order transformation is defined as equal to the function itself. Since the  $\alpha = p\pi/2$  only appeared on the position of the trigonometric function parameter,  $p$  (or  $\alpha$ ) as a parameter is based on the definition of 4 (or  $2\pi$ ) as a cycle.

The reason for choosing fractional Fourier transform to extract the feature vector, besides comparing the time domain and frequency domain, is that Li Chao derived the relationship between the time domain amplitude average  $\bar{A}$  of the signal and the amplitude  $B$  of the fractional domain from the theoretical time width-bandwidth product [20].

According to the fractional domain sampling theorem, the relationship between fractional bandwidth and frequency bandwidth is

$$Bu = B \sin \alpha, \quad (3)$$

where  $\alpha$  is the rotation angle of the frictional domain and  $\alpha = p\pi/2$ .

TABLE 1: Parameters of sensor.

Parameters	Index	Parameters	Index
Accelerometer range	$\pm 18 \text{ g}$	Sample frequency	$30 \text{ Hz}$
Gyroscope range	$\pm 3200 \text{ deg/s}$	Gyroscope static accuracy	$\leq 0.5 \text{ deg}$ (pitching, rolling); $\leq 1 \text{ deg}$ azimuth
Angular resolution	$\leq 0.05 \text{ deg}$	Gyroscope dynamic accuracy	$\leq 1 \text{ degRMS}$



(a)

(b)



(c)

FIGURE 3: Experimental scenario. (a) Movement data collection of prosthetic limb wearers. (b) Imitating the elderly slipping. (c) Imitating the elderly fainting.

The time width and bandwidth of fractional domain signal are defined as follows:

$$\Delta t^2 \triangleq \int_{-\infty}^{+\infty} |(t - t_0)x(t)|^2 dt, \quad (4)$$

$$\Delta u_\alpha^2 \triangleq \int_{-\infty}^{+\infty} |(u_\alpha - u_{\alpha_0})x(u_\alpha)|^2 du_\alpha. \quad (5)$$

Among them,  $u_\alpha$  for frequency of fractional domain and  $x(u_\alpha)$  is the fractional domain transformation of signal  $x(t)$ .

We know the following from the indefinite principle of the signal frequency domain:

$$\Delta t^2 \cdot \Delta u^2 \geq \frac{1}{4} \text{ is } \left( \Delta t \cdot \Delta u \geq \frac{1}{2} \right). \quad (6)$$

The indefinite principle of the fractional domain can be derived from (4)-(6):

$$\Delta t^2 \cdot \Delta u_\alpha^2 \geq \frac{\sin^2 \alpha}{4} \text{ is } \left( \Delta t \cdot \Delta u_\alpha \geq \frac{\sin \alpha}{2} \right). \quad (7)$$

According to Parseval's theorem, the energy of the signal in the time domain is the same as that in the fractional domain:

$$E = E_\alpha. \quad (8)$$

Among them,  $E$  represents total energy in time domain and  $E_\alpha$  represents total energy in frictional domain. Moreover,

$$E = \frac{1}{2} \bar{A} \cdot \Delta t^2, E_\alpha = \frac{1}{2} \bar{B} \cdot \Delta u_\alpha^2. \quad (9)$$

Considering (7)–(9), the average amplitude  $\bar{A}$  of signal in time domain has a nonlinear relationship with amplitude  $\bar{B}$  in fractional domain and is related to the angle change  $\alpha$  of the fractional domain.

Hence, we can convert the sequence of values with no significant difference in amplitude in the time domain into a sequence of fractions with significant differences in amplitude by selecting an appropriate FRFT order [20]. The method can be used to recognize various signals of motion feature effectively and achieve the purpose of recognition.

**2.2.2. Discrete Fractional Fourier Transform.** Equation (2) is a calculation method in the continuous domain. This continuous transformation cannot be calculated in practice. It is usually necessary to sample and interpolate the continuous signal for numerical calculation. Using Shannon's interpolation formula and numerical integration operation, the discrete calculation formula of the fractional Fourier transform is shown in the following equation [19]:

$$f_\alpha\left(\frac{k}{2\Delta}\right) \approx \frac{A_\alpha}{2\Delta} \exp\left[-j\pi \tan(\alpha/2)\left(\frac{k}{2\Delta}\right)^2\right] \cdot \sum_{l=-N}^{N-1} \exp\left[j\pi\left(\frac{k-l}{2\Delta}\right)^2 \csc \alpha\right] \cdot \left\{ \exp\left[-j\pi\left(\frac{l}{2\Delta}\right)^2 \tan(\alpha/2)\right] f\left(\frac{l}{2\Delta}\right) \right\}. \quad (10)$$

The derivation can be shown as below:

$$f_\alpha(x_k) \approx \frac{x_k C_\alpha}{k} E_t(u_k) \sum_{l=-N}^{N-1} \{E_s(x_{k-l}) [E_t(x_l) f(x_l)]\}. \quad (11)$$

Among them,  $E_t(u_k) = \exp(-j\pi t u_k^2)$ ,  $t = \tan(\alpha/2)$ ,  $s = -\csc(\alpha)$ , and  $x_k = u_k = k/2\Delta \cdot \Delta$  is dimensional normalization scale of time domain or frequency domain.

**2.2.3. Fractional Fourier Transform Effect.** According to the above theory, we could use fractional Fourier transform to process the inertial data collected by the inertial sensor. Compared with the simple time domain and frequency domain, the fraction domain has better diversity and can provide more optional characteristic value.

First of all, it is necessary to compare the output of various data after different or the same order of fractional Fourier transform: the results of the same set of data after different order [20] transformation are shown in Figure 4. The results of the same class of data after the transformation of the same order are shown in Figure 5, and the results of different classes of data after the transformation of the same order are shown in Figure 6.

It can be found that the same set of data has obvious differences after transformation of different orders, and data of different types also have certain differences after transformation of the same order. Therefore, we can choose the appropriate order to maximize the difference and then use the appropriate characteristic value to distinguish different motion states.

**2.3. Selection of Characteristic Value.** The experimental data are collected through 15 inertial sensors that integrate three-axis accelerometers and three-axis gyroscopes worn on the body. After the collected data are subjected to fractional Fourier transform, appropriate characteristic value needed to be extracted and then classified based on the characteristic value. However, not all characteristic values are suitable for inertial data. In addition, redundant characteristic values will lead to excessive calculations, time costs, and storage costs and will also reduce the battery life of the device, while too few characteristic values cannot achieve the accuracy required to distinguish various motions, so appropriate characteristic values need to be selected for motion posture recognition.

According to the characteristics of inertial data and fractional Fourier transform, ten characteristic values including the maximum value, maximum absolute value, maximum difference, standard deviation, variance, root mean square, interquartile difference, mean value, peak mean value, and peak number are selected as initial features to construct feature vector, as shown in the following equation:

$$F = [\text{Max MaxABS Range Std Var Rms IQR Mean PksMean PksNum}]. \quad (12)$$

### 3. Preimpact Fall Detection Algorithm

In the field of preimpact fall detection, there are mainly two kinds of commonly used classification algorithms: threshold method [21] and machine learning method [13]. Threshold method refers to judging whether a fall occurs by determining whether a certain feature or several features exceed

the preset threshold [22] in the process of movement. Machine learning trains more accurate prediction models through appropriate features and then judges whether a fall occurs by using the model. In this chapter, the algorithm design of preimpact fall detection is studied from the view of fraction domain and compared with the threshold method and machine learning method in time domain.

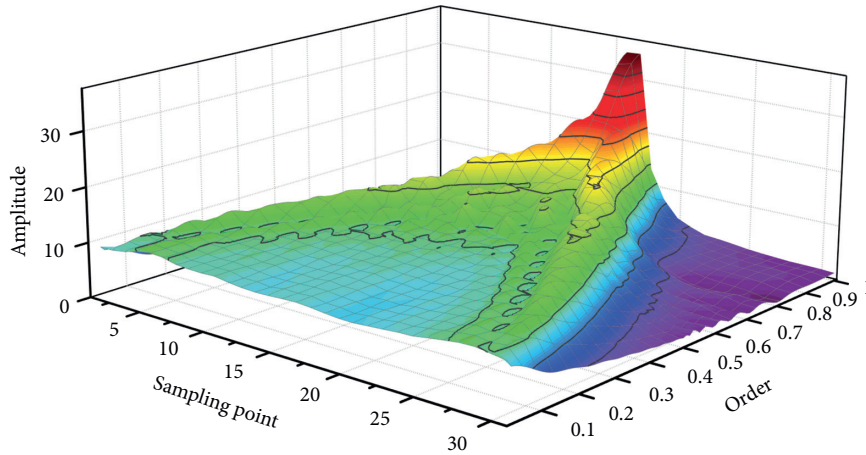


FIGURE 4: 3D diagram of fractional Fourier transform effects of different orders in the same set of data.

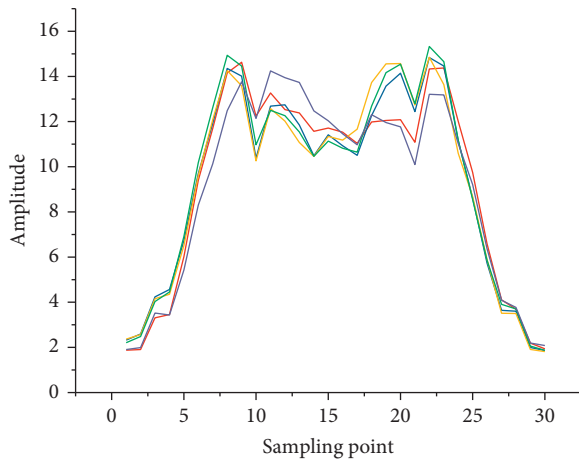


FIGURE 5: The result of the same order transformation of same kind of motion.

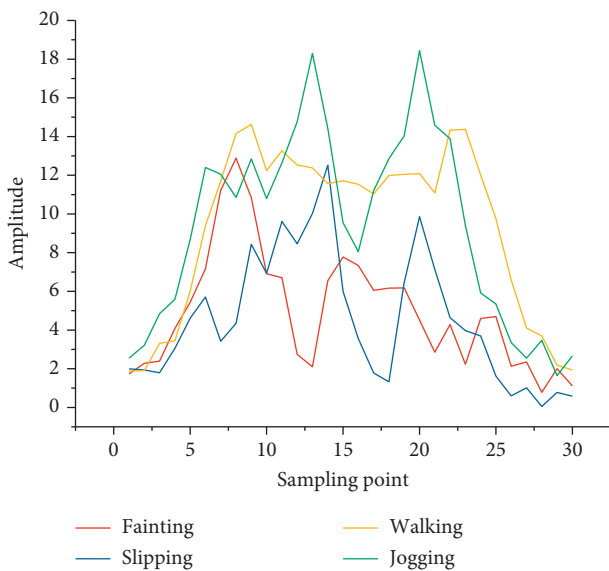


FIGURE 6: The result of the same order transformation of different motions.

### 3.1. Preimpact Fall Detection Based on Threshold Method in Time Domain

3.1.1. *Prealarm Theory.* Based on the time-domain threshold method, the preimpact fall detection is to directly process the acceleration and angular velocity collected by the inertial sensor and set an appropriate threshold value. When the collected data exceed the threshold value, the fall trend is judged. For a piece of data collected in real time, it can be expressed in the following form:

$$\text{Max}(X_t) > T. \tag{13}$$

Among them,  $X_t$  represents the characteristics of the data collected at time  $t$  and  $T$  represents the preset threshold.

3.1.2. *Feature Selection.* In general, the threshold method can choose one feature as the maximum value of the data collected in real time or a combination of multiple features as the classification basis. Generally, a single feature can guarantee a certain precision rate and recall rate. Appropriately increasing the number of features can improve the precision rate and recall rate; on the contrary, excessive features will reduce the precision rate and recall rate. In this paper, four characteristics value including global maximum absolute value (GMA), the global maximum value (GM), interval maximum value (IM), and interval maximum absolute value (IMA) are analyzed.

Figure 7 shows the changes of the characteristics in the time domain with the original data before and after the fall. Among them, Figure 7(a) is the characteristic change with the original angular velocity, and Figure 7(b) is the characteristic change with the original acceleration. There is a clear difference in the amplitude of the inertial data generated by regular exercise and falling. Therefore, falling can be detected by setting the threshold. Among these features, the maximum absolute value of the interval can quickly reflect the occurrence of a fall, while the global maximum can ensure a lower recall rate. Thus, these two features are selected for further classification analysis.

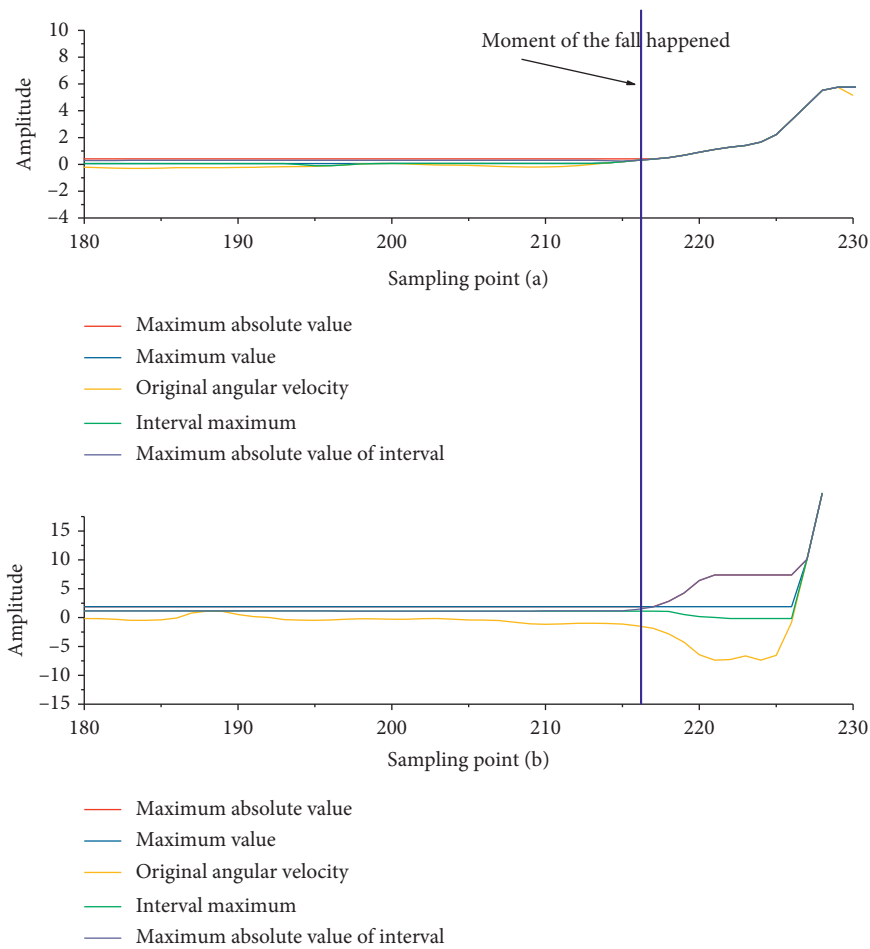


FIGURE 7: The feature in time domain change with original data. (a) The feature varies changes with the original angular velocity. (b) The feature varies changes with the original acceleration.

3.1.3. *Classification Process and Result Analysis.* In this paper, the average  $T$  of the maximum absolute value of the training sample at the time of the fall is calculated, and the threshold is ranged from 60% to 140% of the average. The time  $t_p$  when the threshold is reached is regarded as the predicted time of the fall. We compare  $t_p$  with the actual time of fall  $t_r$  to get the delay time  $t_d$ , while comparing  $t_p$  with the time  $t_c$  that falling collision happened to calculate the lead time  $t_a$ . We count the number of misjudgments  $N_w$  that mixed up a normal movement with a fall and finally evaluate the pros and cons of the classification effect based on the above parameters.

Table 2 shows the prealarm effect of fainting. When the threshold is  $1.32 \text{ m/s}^2$ , there is a misjudgment. When the threshold is  $1.54 \text{ m/s}^2$  while the number of misjudgments is 0, the lead time is the largest.

Table 3 shows the average optimal early warning effect for independent monitoring of falls. It can be seen that when a single fall state is monitored independently, the average advance time without misjudgment can be as high as 413 ms. However, there is a large gap between the threshold values of each fall state. Here, take the 80%–120% of the maximum threshold value and give the early warning for all fall states under the same threshold condition.

Table 4 shows the optimal prealarm effect for all fall states under the same threshold condition ( $8.49 \text{ m/s}^2$ ). It can be seen that multiple fall states need to be monitored at the same time. Compared to independent monitoring, the prealarm performance is obviously reduced.

The fall data of another tester were verified under the condition of keeping the threshold value of  $8.49 \text{ m/s}^2$ , and the prealarm effect is shown in Table 5.

From the data in Table 5, it can be found that walking-slipping and jogging-slipping cannot be detected. It can be seen that the threshold method is seriously restricted by personnel, and the threshold used by different testers is significantly different. In addition, the threshold method also has the defect that it cannot distinguish the type of fall.

3.1.4. *Preimpact Fall Detection Based on Machine Learning.*

In addition to the threshold method, machine learning is also a common class of preimpact fall detection algorithms. Common machine learning algorithms include support vector machine (SVM) [14, 23–25], decision tree (DT), artificial neural network (ANN) [26], and k-nearest neighbor (K-NN) [27], which all can be used for model construction. Some have preferable performance including

TABLE 2: Prealarm effect of fainting.

Threshold $T$ ( $m/s^2$ )	Delay time $T_d$ ( $ms$ )	Lead time $T_a$ ( $ms$ )	Misjudgment frequency $N_w$
1.32	100	400	1
1.54	133	367	0
1.76	133	367	0
2.20	167	333	0

TABLE 3: The average optimal prealarm effect for independent monitoring of falls.

Falls type	Threshold $T$ ( $m/s^2$ )	Delay time $T_d$ ( $ms$ )	Lead time $T_a$ ( $ms$ )
Walking-fainting	1.76	87	413
Walking-slipping	2.27	108	375
Walking-tripping	2.06	153	506
Jogging-fainting	6.08	180	260
Jogging-slipping	6.82	426	193
Jogging-tripping	7.72	453	100

TABLE 4: The optimal prealarm effect under the same threshold condition.

Falls type	Delay time $T_d$ ( $ms$ )	Lead time $T_a$ ( $ms$ )
Walking-fainting	300	233
Walking-slipping	433	67
Walking-tripping	300	400
Jogging-fainting	267	233
Jogging-slipping	433	167
Jogging-tripping	367	67

TABLE 5: The prealarm effect of another tester under the condition of constant threshold.

Falls type	Delay time $T_d$ ( $ms$ )	Lead time $T_a$ ( $ms$ )
Walking-fainting	567	67
Walking-slipping	-	-
Walking-tripping	233	93
Jogging-fainting	367	100
Jogging-slipping	-	-
Jogging-tripping	300	567

convolutional neural network (CNN), a class of support vector machine (1SVM), CNN+1SVM [28], hidden Markov model (HMM) [29], K-NN, and ANN [30].

This paper used the MATLAB machine learning toolbox to employ commonly used machine learning algorithms to conduct classification training on the training set and selected two algorithms with the best classification results. Table 6 shows the selected classification algorithms and classification accuracy.

The data in the test set are intercepted through a sliding window simulating real-time data reading. After extracting its features, the data are sent to the trained model for classification. Finally, the classification results are compared with the actual motion curves. Figure 8 is the real-time simulation classification rendering of walking-fainting. Among them, the red curve represents the original data, the blue curve represents the classification effect of the weighted KNN algorithm, and the yellow curve represents the classification effect of the bagged trees algorithm of integrated learning. Among the classification results, 1–6 correspond to walking, jogging, fainting, tripping, and slipping, respectively.

TABLE 6: The selected classification algorithm and classification accuracy.

Classification algorithm	Accuracy
Integrated learning bagged trees	88.9%
Weighted KNN	89.2%

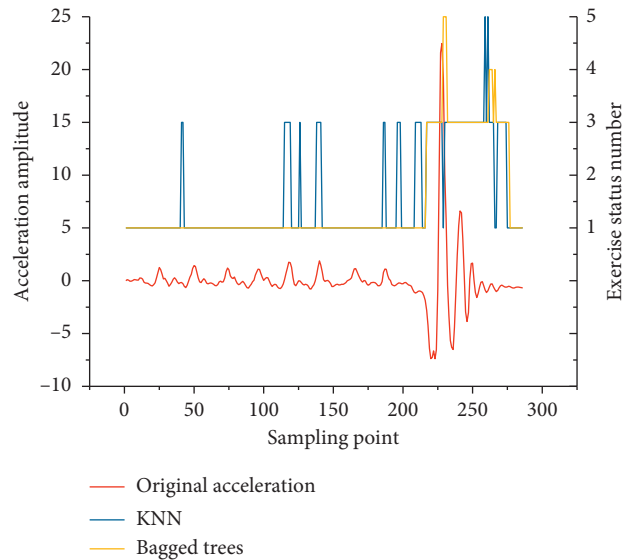


FIGURE 8: Walking-fainting classification rendering.

It can be seen that the bagged trees algorithm of integrated learning has almost no misjudgment before falling, with an accuracy rate of 98.62%, while the weighted KNN algorithm has several misjudgments with an accuracy rate of 88.94%. Moreover, both algorithms give an alarm at the time of 217 sampling points, which are 11 sampling points in advance, namely, 367 ms, compared with 228 sampling points where the fall collision [12] occurs.

Table 7 shows the fall prealarming accuracy and lead time of part of the test set. On the left side, we have the prediction result of the KNN algorithm, and the prediction result of the integrated learning algorithm is on the right side. The lead time is calculated by the time difference



TABLE 7: The fall prealarming accuracy and lead time of part of the test set.

Motion	Accuracy of data set 1	Lead time of data set 1 (ms)	Accuracy of data set 2	Lead time of data set 2 (ms)
Walking-fainting	0.889-0.880	367-433	0.917-0.856	567-500
Walking-slipping	0.845-0.743	333-233	0.880-0.789	433-367
Walking-tripping	0.753-0.688	433-367	0.838-0.668	433-668
Jogging-fainting	0.664-0.930	467-333	0.651-0.846	467-433
Jogging-slipping	0.515-0.874	267-267	0.571-0.920	200-267
Jogging-tripping	0.617-0.807	200-267	0.631-0.719	267-433

Note. The prediction result of KNN algorithm is on the left side, and the prediction result of integrated learning algorithm is on the right side.

between the time when the fall was predicted to be correct for the first time and the time when the fall collision occurred. It can be seen that walking in the motion state is more likely to predict falling than jogging in the motion state, and jogging state is more difficult to distinguish from falling state in the time domain.

**3.2. Preimpact Fall Detection Algorithm Based on Fractional Domain.** On the basis of the research in Sections 3.1 and 3.2, considering that human body motion also contains certain frequency information, if frequency-related features are introduced, the classification accuracy can be improved to a certain extent. Therefore, this paper considers introducing the fractional Fourier transform to extract features from the perspective of the fractional domain, so that it contains both information of time domain and frequency. The theory of the fractional Fourier transform has been explained in Section 2.2. In this section, the data collected in the sliding window take the mixed order fractional Fourier transform, converted to the mixed order fractional domain, and then feature extraction is performed. According to the pairwise classification of these features, the most effective combination and the relief algorithm are used to calculate the weights of each feature and reduced the dimension. Finally, a multilayer dichotomy classifier is constructed.

**3.2.1. Classification Method Selection.** To apply the fractional Fourier transform to fall prealarming, the classification algorithm must first be determined. In the commonly used threshold method and machine learning method, although the threshold method is simple to calculate, it is limited by the data and difficult to achieve the optimization by setting the threshold based on experience. Although most machine learning methods can provide high classification performance, the calculation of their prediction results is more complicated, which is not easy to implement on the on-board chip and will significantly increase computing power consumption and reduce battery life. Support vector machine (SVM) have mature and optimized classification algorithms, which is also suitable for implementation on on-board low-power chips. Thus, support vector machines are the most suitable choice.

Considering the classifier design idea, each step of classification is a dichotomy classification, and the number of features used in the final classification needs to be reduced as much as possible. After verification, only 2–3 features are required to achieve a better classification effect. Therefore,

the kernel function of SVM can be defined as the simplest linear kernel function or polynomial kernel function, and its expression is shown in the following:

$$k(x, y) = x^T y + c, \quad (14)$$

$$k(x, y) = (ax^T y + c)^d. \quad (15)$$

In summary, this article sets the classification algorithm of each subnode to SVM and directly uses the quadratic SVM tool in Matlab classification learner toolbox for classification analysis.

**3.2.2. Evaluation Method of the Transformation Order of the Best Classification.** Considering that the transformation order and features required to separate any two motion states may be different, this paper uses the traversal method to calculate the classification effect of any two sets of motion states using all features at all orders.

Table 8 is the best training effect of any two kinds of sports classified by the 20 sets of features calculated from the acceleration and angular velocity collected by the waist sensor. The upper part and the lower part, respectively, indicate the sensitivity and specificity of the classification. Table 9 shows the order of the best classification effect.

It can be seen from Table 9 that there are two cases for the best classification effect: the classification of walking and jogging has better sensitivity under the 0.3-order transformation and better specificity under the 0.1-order transformation; that means the best sensitivity and best specificity are not in the same transformation order. The sensitivity and specificity of walking-slipping classification can reach 100% under 0.2-order transformation; that is, the best sensitivity and best specificity are in the same transformation order. For the latter, the order can be directly used for classification, while for the former, a new order needs to be found that can satisfy high sensitivity and specificity at the same time.

ROC (Receiver Operating Characteristic) curve is usually used in machine learning to reflect the changes of model performance with the changes of a specific external stimulus. Referring to the ideas of the ROC curve, the true positive rate (TPR) and the false positive rate (FPR) were plotted as the ROC curve with the transformation order as the external stimulus condition during the process that the transformation order change from 0 to 1 in the process plots order. According to the

TABLE 8: The best accuracy rate using all features for any two sets of motions at all transformation orders.

Motion type	Walking	Jogging	Fainting	Tripping	Slipping
Walking		0.9943	0.9903	0.9961	1
Jogging	0.9954		0.9904	1	0.9976
Fainting	0.9942	0.9928		0.9821	1
Tripping	1	1	0.9821		0.9483
Slipping	1	0.9976	1	0.9655	

TABLE 9: The order of the best accuracy.

Motion type	Walking	Jogging	Fainting	Tripping	Slipping
Walking		0.3	0.9	0.2	0.2
Jogging	0.1		0.1	0.8	0.9
Fainting	0.8	0.8		0.1	0.2
Tripping	0.4	0.2	0.3		0.1
Slipping	0.2	0.1	0.4	0.6	

nature of the ROC curve, the point closer to the coordinate (0, 1) indicates that the classification effect of the point is better. Define the rectangular area  $S_{TF}$  surrounded by four straight lines  $y = TPR$ ,  $y = 0$ ,  $x = FPR$ ,  $x = 1$  to represent the distance from (0,1). Draw the rectangular area corresponding to each order as a curve, as shown in Figure 9.

Figure 9 shows the curve of the classification effect of walking-fainting with the change of transformation order. Among them, Figure 9(a) is the ROC curve. Since the TPR and FPR evaluating the classification effect do not show monotonicity with the increase of the change order, thus Figure 9(a) is not the ROC curve in the usual sense. Figure 9(b) is the curve. According to the property of the ROC curve, the curve directly reflects the classification effect. Therefore, the optimal classification order and the corresponding optimal classification effect of this order can be determined according to Figure 9(b).

In this figure, the classification effect is best after the fractional Fourier transform of order 0.4, with the corresponding true positive rate of 99.74% and false positive rate of 2.86%.

In summary, on the basis of using all the feature values, the best transformation order for distinguishing walking and fainting through the waist sensor is 0.4 order.

**3.2.3. Feature Dimension Reduction.** The feature vector used in the current classification is shown in (12); both acceleration and angular velocity need to be calculated once for each feature vector, so there are a total of 20 features actually used for classification. Considering that the classification algorithm is employed on a wearable device, exorbitant calculation amount will significantly increase the power consumption of the device, thereby reducing the battery life of the device. Thus, the number of features used in classification must be reduced.

Relief's feature classification algorithm is a feature dimensionality reduction algorithm mainly for dichotomy classification problems. Considering that the classifier used in this article is a multilayer dichotomy classifier, only two

classes of objects at a time, there is no need to use Relief's extended algorithm Relief-F.

This section still takes the classification of walking-fainting as an example to reduce the dimensionality of the walking-fainting data in the training set through the Relief algorithm. Table 10 is the evaluation value of each feature of the walking-fainting motion state under each order of fractional Fourier transform, in which MA stands for MaxABS, PM stands for PksMean, and PN stands for PksNum. The first ten features are calculated by acceleration, while the last ten features are calculated by angular velocity.

It can be seen from Table 10 that the Var., Std., and RMS of acceleration and the Var., MaxABS, and Std. of angular velocity have higher weights, indicating that these features have a higher contribution to distinguish between walking and fainting. The lower weights of the PksNum and PksMean features of acceleration and angular velocity indicate that these features have a lower contribution to distinguish between walking and fainting. The IQR of angular velocity has lower weights under the transformations of 0.6, 0.7, 0.8, and 0.9 order. Referring to the conclusion of previous Section 3.2.2, walking and fainting have the best classification effect under the 0.1-order and 0.4-order transformation, while the highest feature weight also appears under the 0.1-order transformation, indicating the reliability of the algorithm. Therefore, the feature can be filtered by the weight, namely, feature dimensionality reduction.

Considering the combination relationship between features, the classification effect of selecting the two features with the highest weights for classification is not necessarily better than the other feature combinations, so the optimal solution for feature dimensionality reduction is not necessarily the top feature. The main idea of the boosting algorithm in integrated learning is selected in this paper to make feature selection.

The main idea of the boosting algorithm is to construct a strong classifier through several weak classifiers, which is a serial iterative model. For the series of weak classifier combinations ABC, classifier A will generate a prediction result for the training sample, send the wrong part of this result to classifier B for training, and then send the wrong part of the prediction result of B into the classifier C for training. Through continuous iteration, several weak classifiers can be used to form a strong classifier with outstanding performance.

Referring to the main idea of the boosting algorithm, first the angular velocity var. of order 0.1 is selected, which is the feature with the highest weight in Table 10, and then the classification effect of the combination with other features based on this feature is traversed. Find the feature vector composed of two features with the best classification effect.

Table 11 shows the result of the feature pair composed of the angular velocity var. and other features and sent to the SVM for training. The first ten features in the table are calculated from acceleration, and the last ten are calculated from angular velocity.

Since the order and feature group of the maximum value in Tables 11 and 12 are not completely the same, this paper chooses the form of obtaining the maximum value of the product to select the final transformation order and feature

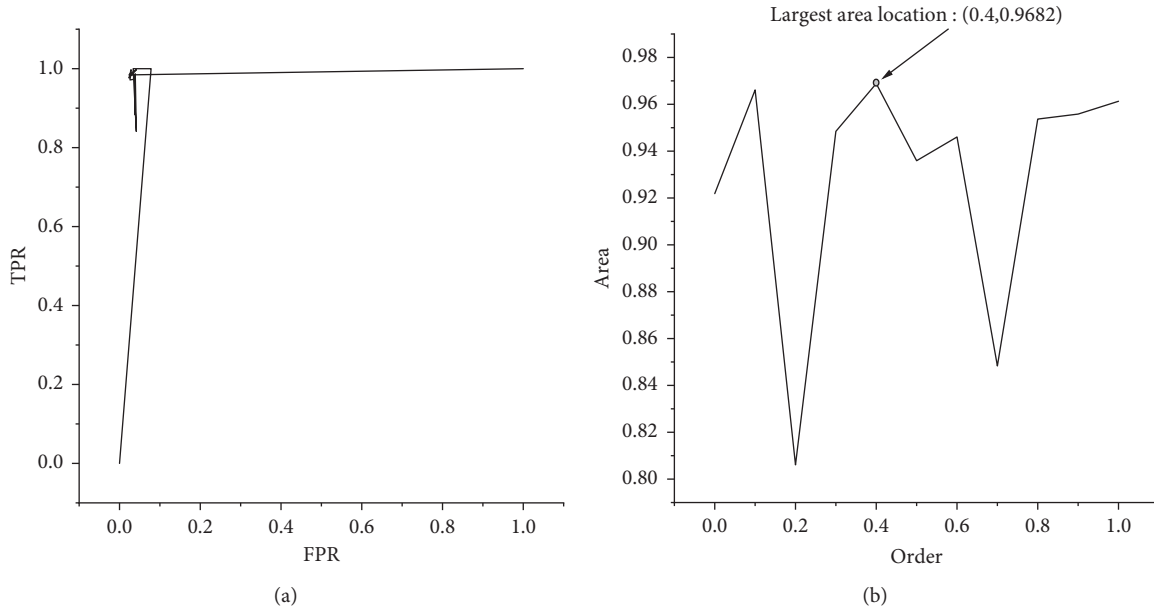


FIGURE 9: (a) ROC curve and (b)  $S_{TF}$  curve of walking-fainting with the change of transformation order.

TABLE 10: The evaluation value of each feature of walking-fainting motion state under each order of fractional Fourier transform.

Feature	0.1 order	0.2 order	0.3 order	0.4 order	0.5 order	0.6 order	0.7 order	0.8 order	0.9 order
Max	0.243	0.245	0.249	0.271	0.277	0.251	0.237	0.136	0.218
MA	0.243	0.245	0.249	0.271	0.277	0.251	0.237	0.136	0.218
Range	0.250	0.246	0.264	0.274	0.269	0.259	0.232	0.127	0.214
Std.	0.281	0.291	0.296	0.302	0.287	0.278	0.270	0.217	0.228
Var.	0.399	0.402	0.404	0.405	0.398	0.393	0.388	0.355	0.360
RMS	0.239	0.237	0.238	0.238	0.233	0.232	0.237	0.224	0.236
IQR	0.218	0.257	0.250	0.276	0.239	0.206	0.179	0.144	0.140
Mean	0.209	0.204	0.202	0.197	0.194	0.195	0.210	0.213	0.221
PM	0.168	0.176	0.192	0.175	0.171	0.152	0.124	0.107	0.174
PN	0.057	0.015	0.029	0.016	0.016	0.003	0.020	0.009	0.0050
Max	0.368	0.359	0.337	0.330	0.317	0.295	0.263	0.268	0.243
MA	0.368	0.359	0.337	0.330	0.317	0.295	0.263	0.268	0.243
Range	0.357	0.347	0.326	0.326	0.309	0.294	0.260	0.261	0.244
Std.	0.326	0.325	0.326	0.329	0.308	0.304	0.293	0.283	0.278
Var.	0.408	0.407	0.406	0.406	0.397	0.394	0.388	0.383	0.380
RMS	0.289	0.288	0.288	0.288	0.284	0.287	0.279	0.292	0.287
IQR	0.265	0.230	0.230	0.226	0.188	0.149	0.065	0.040	0.108
Mean	0.201	0.204	0.210	0.217	0.225	0.235	0.239	0.285	0.254
PM	0.164	0.211	0.124	0.142	0.059	0.122	0.025	0.082	0.035
PN	0.030	0.013	0.005	0.076	0	0	0.056	0.006	0.004

group. Table 13 is the result of multiplying the values in Tables 11 and 12 in the same position.

According to the calculation results in Table 13, take the maximum value to determine the classification parameters of walking and fainting: the angular velocity (var.) and acceleration (range) under the 0.1-order transformation. At this time,  $S_{TF}$  is 0.951.

**3.2.4. Multilayer Dichotomy Classifier Construction.** From the conclusions in Sections III-C-(b) and III-C-(c), the best classification order and feature group for separating a certain

individual motion state from all sports can be calculated in turn and then iteratively, until the classification of all motion states is completed. Finally, we build a multilayer dichotomy classifier on the grounds of the calculation results. In addition, to ensure efficiency, the feature values used for classification are limited to no more than three.

Table 14 shows the optimal classification effect between every two movements and the transformation order and the feature group used. Among them, the left side of the first row of each grid is that of the training set, and that of the test set is on the right side. In the second row of each grid, the transformation order is outside the brackets, and the feature

TABLE 11:  $S_{TF}$  of the trained SVM model composed of the angular velocity var. and other features.

Feature	0.1 order	0.2 order	0.3 order	0.4 order	0.5 order	0.6 order	0.7 order	0.8 order	0.9 order
Max	0.977	0.974	0.974	0.974	0.971	0.971	0.971	0.969	0.971
MA	0.977	0.974	0.974	0.974	0.971	0.971	0.971	0.969	0.971
Range	0.977	0.977	0.974	0.974	0.971	0.971	0.971	0.969	0.971
Std.	0.977	0.977	0.974	0.974	0.974	0.971	0.971	0.969	0.971
Var.	0.974	0.974	0.974	0.974	0.971	0.971	0.969	0.969	0.969
RMS	0.971	0.971	0.971	0.971	0.969	0.969	0.969	0.969	0.969
IQR	0.971	0.971	0.971	0.971	0.969	0.966	0.964	0.964	0.964
Mean	0.969	0.971	0.969	0.969	0.969	0.969	0.969	0.969	0.969
PM	0.971	0.969	0.966	0.969	0.969	0.969	0.966	0.966	0.966
PN	0.961	0.958	0.953	0.953	0.953	0.948	0.948	0.945	0.945
Max	0.966	0.961	0.961	0.958	0.953	0.948	0.951	0.943	0.721
MA	0.966	0.964	0.961	0.958	0.953	0.948	0.953	0.943	0.938
Range	0.966	0.964	0.961	0.956	0.953	0.948	0.953	0.945	0.922
Std.	0.961	0.961	0.953	0.953	0.953	0.953	0.948	0	0
Var.	0.958	0.958	0.953	0.953	0.953	0.948	0.864	0.909	0.919
RMS	0.969	0.961	0.956	0.953	0.951	0.948	0.953	0.969	0.953
IQR	0.964	0.961	0.961	0.958	0.953	0.953	0.948	0.945	0.945
Mean	0.977	0.964	0.958	0.953	0.951	0.953	0.956	0.961	0.958
PM	0.961	0.961	0.958	0.953	0.951	0.948	0.948	0.945	0.945
PN	0.958	0.956	0.953	0.961	0.948	0.948	0.945	0.945	0.280

TABLE 12: The test result of the above model,  $S_{TF}$ .

Feature	0.1 order	0.2 order	0.3 order	0.4 order	0.5 order	0.6 order	0.7 order	0.8 order	0.9 order
Max	0.971	0.969	0.971	0.971	0.971	0.969	0.969	0.965	0.967
MA	0.971	0.969	0.971	0.971	0.971	0.969	0.969	0.965	0.967
Range	0.973	0.973	0.971	0.971	0.971	0.969	0.971	0.965	0.967
Std.	0.969	0.973	0.971	0.971	0.971	0.971	0.969	0.967	0.965
Var.	0.971	0.971	0.971	0.971	0.969	0.969	0.969	0.967	0.967
RMS	0.963	0.963	0.965	0.965	0.965	0.965	0.965	0.965	0.965
IQR	0.959	0.967	0.967	0.961	0.947	0.941	0.935	0.939	0.939
Mean	0.961	0.959	0.959	0.953	0.957	0.957	0.959	0.965	0.963
PM	0.959	0.959	0.955	0.943	0.937	0.943	0.951	0.945	0.953
PN	0.929	0.933	0.929	0.933	0.931	0.929	0.931	0.933	0.937
Max	0.931	0.931	0.933	0.931	0.931	0.931	0.931	0.933	0.268
MA	0.931	0.933	0.933	0.931	0.931	0.931	0.931	0.935	0.935
Range	0.933	0.933	0.931	0.931	0.929	0.931	0.931	0.931	0.953
Std.	0.929	0.931	0.931	0.931	0.931	0.931	0.931	0.006	0.0260
Var.	0.929	0.929	0.931	0.931	0.931	0.931	0.353	0.942	0.949
RMS	0.925	0.929	0.929	0.929	0.931	0.933	0.877	0.933	0.846
IQR	0.929	0.929	0.929	0.931	0.937	0.927	0.927	0.935	0.933
Mean	0.894	0.925	0.929	0.929	0.931	0.931	0.881	0.939	0.877
PM	0.931	0.929	0.929	0.927	0.931	0.933	0.933	0.933	0.933
PN	0.931	0.929	0.933	0.931	0.931	0.933	0.933	0.933	0.362

group number is inside the brackets. The feature combinations are represented by numbers instead of feature names, and the order is consistent with formula (12), 1–10 indicates the feature calculated from acceleration, and 11–20 indicates the feature calculated from angular velocity.

According to general laws, falls usually occur after the state of regular motion. Therefore, the first layer of the classifier is the classification of regular motion, which divides the current motion into walking or jogging and then monitors the occurrence of falls in real time and finally divides the types of falls. Table 15 shows the optimal classification effect of the preimpact fall detection for walking and jogging and its transformation order and feature group

used. Figure 10 shows the distribution of the features for classification of walking and jogging with the three falling states. According to the results, the preimpact fall detection parameters of two conventional states are classified.

After separating the fallen state from the regular motion state, the preliminary preimpact fall detection has been completed. The next step is to classify the three falling states. Using the same method, find the best classification order and feature group that separates one of the three falling states. Finally, separate the remaining two falling states according to Table 14. At this point, the preimpact fall detection and the classification of falling states are completed. Table 16 shows the optimal classification effect of a falling state

TABLE 13: The product of training result  $S_{TF}$  and testing result  $S_{TF}$ .

Feature	0.1 order	0.2 order	0.3 order	0.4 order	0.5 order	0.6 order	0.7 order	0.8 order	0.9 order
Max	0.949	0.944	0.946	0.946	0.944	0.942	0.942	0.935	0.940
MA	0.949	0.944	0.946	0.946	0.944	0.942	0.942	0.935	0.940
Range	0.951	0.951	0.946	0.946	0.944	0.942	0.943	0.935	0.940
Std.	0.947	0.951	0.946	0.946	0.946	0.944	0.942	0.937	0.938
Var.	0.946	0.946	0.946	0.946	0.942	0.942	0.939	0.937	0.937
RMS	0.936	0.936	0.938	0.938	0.935	0.935	0.935	0.935	0.935
IQR	0.932	0.940	0.940	0.934	0.918	0.909	0.901	0.905	0.905
Mean	0.931	0.932	0.929	0.923	0.927	0.927	0.929	0.935	0.933
PM	0.932	0.929	0.923	0.914	0.908	0.914	0.919	0.913	0.921
PN	0.893	0.894	0.886	0.889	0.888	0.881	0.883	0.882	0.886
Max	0.900	0.895	0.897	0.892	0.888	0.883	0.885	0.880	0.194
Std.	0.893	0.895	0.888	0.888	0.888	0.888	0.883	0	0
Var.	0.890	0.890	0.888	0.888	0.888	0.883	0.305	0.856	0.872
RMS	0.896	0.893	0.888	0.886	0.885	0.885	0.836	0.904	0.806
IQR	0.895	0.893	0.893	0.892	0.893	0.884	0.879	0.884	0.882
Mean	0.874	0.891	0.890	0.886	0.885	0.888	0.842	0.903	0.841
PM	0.895	0.893	0.890	0.884	0.885	0.885	0.885	0.882	0.882
PN	0.892	0.888	0.889	0.895	0.883	0.885	0.882	0.882	0.101
Max	0.949	0.944	0.946	0.946	0.944	0.942	0.942	0.935	0.940
MA	0.949	0.944	0.946	0.946	0.944	0.942	0.942	0.935	0.940

TABLE 14: The optimal classification effect between every two motions and the transformation order and the feature group used.

Motion	Walking	Jogging	Fainting	Tripping	Slipping
Walking		0.94-0.98 0.3-(5, 18)	0.865-0.825 0.5-(5, 11, 18)	1-0.98 0.5-(5, 6)	0.98-1 0.1-(5, 7)
Jogging	0.941-0.98 0.5-(5, 14)		0.865-0.902 0.1-(15, 7, 11)	0.9-0.884 0.7-(15, 7, 17)	0.902-0.96 0.5-(15, 13, 9)
Fainting	0.865-0.864 0.5-(5, 7, 20)	0.865-0.902 0.2-(15, 7, 10)		0.718-0.791 0.1-(10, 11, 18)	0.81-0.845 0.9-(10, 7, 14)
Tripping	1-1 0.4-(5, 8)	0.9-0.92 0.1-(15, 11, 20)	0.739-0.768 0.4-(10, 11, 19)		0.846-0.81 0.9-(5, 11, 19)
Slipping	1-0.98 0.2-(5, 6)	0.921-0.98 0.5-(15, 8)	0.826-0.843 0.9-(10, 7, 15)	0.846-0.810 0.9-(5, 11, 19)	

TABLE 15: The optimal classification effect of the preimpact fall detection for walking and jogging and its transformation order and feature group used.

Motion states	Optimal order	Feature group	Training set	Testing set
Walking	0.7	6,7,17	0.921	0.946
Jogging	0.1	15,3,13	0.868	0.902

separated from the three falling states, its transformation order, and the feature group used. The classification that prioritizes the fainting state for separation works best, while the classification parameters of separating slipping and tripping states can be found in Table 14, which are 0.9 order and No. 5, 11, and 19 features. According to the above, a multilayer dichotomy classifier is constructed, as shown in Figure 11.

In Figure 11, the training set accuracy of classifier a-g is, respectively, 99%, 95%, 94.5%, 87.3%, 87.3%, 85%, and 85%. The accuracy of the testing set is 95%, 94%, 91.5%, 92%, 92%, 88%, and 88%, respectively.

## 4. Experiment and Analysis

*4.1. Experimental Design.* In the experiment of this section, we use the method described in Chapter II-A to obtain data, and the software is used to preprocess the collected data, specify order fractional Fourier transform, extract feature, and classify motion state.

In the experiment, we recorded 50 fall events for each tester. Since walking and jogging are indistinguishable in static pictures, the following only shows the cases of fainting, tripping, and slipping in the walking state in the experiment, as shown in Figure 12.

*4.2. Experimental Result.* The classification result of the collected motion data is shown in Figure 13. Among them, Figures 13(a), 13(c), and 13(e) are the preimpact fall detection effects of fainting, tripping, and slipping during walking; Figures 13(b), 13(d), and 13(f) are the preimpact fall detection effect of fainting, tripping, and slipping during jogging. Walking, jogging, fainting, tripping, and slipping are numbered 1–5 in sequence. It can be seen that during

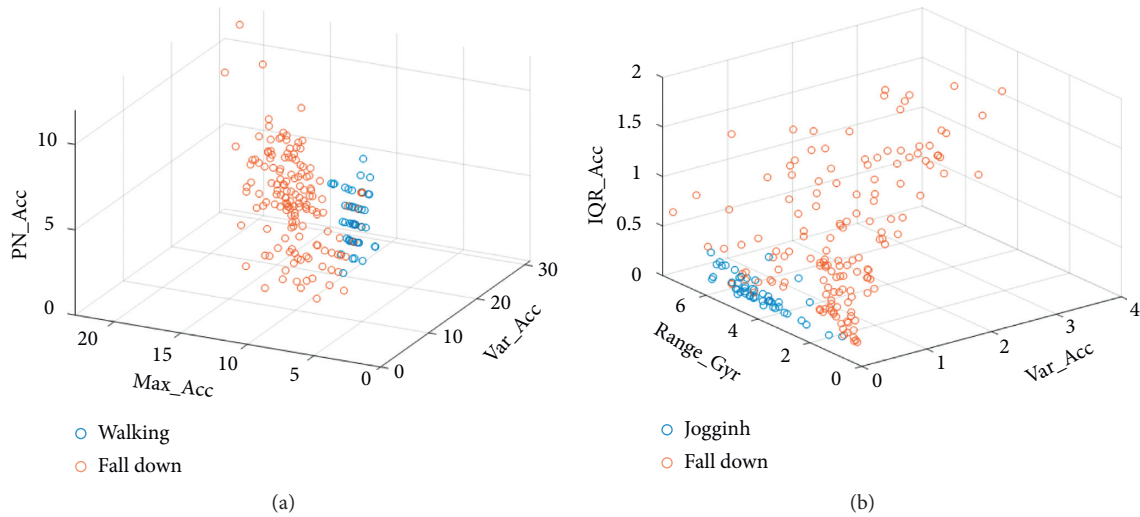


FIGURE 10: (a) The feature distribution of walking state with three falling states. (b) The feature distribution of jogging state with three falling states.

TABLE 16: The optimal classification effect of a falling state separated from the three falling states and its transformation order and the feature group used.

Separated state	Transformation order	Feature group	Training set $S_{TF}$	Separated state $S_{TF}$
Fainting	0.1	10,9,5	0.78	0.76
Tripping	0.2	10,11,4	0.639	0.691
Slipping	0.9	5,19,14	0.619	0.605

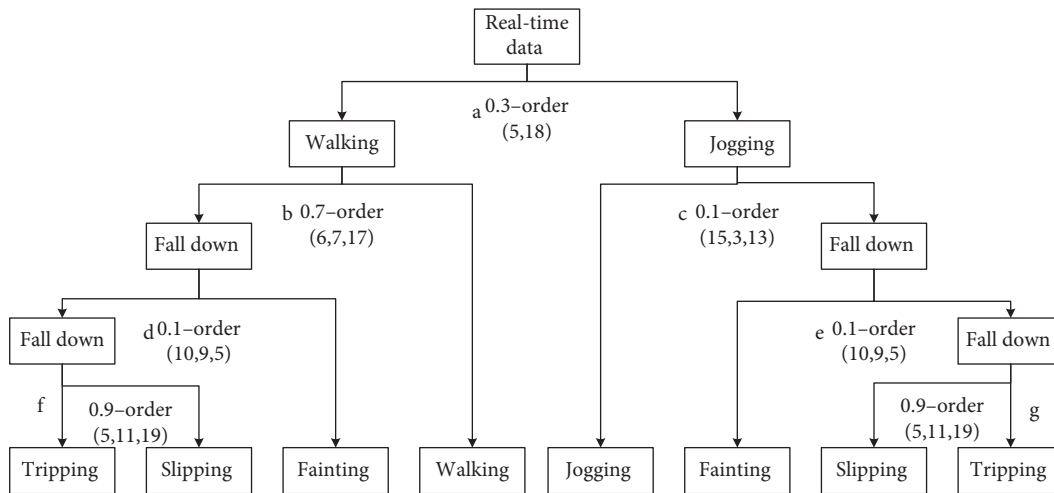


FIGURE 11: Structure diagram of multilayer dichotomy classifier.

walking, the occurrence of falls can be better warned, the average delay time is 89 ms, and the average lead time is 456 ms; the inertial data caused by jogging fluctuate greatly, so it is easy to be confused with falling. Therefore, the preimpact fall detection during jogging is generally inferior to walking, with an average delay time of 322 ms and an average lead time of 240 ms.

The algorithm proposed in this paper is compared with the threshold method and machine learning method, as shown in Table 17.

### 5. Discussion

A key goal of this article is to achieve efficient classification and early warning of falls for the elderly. The previous assumption is that it has higher classification performance and accuracy. From the experimental results, we can see that compared with other methods such as threshold method, this article proposes the algorithm which had enhanced the classification ability of falls and improves the accuracy and time of early warning. This will provide useful help for future fall detection research.

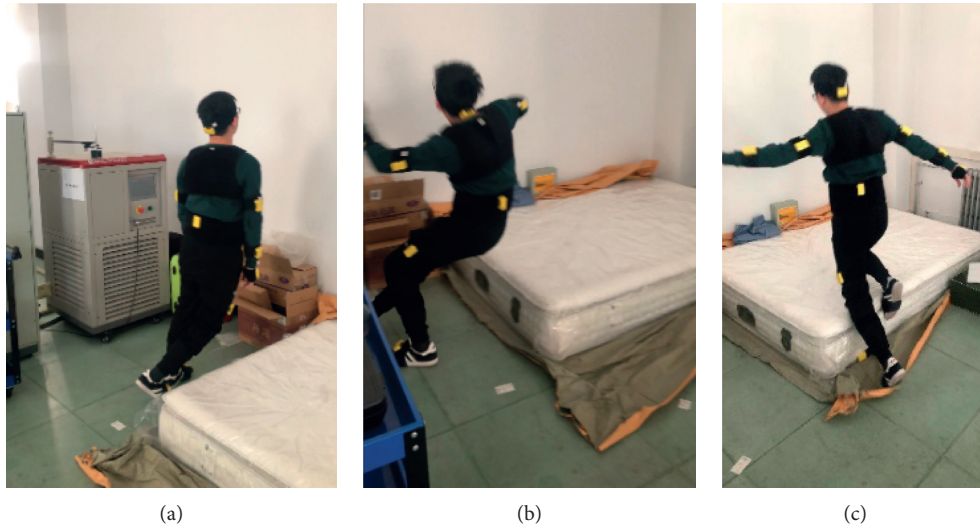


FIGURE 12: (a) Fainting, (b) slipping, and (c) tripping, while in the state of walking.

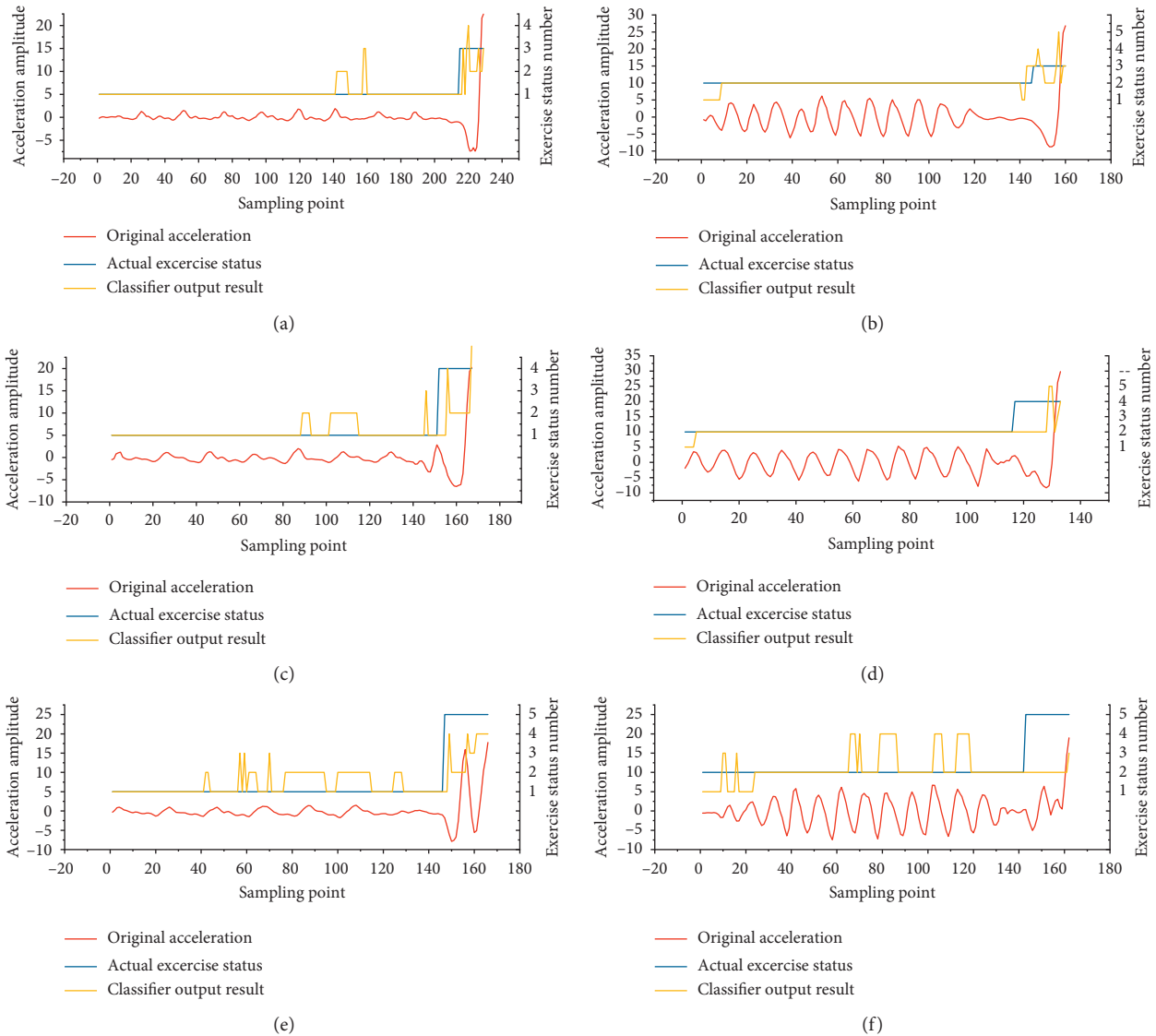


FIGURE 13: Real-time motion state classification renderings. (a) The preimpact fall detection effects of fainting while in walking state. (b) The preimpact fall detection effect of fainting while in jogging state. (c) The preimpact fall detection effects of tripping while in walking state. (d) The preimpact fall detection effect of tripping while in jogging state. (e) The preimpact fall detection effects of slipping while in walking state. (f) The preimpact fall detection effect of slipping while in jogging state.

TABLE 17: Comparison of the preimpact fall detection effect between fractional domain + multilayer dichotomy and other algorithms.

Classification algorithm	Average lead time (walk/run) (ms)	Number of used features	Average accuracy (walk/run)	Fall segmentation	Unable to detect the state of fall
Threshold method (single person, independent threshold)	431/184	1		Yes	No
Threshold method (single person, unified threshold)	233/156	1		No	No
Threshold method (multiple people, unified threshold)	270/244	1		No	Yes
KNN	427/311	20	0.85/0.61	Yes	No
Bagged trees	428/333	20	0.77/0.85	Yes	No
Fractional domain + multilayer dichotomy	456/240	2-3	0.9/0.87	Yes	No

However, the physical fitness of the experimenters in this study has limitations. It can be seen from the previous experiments that it has some influence on the fall classification results. In the follow-up, we consider collecting more fall data of the elderly to further improve the training of the classification model.

## 6. Conclusion

According to Section IV-B, the method proposed in this paper can better adapt to the movement feature of different people and distinguish different falling states compared to the threshold method. Compared with most machine learning methods, fewer feature values are used, which improves accuracy and lead time. Meanwhile, the classification boundary function of each subclassifier using SVM is a quadratic polynomial, which is more suitable for running on embedded systems.

The experiment results show that compared with other methods, the method proposed in this paper has a better preimpact fall detection effect during walking. It can ensure better preimpact fall detection effect and classification accuracy using fewer features. The average lead time reaches 456 ms, and the average delay time is 89 ms. For the pre-impact fall detection during jogging, the average lead time of the method proposed in this paper is 240 ms, and the average delay time is 322 ms.

## Data Availability

The data were collected by the project team using public equipment.

## Conflicts of Interest

The authors declare that they have no conflicts of interest.

## Authors' Contributions

Ning Liu, Dedi Zhang, Zhong Su, and Tianrun Wang contributed equally to this work.

## Acknowledgments

This work was supported by the National Natural Science Foundation of China (61801032), National Key Research and

Development Project (2020YFC1511702), the Beijing Key Laboratory of High Dynamic Navigation Technology, Laboratory of Modern Measurement and Control Technology (Beijing Information Science & Technology University, Ministry of Education), and Qin Xin Talents Cultivation Program, Beijing Information Science & Technology University.

## References

- [1] World Health Organization: <https://www.who.int/zh/news-room/fact-sheets/detail/falls>, Falls[EB/OL].
- [2] S. Loue and M. Sajatovic, *Encyclopedia of Immigrant Health [M]*, pp. 671-672, Springer-Verlag New York Incorporation, New York, NY, 2012.
- [3] P. PIERLEONI, A. BELLI, L. PALMA, M. Pellegrini, L. Pernini, and S. Valenti, "A high reliability wearable device for elderly fall detection," *IEEE Sensors Journal*, vol. 15, no. 8, pp. 4544-4553, 2015.
- [4] S. Tao, M. Kudo, and H. Nonaka, "Privacy-preserved behavior analysis and fall detection by an infrared ceiling sensor Network," *Sensors*, vol. 12, no. 12, pp. 16920-16936, 2012.
- [5] Y. Su, D. Liu, and Y. Wu, "A multi-sensor based pre-impact fall detection system with a hierarchical classifier," in *Proceedings of the 2016 9th International Congress on Image and Signal Processing, BioMedical Engineering and Informatics (CISP-BMEI)*, pp. 1727-1731, IEEE, Datong, China, October 2016.
- [6] G. Zhao, Z. Mei, D. Liang et al., "Exploration and implementation of a pre-impact fall recognition method based on an inertial body sensor Network," *Sensors*, vol. 12, no. 11, pp. 15338-15355, 2012.
- [7] J. Liu and T. E. Lockhart, "Development and evaluation of a prior-to-impact fall event detection algorithm," *IEEE Transactions on Biomedical Engineering*, vol. 61, no. 7, pp. 2135-2140, 2014.
- [8] J. K. Lee, S. N. Robinovitch, and E. J. Park, "Inertial sensing-based pre-impact detection of falls involving near-fall scenarios [J]," *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, vol. 23, no. 2, pp. 258-266, 2014.
- [9] M. Yao, Q. Zhang, M. Li et al., "A wearable pre-impact fall early warning and protection system based on MEMS inertial sensor and GPRS communication," in *Proceedings of the 2015 IEEE 12th International Conference on Wearable and Implantable Body Sensor Networks (BSN)*, pp. 1-6, IEEE, Cambridge, Massachusetts, USA, June 2015.
- [10] S. Ahn, D. Choi, J. Kim et al., "Optimization of a pre-impact fall detection algorithm and development of hip protection



- airbag system,” *Sensors and Materials*, vol. 30, no. 8, pp. 1743–1752, 2018.
- [11] Z. Zhong, F. Chen, Q. Zhai et al., “A real-time pre-impact fall detection and protection system,” in *Proceedings of the 2018 IEEE ASME International Conference on Advanced Intelligent Mechatronics (AIM)*, pp. 1039–1044, IEEE, Auckland, New Zealand, July 2018.
- [12] G. Wu and S. Xue, “Automatic Fall Detection Based on Kinematic Characteristics during the Pre-impact Phase of falls,” in *Proceedings of the 6th World Congress of Biomechanics (WCB 2010)*, pp. 342–345, Springer, Berlin, Heidelberg, August 2010.
- [13] D. Martelli, F. Artoni, V. Monaco et al., “Pre-impact fall detection: optimal sensor positioning based on a machine learning paradigm,” *PLoS One*, vol. 9, no. 3, 2014.
- [14] S. Liang, T. Chu, D. Lin et al., “Pre-impact alarm system for fall detection using MEMS sensors and HMM-based SVM classifier,” in *Proceedings of 2018 40th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*, pp. 4401–4405, IEEE, Honolulu, Hawaii, July 2018.
- [15] F. De Cillis, F. De Simio, F. Guidoy et al., “Fall-detection solution for mobile platforms using accelerometer and gyroscope data,” in *Proceedings of the International Conference on Engineering in Medicine and Biology Society*, pp. 3727–3730, IEEE, Milan, Italy, August 2015.
- [16] S. H. Cheng, “An intelligent fall detection system using triaxial accelerometer integrated by active RFID,” in *Proceedings of International Conference on Machine Learning and Cybernetics*, pp. 517–522, IEEE, Lanzhou, China, July 2014.
- [17] J. Cheng, X. Chen, and M. Shen, “A framework for daily activity monitoring and fall detection based on surface electromyography and accelerometer signals,” *IEEE Journal of Biomedical and Health Informatics*, vol. 17, no. 1, pp. 38–45, 2013.
- [18] S. Liu, Z. Zeng, Y. D. Zhang et al., “Automatic human fall detection in fractional fourier domain for assisted living,” in *Proceedings of the 2016 IEEE international conference on acoustics, speech and signal processing (ICASSP)*, pp. 799–803, IEEE, Shanghai, China, March 2016.
- [19] A. Bultheel and H. E. Martínez Sulbaran, “Computation of the fractional fourier transform,” *Applied and Computational Harmonic Analysis*, vol. 16, no. 3, pp. 182–202, 2004.
- [20] C. Li, Z. Su, Q. Li, and H. Zhao, “An indoor positioning error correction method of pedestrian multi-motions recognized by hybrid-orders fraction domain transformation,” *IEEE Access*, vol. 7, no. 7, pp. 11360–11377, 2019.
- [21] N. Otanasap and P. Boonbrahm, “Pre-impact fall detection system using dynamic threshold and 3D bounding box[A]. Eighth International Conference on Graphic and Image Processing (ICGIP 2016),” *International Society for Optics and Photonics*, vol. 10225, p. 102250D, 2017.
- [22] L. Ren, W. Shi, Z. Yu et al., “ALARM: a novel fall detection algorithm based on personalized threshold,” in *Proceedings of International Conference on E-Health Networking, Application & Services*, pp. 410–415, IEEE, Boston, MA, USA, October 2015.
- [23] O. AZIZ, C. M. RUSSELL, E. J. PARK et al., “The effect of window size and lead time on pre-impact fall detection accuracy using support vector machine analysis of waist mounted inertial sensor data,” in *Proceedings of International Conference on Engineering in Medicine and Biology Society*, pp. 30–33, IEEE, Chicago, Illinois, August 2014.
- [24] S. Shan and T. Yuan, “A wearable pre-impact fall detector using feature selection and support vector machine,” in *Proceedings of IEEE 10th International Conference On Signal Processing Proceedings. IEEE*, pp. 1686–1689, Beijing, China, October 2010.
- [25] C. C. Zhou, C. L. Tu, Y. Gao et al., “A low-power, wireless, wrist-worn device for long time heart rate monitoring and fall detection,” in *Proceedings of International Conference on Orange Technologies*, pp. 33–36, IEEE, Xi’an, China, September 2014.
- [26] B. T. Nukala, N. Shibuya, A. I. Rodriguez et al., “A real-time robust fall detection system using a wireless gait analysis sensor and an Artificial Neural Network,” in *Proceedings of International Conference on Healthcare Innovation*, pp. 219–222, IEEE, Seattle, Washington, 2014.
- [27] H. Jian and H. Chen, “A portable fall detection and alerting system based on k-NN algorithm and remote medicine,” *China Communications*, vol. 12, no. 4, pp. 23–31, 2015.
- [28] A. Lisowska, G. Wheeler, V. Inza et al., “An evaluation of supervised, novelty-based and hybrid approaches to fall detection using silmee accelerometer data,” in *Proceedings of International Conference on Computer Vision Workshops*, pp. 10–16, IEEE, Santiago, Chile, December 2015.
- [29] J. Wang, R. Chen, X. Sun et al., “Generative models for automatic recognition of human daily activities from a single triaxial accelerometer,” in *Proceedings of International Joint Conference on Neural Networks*, pp. 1–6, IEEE, Brisbane, Australia, June 2012.
- [30] C. Li, M. Lin, L. T. Yang, and C. Ding, “Integrating the enriched feature with machine learning algorithms for human movement and fall detection,” *The Journal of Supercomputing*, vol. 67, no. 3, pp. 854–865, 2014.