A Review of Step Detection Algorithms Based on MEMS Accelerometers (Postprint)

Authors: Xie Guangqiang, Huang Xianglong, Li Yang, Xu Feng

Date: 2018-05-20T00:00:00+00:00

Abstract

With the rapid development of smart mobile terminals, smart wearable devices, and pedestrian positioning and navigation systems based on inertial sensors, extensive research on step detection algorithms using MEMS accelerometers has been conducted to address the step counting requirements of these devices and systems, yielding excellent results. This paper first examines the existing technical methods in this field, delineating its development status, principal research focuses, methodological categories, and application contexts. It then provides a comprehensive review of the current research landscape, elaborating on various step detection algorithms and analyzing their respective advantages and disadvantages from the perspectives of data preprocessing methods and step validation approaches. Subsequently, an in-depth exploration of the key research points in this domain is presented, with an analysis and synthesis of the technical methods in step detection algorithms pertinent to these research focuses. Finally, future development directions for the field are discussed and envisioned, offering insights for prospective research endeavors.

Full Text

Preamble

Review of Research on Step Detection Algorithm with MEMS-Based Acceleration Sensor

Xie Guangqiang, Huang Xianglong, Li Yang†, Xu Feng
(School of Computers, Guangdong University of Technology, Guangzhou 510000, China)

Abstract: With the rapid development of intelligent mobile terminals, smart wearable devices, and pedestrian navigation systems based on inertial sensors, a large number of research efforts have been carried out on step detection algorithms using MEMS-based acceleration sensors to meet the step-counting demands of these devices and systems, achieving excellent results. This paper reviews the existing technical methods in this field. First, it elaborates on the development of the field, identifying the main research points, method categories, and application scenarios. Then, it surveys the current research status, describing and analyzing various step detection algorithms from two perspectives: data preprocessing methods and step validation methods, along with their respective advantages and disadvantages. Next, it conducts an in-depth discussion of the key research points in this field, analyzing and summarizing the technical methods related to these points. Finally, it discusses and prospects future development directions in this field to provide references for subsequent research.

Key Words: step detection; MEMS; acceleration sensor

0 Introduction

Traditional step detection methods utilized the inertia of metal pendulums or steel balls [1]. With the development of MEMS technology, MEMS-based acceleration sensor technology has matured, offering small physical size, light weight, and portability, while achieving accuracy and reliability in acceleration measurement that meets practical application requirements. Many smart mobile terminals such as smartphones and smart bracelets have already embedded MEMS acceleration sensors. As these devices demand step-counting functionality, step detection algorithms based on MEMS acceleration sensors have received increasingly widespread attention.

MEMS acceleration sensor-based step detection involves using predetermined algorithms or methods to calculate and identify steps from data collected by the sensor during human walking, thereby deriving the number of steps taken. Therefore, the accuracy of step detection algorithms—the error between the step count calculated by the algorithm and the actual step count—is the primary factor in judging algorithm quality. As research in this field continues to deepen, issues beyond accuracy, such as algorithm adaptability and real-time performance, have become prominent. For example, enabling step detection algorithms to adapt to different sensor positions on the human body and various walking patterns ultimately aims to improve accuracy across multiple scenarios. In some application scenarios, step counting results need to be generated within a short time. Thus, the current research points in this field mainly focus on accuracy, adaptability, and real-time performance.

In 2002, Ladetto et al. [2] proposed in their designed pedestrian navigation system that acceleration sensors could be used to detect pedestrian steps and calculate walking distance by combining step length. In 2007, Jang et al. [3] implemented a highly robust step detection algorithm based on a two-axis MEMS acceleration sensor. In recent years, step detection algorithm research has developed rapidly, with many domestic and foreign scholars contributing to this field and numerous methods emerging. These mainly include step cycle recognition methods [1], autocorrelation analysis methods [4-7], peak detection methods [8-18], fuzzy logic methods [19], zero-crossing detection methods [20,21], finite state machine methods [22-24], zero-velocity update methods [25-27], and dynamic time warping methods [1,8]. Research in this field strongly promotes studies in smartphone step-counting applications [28,29], smart pedometer design [30-32], sports health monitoring applications [33,34], and pedestrian localization and navigation technology based on PDR [35,36]. Meanwhile, with the development of smartphones and mobile intelligent terminal devices, applications in this field have also advanced. Many manufacturers, such as Jawbone [37] and Apple [38], have developed applications related to step detection for the MEMS acceleration sensors inside their devices. By detecting and analyzing users' walking steps and combining them with a series of health indicators, they provide users with health guidance.

1 Research Status of Step Detection Algorithms

As described in the introduction, step detection algorithms employ appropriate methods to calculate and identify steps from data collected by acceleration sensors during human walking. The process consists of two stages: the data preprocessing stage and the step detection and verification stage. This section analyzes the research status of step detection algorithms from these two stages, combined with an analysis of the advantages and disadvantages of each algorithm.

1.1 Data Preprocessing Methods

Preprocessing data collected by MEMS acceleration sensors to restore data regularity is a crucial component of step detection algorithms. Taking the acceleration sensor in a mobile phone as an example, during human walking, the way a person holds the phone is arbitrary, which may cause the collected data to exhibit irregular variations—commonly referred to as noisy data. The purpose of data preprocessing is to minimize noise in the raw data to serve the subsequent step detection algorithm' s calculation and identification of steps.

1.1.1 Calculation of Synthetic Acceleration Currently, 3-axis MEMS acceleration sensors are widely used to collect human walking data, with the three axes being X, Y, and Z—representing a three-dimensional data representation method. The synthetic acceleration can be calculated using the following equation:

$$A_c(t) = \sqrt{a_x(t)^2 + a_y(t)^2 + a_z(t)^2}$$

where $a_x(t)$, $a_y(t)$, and $a_z(t)$ represent the sampled values of the X, Y, and

Z axes at time t, respectively, and $A_c(t)$ is the synthetic acceleration, whose physical meaning represents the instantaneous acceleration magnitude of the object measured by the acceleration sensor at time t.

References [6,7,9,16–18,24,30,44] adopted this method for preprocessing acceleration data. Research shows that this method is simple to compute and consumes fewer resources. The arbitrariness of how humans carry acceleration sensors may cause the three-axis acceleration direction to be inconsistent with the direction of human motion. Calculating synthetic acceleration can partially eliminate noise caused by this factor [6].

Tang et al. [1], when studying an adaptive step detection algorithm, utilized gravity to calculate vertical and horizontal acceleration based on synthetic acceleration. Research shows that this method further eliminates noise in raw data caused by the sensor' s carrying method and human walking habits. The methods for calculating vertical and horizontal acceleration are as follows:

$$g = \sqrt{g_x^2 + g_y^2 + g_z^2}$$

$$a = \sqrt{a_x^2 + a_y^2 + a_z^2}$$

$$a_v = \frac{a_x g_x + a_y g_y + a_z g_z}{g}$$

$$a_h = \sqrt{a^2 - a_v^2}$$

In equation (2), g_x, g_y, and g_z represent the gravity components in the X, Y, and Z directions, respectively, and g represents the complete gravity. In equation (3), a_x, a_y, and a_z represent the accelerations in the X, Y, and Z directions, respectively, and a represents the complete acceleration. In equation (4), a_v represents the calculated vertical acceleration, and in equation (5), a_h represents the calculated horizontal acceleration.

Pan et al. [4] proposed an acceleration data preprocessing method based on coordinate rotation when studying step-counting algorithms on smartphones. The characteristic of this method is that it processes the data of the three axes separately, achieving good results for single-axis data processing, but with higher computational cost compared to calculating synthetic acceleration.

Assume a point $P(p_x, p_y, p_z)$ on a three-dimensional plane, with the three-dimensional coordinate origin located on this plane. Let the distance from this point to the coordinate origin be L, and the angle between the line connecting this point to the origin and the Z-axis be α. Now, rotate this point along the Y-axis by an angle α to become $P'(p'_x, p'_y, p'_z)$. According to trigonometric definitions, the following relationships exist:

$$\sin\alpha = \frac{p_y}{L}, \quad \cos\alpha = \frac{p'_y}{L}$$

$$\frac{x\sin(\alpha+\beta)}{L} = \frac{z\cos(\alpha+\beta)}{L}$$

Based on equation (6) and the trigonometric sum law, the following mathematical relationship exists:

$$\sin(\alpha+\beta) = \sin\alpha\cos\beta + \cos\alpha\sin\beta$$

$$\cos(\alpha+\beta) = \cos\alpha\cos\beta - \sin\alpha\sin\beta$$

After eliminating L in equation (7), the relationship between p'_x, p'_y, p'_z and p_x, p_y, p_z can be derived as follows:

$$p'_x = p_x\cos\beta + p_z\sin\beta$$

$$p'_y = p_y$$

$$p'_z = -p_x\sin\beta + p_z\cos\beta$$

Let the angle between the line connecting point P to the origin and the Y-axis be β. Then rotate point P' along the X-axis by an angle β to become $P''(p''_x, p''_y, p''_z)$. Similarly, based on the above trigonometric relationships, the relationship between point P'' and P' can be derived as follows:

$$p''_x = p'_x$$

$$p''_y = p'_y\cos\beta - p'_z\sin\beta$$

$$p''_z = p'_y\sin\beta + p'_z\cos\beta$$

By combining equations (8) and (9), the rotation angles α and β when point P transforms to point P'' can be calculated. Extending to the plane where this point lies, α and β also represent the rotation angles of the plane around the Y-axis and X-axis, respectively, approximately substituting the deviation degree of the sensor relative to the horizontal plane at this time, thereby calculating the horizontal component value of the acceleration at this moment and eliminating noise generated when the sensor plane is not horizontal.

1.1.3 Filter Algorithms The use of filter algorithms is also common in the data preprocessing stage. References [6,12,15,18,24] adopted the Kalman filter algorithm; references [16,20,29,43] adopted the mean filtering method; references [5,9,40] adopted the Butterworth filter; and reference [36] adopted Fourier transform filtering, among others.

Tran et al. [15], based on results calculated using the discrete Kalman filter algorithm, proposed a new filtering method according to the characteristic that acceleration data for each step first increases and then decreases. This method can identify and mark important acceleration changes in the acceleration data for step verification. The proposed filtering method is as follows:

$$a'(t) = \begin{cases} a(t) & \text{if } a(t) - a(t-1) > \gamma \\ a(t-1) & \text{otherwise} \end{cases}$$

where γ is the threshold, $a(t)$ is the acceleration signal after Kalman filtering, and $a'(t)$ is the acceleration signal after using the above filtering method. Through filtering algorithms, burrs in the data waveform can be effectively eliminated, making the collected sample data smoother or more regular, further eliminating noise in the data. Typically, to obtain good acceleration data, a combination of synthetic acceleration calculation and filtering algorithms is employed.

1.2 Step Detection and Verification Methods

Methods for recognizing steps in step detection research mainly include those described in the introduction section. These methods are briefly introduced below.

1.2.1 Autocorrelation Analysis Method The autocorrelation analysis method employs the theoretical foundation of correlation coefficients, whose mathematical formulation is described as follows:

$$\rho(m,t) = \frac{\sum_{k=0}^{t-1}[a(k) - \mu(m,t)][a(k+m) - \mu(m,t)]}{\sigma(m,t) \cdot \sigma(m,t)}$$

where:

$$\mu(m,t) = \frac{1}{t}\sum_{k=0}^{t-1} a(k)$$

$$\sigma(m,t) = \sqrt{\frac{1}{t}\sum_{k=0}^{t-1}[a(k) - \mu(m,t)]^2}$$

$\mu(m,t)$ represents the mean of the acceleration sequence, $\sigma(m,t)$ represents the standard deviation, m is the sampling period, and $\rho(m,t)$ is the correlation coefficient, which calculates the correlation between acceleration sequences within two periods.

In Pan et al.' s [4] step detection algorithm using autocorrelation analysis, for the Y-axis and Z-axis data of the acceleration sensor, a starting point detection method was adopted. During step verification, verification was not performed on a single-step basis but rather by calculating correlation coefficients between acceleration sequences of 2 or 4 steps, achieving good step detection results. Soaz et al. [5] proposed a method for parameterizing steps to construct a walking data template, finally using autocorrelation analysis for template matching to exclude step counting results generated by non-real walking data. Chen et al. [6] implemented a pedometer function on a mobile phone based on autocorrelation analysis, adopting a windowing calculation method. Within the windowed range, correlation coefficients were calculated with a gradient until a correlation coefficient value not less than a predetermined threshold was calculated, at which point effective step recognition was performed.

Rai et al. [7] implemented a step-counting algorithm based on autocorrelation in an indoor positioning system. By setting a threshold for the calculated correlation coefficient, step detection was performed. The algorithm effectively improved the accuracy and reliability of step detection based on autocorrelation analysis.

1.2.2 Peak Detection Method Jang et al. [3] proposed that the horizontal and vertical acceleration collected during human walking can be modeled as sinusoidal data waveforms. By detecting consecutive peaks and troughs, step recognition can be performed. Assuming the current algorithm window size is n, and the acceleration sequence within the window is $\{a(k), a(k+1), \ldots, a(k+n-1)\}$, then an acceleration value $a(k+i)$ satisfying the condition $a(k+i) > a(k+i-1)$ and $a(k+i) > a(k+i+1)$ can be represented as a peak. Due to jitter of the acceleration sensor during human walking, which introduces noise in the data, data filtering is generally also performed when using the peak detection method for step recognition.

Tang et al. [1] proposed an adaptive peak detection method for step detection that considers multi-dimensional parameters and uses adjustable optimal thresholds, improving the robustness of the peak detection method. Susi et al. [9] adopted an adaptive peak threshold step detection method, improving the adaptability of peak detection. Yoneyama et al. [10] extracted information such as correlation coefficients and the degree of positive and negative deviation of acceleration data from the acceleration data, proposing a new peak detection method based on this information to detect pedestrian steps. Lee et al. [11] proposed a peak detection method based on a morphological filter (MF), which uses MF to change the waveform of acceleration data, thereby making peak features more obvious and improving the accuracy of peak detection. Shin et al. [13], based on motion recognition, set different thresholds for the peak detection method according to different motion modes, improving the adaptive ability of peak detection for different motion modes. Chon et al. [14] proposed a step detection algorithm based on peak detection, which determines multiple local peaks in a series of acceleration sensor measurements, uses mean and standard deviation to eliminate possible false peaks, and generates candidate peaks, with each candidate peak representing one step taken by a person. Wang et al. [16] proposed a scoring mechanism based on the peak detection algorithm, which can enhance the peak detection method' s ability to handle false peaks. Chen et al. [17] proposed an adaptive peak detection step detection algorithm based on the characteristic that different motion states have different acceleration peak values, setting multiple fixed peak thresholds. This method improves the adaptability of the peak detection method for multiple motion modes.

1.2.3 Zero Velocity Update Method During human walking, there is a moment when the foot is perpendicular to the horizontal ground, at which time the human body' s motion velocity is close to zero. This characteristic can be used for step detection. In reference [25], the authors fixed a pressure sensor inside a shoe to detect the zero-velocity value during human walking. The mathematical formula for calculating the zero-velocity value is as follows:

$$v_{cntr} = \frac{1}{2\pi} \oint \frac{v}{r} \hat{r} \cdot d\hat{r}$$

where v is the velocity at a point on the isobar of the pressure sensor, r is the radius of that point, and $\hat{r}$ is the unit vector in the direction from the center of the isobar to that point. In equation (13), v_{cntr} is the velocity value at the center of the isobar that needs to be calculated.

Lo et al. [26] detected zero velocity during human walking by fixing an inertial sensor on the calf. Gu et al. [27] proposed a trajectory calculation method that improves the zero-velocity update method using building structure prior knowledge under a particle filter framework, improving the accuracy of personnel autonomous positioning.

1.2.4 Dynamic Time Warping Method The dynamic time warping algorithm is a relatively widely used method for comparing the similarity between two time sequences. The difference between this method and the autocorrelation analysis method is that the latter is a linear method, while the former is nonlinear [8]. Assuming there are sampled acceleration sequences $A(n,t) = \{a(1,t), a(2,t), \ldots, a(n,t)\}$ and $B(n,t) = \{b(1,t), b(2,t), \ldots, b(n,t)\}$, the dynamic time warping algorithm needs to calculate $dist[A,B]$ to determine the similarity between the two sequences, and this distance is generally Euclidean distance:

$$dist[A, B] = \sum_{k=1}^{n} [b(k, t) - a(k, t)]^2$$

where $a(k, t)$ and $b(k, t)$ are the sampled acceleration values of two steps, respectively, and $dist[A, B]$ is the distance between two acceleration values in the sequences. When calculating the similarity of horizontal acceleration sequences, the method is as follows:

Tang et al. [1] proposed an adaptive step detection algorithm based on dynamic time warping, which uses dynamically changing walking templates during matching verification, improving the adaptability of step detection algorithms based on dynamic time warping. Brajdic et al. [8] implemented a step detection algorithm based on dynamic time warping for comparison with other step detection algorithms.

1.2.5 Zero Crossing Detection Method During human walking, there are instances where the acceleration magnitude is zero, a characteristic that can be used for step detection.

Seo et al. [20] built a linear relationship between step count and the number of zero acceleration values based on this method. In practice, effective steps are detected by counting the number of times the acceleration value is zero. The authors also proposed an improved zero-crossing detection technique (AZDC) as follows:

$$AZDC_{range} = c \pm 3 \times \sigma_{rest}$$

where c is a constant and σ_{rest} is the standard deviation of the acceleration sequence at rest. Due to noise in the sampled acceleration data, by setting this boundary, only zero-crossing situations caused by accelerations outside this boundary are counted as effective zero crossings, effectively reducing the impact of noise on zero-crossing detection.

Shin et al. [21] implemented a simple zero-crossing step counting method by detecting the number of times the synthetic acceleration reaches zero during human walking.

1.2.6 Step Length Period Recognition Method Tang et al. [1] proposed the step length period recognition algorithm to address the limitation of peak detection methods that focus only on peaks and their poor adaptability. When calculating the similarity between two acceleration sequences, this method considers the similarity of vertical acceleration between the two sequences, the similarity between horizontal acceleration peaks, and the overlap degree of sensor orientation angles. The method for calculating vertical acceleration similarity is as follows:

$$S_{va} = \frac{d_h}{\max\{S_x\} - \min\{S_x\}}$$

where S_x and T_y represent vectors, and d_h represents the distance in horizontal acceleration.

The method differs significantly from dynamic time warping in that it no longer simply calculates Euclidean distance for distance computation. The authors reduced the algorithm' s time complexity by narrowing the search space to ensure execution efficiency.

1.2.7 Finite State Machine Method This method divides each step' s acceleration data, which first reaches a peak and then falls to a trough, into multiple states, and then sets some state transition conditions to combine these states into a finite state machine. By transitioning state conditions and judging the input 3-axis acceleration data, the current acceleration state is determined for step detection.

Alzantot et al. [22] divided the acceleration data of one step into six states based on acceleration magnitude characteristics. These states from S0 to S6 are: non-walking state, beginning to walk one step state, peak state, trough state, two noise accommodation states, and one-step termination state. Transitions between states are performed through set thresholds, with state transitions shown in [Figure 1: see original paper] and threshold definitions shown in [22].

Yim et al. [23] added a noise processing state based on the six states in reference [18], making the method' s noise handling more effective. Wang et al. [24] proposed an acceleration differential finite state machine step counting algorithm, which incorporates a data preprocessing stage to handle data noise, rather than adding noise shielding states in the state machine, simplifying the state transition process and improving step verification efficiency.

1.2.8 Fuzzy Logic Method Kammoun et al. [19] proposed a step detection method based on this approach. The method divides the gravity-removed acceleration signal into multiple segments, with each segment being a candidate step, and uses a fuzzy classifier to verify whether the candidate segment is an effective step. The gravity removal method is as follows:

$$a_{LPF}(t) = \text{low-pass filter of } a(t)$$

$$a_{BP}(t) = a(t) - a_{LPF}(t)$$

where $a_{LPF}(t)$ represents the acceleration sequence after low-pass filtering, and $a_{BP}(t)$ represents the acceleration sequence after gravity removal. Based on the selected candidate segments, four features f_1 to f_4 as shown in are calculated [19].

In , X_k represents an acceleration sequence segment, t_0 represents the sampling time corresponding to the first acceleration in the segment, where acceleration increases toward a positive threshold; t_1 represents the sampling time corresponding to the acceleration that crosses zero or equals zero during the decreasing process of acceleration in X_k; t_2 represents the sampling time corresponding to the last acceleration value in X_k, where the acceleration magnitude tends toward a positive threshold again; Amp_k represents the difference between the maximum and minimum acceleration in X_k, i.e., $Amp_k = \max\{X_k\} - \min\{X_k\}$; and σ_k^2 represents the variance of the X_k segment.

Based on the extracted four features, the membership degree between features is calculated, and with the vector:

$$D = [D_1, D_2, D_3, D_4]$$

where D_i represents the membership degree of feature f_i, step verification is performed using the following defuzzification rules:

Rule 1: IF $(T_{step} < T_{min})$ THEN $y_k = -1$

Rule 2: IF $(T_{min} \leq T_{step} \leq T_{max})$ AND $(Amp_k \geq \alpha)$ THEN $y_k = +1$

Rule 3: IF $(T_{min} \leq T_{step} \leq T_{max})$ AND $(\beta \leq d_k)$ THEN $y_k = -1$

Rule 4: IF $(T_{min} \leq T_{step} \leq T_{max})$ AND $(\beta > d_k)$ THEN $y_k = +1$

In the above rules, T_{min}, T_{max}, T_{step}, α, and β are preset thresholds. $y_k = +1$ indicates that vector X_k represents a step, $y_k = -1$ indicates that vector X_k cannot represent a step, and $y_k = 0$ indicates uncertainty about whether X_k represents a step.

1.3 Analysis of Advantages and Disadvantages of Step Detection and Verification Methods

As described in the introduction, the application fields of step detection algorithms are diverse. For different application scenarios, the data to be collected and used and the data preprocessing methods will differ. Additionally, sensor data itself has temporal characteristics, and even for the same person under the same conditions, different data will be collected. Currently, there is no standardized data sample available for research, which is a difficulty in the step detection algorithm research field and a direction the authors are currently working on. Therefore, in the absence of standardized reference indicators, the analysis of the advantages and disadvantages of various step detection algorithms can be conducted from the perspectives of algorithm adaptability, step detection accuracy in respective application scenarios, and real-time performance of step counting.

Based on the above elaboration of each method and the analysis of advantages and disadvantages in , the following conclusions can be drawn:

a) Step detection methods based on zero-crossing detection and peak detection have simple implementation and consume few resources. However, the accuracy of peak detection results depends on the selection of window size [6]. Additionally, reasonable threshold setting is crucial for peak detection methods [10]. Zero-crossing detection methods may have low accuracy due to detecting too many or too few zero-crossing points.

b) Step detection methods based on zero-velocity update utilize the natural phenomenon of instantaneous zero velocity when the leg is perpendicular to the ground during human walking, resulting in high accuracy. However, due to their dependence on fixed sensor positions, these methods have limited practicality.

c) Methods based on autocorrelation analysis, dynamic time warping, and step length period recognition utilize the characteristic that acceleration sequences of each cycle show similarity under the same motion, resulting in high accuracy and certain adaptability. However, they require relatively high computational resources, with algorithm time complexity reaching $O(n^2)$. Meanwhile, dynamic changes in step length period are also important factors affecting the accuracy of these methods [34].

d) The finite state machine method overcomes the disadvantage of peak detection methods that only focus on the relationship between peaks and thresholds. It sets peaks and troughs as states, paying more attention to changes in acceleration data. However, due to its fixed state transition thresholds and lack of consideration for periodic characteristics, this method has poor adaptability. Generally, it has good processing capability for acceleration data with good sinusoidal patterns [35].

e) Step detection methods based on fuzzy logic have high accuracy due to using multiple features as basis. However, because the thresholds of this method are relatively fixed, its adaptability is poor.

2 Analysis of Key Research Points in Step Detection Algorithms

In addition to preprocessing acceleration data, analyzing the motion state and gait classification of acceleration data and eliminating abnormal step detection are also effective methods to improve the accuracy of step detection algorithms.

2.1 Accuracy of Step Detection Algorithms

Accuracy is the primary criterion for judging the quality of step detection algorithms. Currently, to ensure step detection accuracy, data preprocessing is a commonly used method, such as calculating synthetic acceleration, using filtering algorithms, and other processing methods as described above. The key issue to be addressed is to select and apply appropriate motion classification and recognition methods by combining step detection application scenarios, device operation resources, and real-time requirements. Taking the autocorrelation analysis method as an example, because it focuses too much on data similarity without caring whether the data is generated by normal human walking, detection results are easily disturbed. Therefore, performing walking detection or gait analysis on acceleration data before detection can largely ensure the accuracy and reliability of step detection results.

2.2 Adaptability of Step Detection Algorithms

With the deepening of step detection algorithm research, the adaptability of step detection algorithms has received attention from domestic and foreign scholars in recent years. In practical applications, human walking patterns and sensor carrying methods are variable and unpredictable. How to make algorithms better adapt to these conditions is an important research aspect.

Tang et al. [1] proposed an adaptive peak detection algorithm that uses multi-dimensional data such as vertical acceleration, horizontal acceleration, sensor rotation angle, step frequency, and step length as parameters for step detection. After each step is verified, the optimal threshold of parameters is adjusted, and the initialized optimal threshold is used as a basic basis to prevent thresholds that do not conform to reality during step detection. This method effectively improves the adaptability of peak detection-based step detection algorithms for multiple motion modes through multi-dimensional parameters and variable parameter optimal values.

Pan et al. [4] used a coordinate rotation-based method for data preprocessing, largely eliminating data noise caused by the arbitrariness of how humans hold mobile phones by obtaining the horizontal component value of acceleration data. Experiments proved that this data preprocessing method improved the adaptability of autocorrelation analysis-based step detection algorithms for various acceleration sensor carrying methods.

Ryu et al. [10] designed an adaptive step detection algorithm that addresses the disadvantage of using a single threshold in previous step detection algorithms by designing a variable threshold algorithm. This method uses the magnitude of acceleration data changes as a factor, and the variable threshold is updated based on the magnitude of historical data, improving the adaptability of peak detection-based step detection algorithms.

Chen et al. [17] proposed an adaptive peak detection step detection algorithm that sets multiple fixed peak thresholds based on the characteristic that acceleration data peak ranges differ under different human motion modes. This method improves the adaptability of peak detection methods for multiple motion modes.

From the above research, it is found that the study of adaptability issues in step detection algorithms can be conducted from two aspects: improving and innovating data preprocessing methods and step detection and verification methods. Through reasonable data preprocessing, noise in acceleration data caused by sensor jitter, inconsistency between sensor orientation and walking direction, and diversity of walking patterns can be effectively reduced, fully restoring the regularity of acceleration data and improving algorithm adaptability to these factors. Improvements and innovations in step detection and verification methods, such as setting variable thresholds and adopting new mathematical models, can further improve the accuracy of step detection algorithms in multiple environments by overcoming the low adaptability disadvantages of existing methods.

2.3 Real-Time Performance of Step Detection Algorithms

In practical applications, the real-time performance of step detection algorithms is also important. Human walking frequency ranges from 1 to 2.5 Hz, meaning the walking period is approximately 0.4 to 1 s [6]. Therefore, step detection algorithms should theoretically produce step recognition results within about 1 s. Due to algorithm complexity and hardware resource limitations, the real-time performance requirements of step detection algorithms pose certain challenges. In some cases, to ensure step detection accuracy, some time cost is sacrificed.

Tang et al. [1] pointed out that their designed step length period recognition algorithm has lag in step detection because when verifying current acceleration data, it compares the data with data from subsequent steps. Only when the comparison result meets threshold conditions will step detection results for the current acceleration data be generated. This situation sacrifices time to ensure step detection accuracy.

Oshin et al. [39] designed an energy-efficient real-time step detection algorithm based on smartphones. This method does not perform filtering or denoising on acceleration data, thereby saving the time and space consumption of filtering and denoising methods. The method saves mobile phone energy through low sampling frequency and performs step detection by extracting 5 features from acceleration data and setting thresholds for the features. Experiments show that this method can produce step detection results within 2 s.

3 Conclusion

MEMS acceleration sensor-based step detection algorithms have a wide range of application scenarios. Research on step detection algorithms mainly includes two stages: data acquisition and processing, and step detection and verification algorithm design. Accuracy is the primary measure of algorithm quality, adaptability serves as a guarantee for enhanced accuracy, and real-time performance requirements are met as much as possible while ensuring high algorithm accuracy. Future research directions in this field focus on the following aspects:

a) **Improvement and innovation of step detection methods.** Many existing step detection algorithms still have many defects in accuracy, adaptability, and real-time performance, such as using fixed thresholds, depending on sensor carrying methods, being unsuitable for low-speed walking state step detection, and having time delays in detection results. In subsequent step detection method research, improvements can be made to the adaptability and real-time performance of step detection algorithms. On one hand, algorithms should meet requirements such as adapting to sensor carrying methods, walking postures, and dynamically adjusting optimal values. On the other hand, for real-time performance requirements, corresponding data preprocessing methods and step verification methods should be studied.

b) **Combination of step detection methods and motion detection methods.** Detecting and classifying a series of normal human activities or motions, such as running, walking, going up and down stairs, standing still, and jumping, and performing step detection on this basis can effectively filter out data generated by abnormal walking. At the same time, algorithm optimal values can be dynamically adjusted according to different motion detection results, further enhancing the adaptability of step detection algorithms and improving detection accuracy.

c) **Development of diversified applications based on step detection algorithms.** For example: a) Development of high-precision indoor positioning and navigation systems, including path tracking for indoor pedestrians and providing reliable navigation suggestions; b) Development of health applications, including health monitoring APPs based on smartphones and step-counting functions based on smart wearable devices. Meanwhile, combining massive step-counting data with some personal characteristic data (such as gender, height, age, occupation, etc.) collected by smart wearable devices or health monitoring APPs for statistics and analysis can further understand user behavior habits, analyze user health levels, predict personal characteristics of new users, etc., and develop corresponding applications based on analysis results.

References

[1] Tang Z, Guo Y, Chen X. Self-adaptive step counting on smartphones under unrestricted stepping modes [C]// Proc of IEEE Computer Software and Applications Conference. 2016: 788-797.

[2] Ladetto Q, Merminod B. In step with INS [J]. Gps World, 2002 (3): 749-751.

[3] Jang H J, Kim J W, Hwang D H. Robust step detection method for pedestrian navigation systems [J]. Electronics Letters, 2007, 43 (14): 749-751.

[4] Pan M S, Lin H W. A step counting algorithm for smartphone users: design and implementation [J]. IEEE Sensors Journal, 2015, 15 (4): 2296-2305.

[5] Soaz C, Diepold K. Step detection and parameterization for gait assessment

using a single waist-worn accelerometer [J]. IEEE Trans on Biomedical Engineering, 2016, 63 (5): 933-942.

[6] Chen G L, Zhang Y Z, Yang Z. A pedometer implementation method based on autocorrelation analysis of mobile phone sensors [J]. Chinese Journal of Inertial Technology, 2014, 22 (6).

[7] Rai A, Chintalapudi K K, Padmanabhan V N, et al. Zee: zero-effort crowdsourcing for indoor localization [C]// Proc of the 18th Annual International Conference on Mobile Computing and Networking. 2012: 293-304.

[8] Brajdic A, Harle R. Walk detection and step counting on unconstrained smartphones [C]// Proc of ACM International Joint Conference on Pervasive and Ubiquitous Computing. 2013: 225-234.

[9] Susi M, Renaudin V, Lachapelle G. Motion mode recognition and step detection algorithms for mobile phone users [J]. Sensors, 2013, 13 (2): 1539-1562.

[10] Yoneyama M, Kurihara Y, Watanabe K, et al. Accelerometry-based gait analysis and its application to Parkinson' s disease assessment, part 1: detection of stride event [J]. IEEE Trans on Neural Systems & Rehabilitation Engineering, 2014, 22 (3): 613-620.

[11] Lee H K, You J, Cho S P, et al. Computational methods to detect step events for normal and pathological gait evaluation using accelerometer [J]. Electronics Letters, 2010, 46 (17): 1185-1187.

[12] Hu W Y, Lu J L, Jiang S, et al. WiBEST: a hybrid personal indoor positioning system [J]. Implementation Science, 2013, 2 (1): 2149-2154.

[13] Shin B, Kim C, Kim J, et al. Motion recognition-based 3D pedestrian navigation system using smartphone [J]. IEEE Sensors Journal, 2016, 16 (18): 6977-6989.

[14] Chon J, Cha H. LifeMap: a smartphone-based context provider for location-based services [J]. IEEE Pervasive Computing, 2011, 10 (2): 58-67.

[15] Tran K, Le T, Dinh T. A high-accuracy step counting algorithm for iPhones using accelerometer [C]// Proc of IEEE International Symposium on Signal Processing and Information Technology. 2012: 000213-000217.

[16] Wang W J, Li J. Design of step counting algorithm based on mobile phone acceleration sensor [J]. Industrial Control Computer, 2016, (01): 75-76+79.

[17] Chen G L, Li F, Zhang Y Z. A MEMS step counting algorithm based on adaptive peak detection [J]. Chinese Journal of Inertial Technology, 2015 (3): 315-321.

[18] Liang J Z, Zhu X J, Chen J. High-precision low-sampling step counting algorithm design based on mobile phone acceleration sensor [J]. Journal of Northwest University: Natural Science Edition, 2015, 45 (5): 738-744.

[19] Kammoun S, Pothin J B, Cousin J C. An efficient fuzzy logic step detection algorithm for unconstrained smartphones [C]// Proc of International Symposium on Personal, Indoor, and Mobile Radio Communications. 2015: 259-263.

[20] Seo J, Chiang Y, Laine T H, et al. Step counting on smartphones using advanced zero-crossing and linear regression [C]// Proc of the 9th International Conference on Ubiquitous Information Management and Communication. 2015: Article No. 106.

[21] Shin S H, Park C G, Kim J W, et al. Adaptive step length estimation algorithm using low-cost MEMS inertial sensors [C]// Proc of IEEE Sensors Applications Symposium. Washington DC: IEEE Computer Society, 2007: 1-5.

[22] Alzantot M, Youssef M. UPTIME: ubiquitous pedestrian tracking using mobile phones [C]// Proc of Wireless Communications and NETWORKING Conference. 2012: 3204-3209.

[23] Yim J. A smartphone indoor positioning method [J]. International Journal of Smart Home, 2013, 7 (5): 9-18.

[24] Wang G C, Liang J Z, Chen J, et al. Acceleration differential finite state machine step counting algorithm [J]. Computer Science and Exploration, 2016, 10 (8): 1133-1142.

[25] Bebek O, Suster M A, Rajgopal S, et al. Personal navigation via shoe mounted inertial measurement units [C]// Proc of IEEE//RSJ International Conference on Intelligent Robots and Systems. 2010: 1052-1058.

[26] Lo C C, Chiu C P, Tseng Y C, et al. A walking velocity update technique for pedestrian dead-reckoning applications [C]// Proc of International Symposium on Personal, Indoor and Mobile Radio Communications. 2011: 2194-2198.

[27] Gu Y, Song Q, Li Y H, et al. Particle filter method for personnel autonomous positioning based on inertial shoe-mounted sensors [J]. Journal of Electronics & Information Technology, 2015, 37 (2): 484-488.

[28] Naqvi M N Z, Kumar A, Chauhan A, et al. Step counting using smartphone-based accelerometer [J]. International Journal on Computer Science & Engineering, 2012, 4 (5): 675-681.

[29] Ma S Y. Design and implementation of pedometer application based on iOS platform [J]. Software, 2012, 33 (12): 66-68.

[30] Ryu U J, Ahn K, Kim E, et al. Adaptive step detection algorithm for wireless smart step counter [C]// Proc of International Conference on Information Science and Applications. 2013: 1-4.

[31] Han W Z, Feng D, Li P, et al. Design of pedometer based on acceleration sensor LIS3DH [J]. Sensors and Microsystems, 2012, 31 (11): 97-99.

[32] Song H R, Liao W S, Zhao Y M. High-precision pedometer based on acceleration sensor ADXL330 [J]. Chinese Journal of Sensors and Actuators, 2006, 19 (4): 1005-1008.

[33] Delgado-Gonzalo R, Celka P, Renevey P, et al. Physical activity profiling: Activity-specific step counting and energy expenditure models using 3D wrist acceleration [C]// Proc of the 37th Annual International Conference of Engineering in Medicine and Biology Society. 2015: 8091-8094.

[34] Jr D R B, Wyatt H R, Thompson H, et al. Pedometer-measured physical activity and health behaviors in united states adults [J]. Medicine & Science in Sports & Exercise, 2010, 42 (10): 1819-1825.

[35] Ladetto Q. On Foot navigation: continuous step calibration using both complementary recursive prediction and adaptive Kalman filtering [J]. Ion Gps Salt Lake City Utah Usa, 2000: 1735-1740.

[36] Song M, Shen Y C. Research and implementation of indoor positioning dead reckoning algorithm [J]. Computer Engineering, 2013, 39 (7): 293-297.

[37] Jawbone. JAWBONE introduces UP3 [EB/OL]. (2014-11-5) [2017-8-25]. http://content.jawbone.com/static/www/pdf/press-releases/jawbone-introduces-up3.pdf.

[38] Apple. Apple watch series 3: technical specifications [EB/OL]. (2017-09-13) [2017-09-30]. https://support.apple.com/kb/SP766?viewlocale=en_US&locale=zh_CN.

[39] Oshin T O, Poslad S. ERSP: An Energy-Efficient Real-Time Smartphone Pedometer [C]// Proc of IEEE International Conference on Systems, Man, and Cybernetics. 2013: 2067-2072.

[40] Jayalath S, Nimsiri A, Murray I. A Gyroscope based Accurate Pedometer Algorithm [C]// Proc of International Conference on Indoor Positioning and Indoor Navigation. 2013.

[41] Wang J H, Ding J J, Chen Y, et al. Real time accelerometer-based gait recognition using adaptive windowed wavelet transforms [C]// Circuits and Systems. 2012: 591-594.

[42] Karantonis D M, Narayanan M R, Mathie M, et al. Implementation of a real-time human movement classifier using a triaxial accelerometer for ambulatory monitoring [J]. IEEE Trans on Information Technology in Biomedicine, 2006, 10 (1): 156-167.

[43] Das S, Green L, et al. Detecting user activities using the accelerometer on Android smartphones [EB/OL]. (2010-07-30) [2017-05-20]. https://www.truststc.org/education/reu/10/Papers

[44] Lu X L, Wang H B, Wang Y Y, et al. A human fall recognition method based on acceleration sensor [J]. Application Research of Computers, 2013, 30 (4): 1109-1111.

[45] Li Y X, Liu Y, Yuan T, et al. Multi-level classification algorithm for walking patterns based on acceleration signals [J]. Acta Electronica Sinica, 2009, 37 (8): 1794-1798.

[46] Yao Z J, Zhang Z P, Xu L Q. An Effective Algorithm to Detect Abnormal Step Counting Based on One-Class SVM [C]// Proc of IEEE International Conference on Computational Science and Engineering. Washington DC: IEEE Computer Society, 2014: 964-969.

Note: Figure translations are in progress. See original paper for figures.