



# Article A Novel Algorithm for Scenario Recognition Based on MEMS Sensors of Smartphone

Xianghong Li 몓, Hong Yuan, Guang Yang, Yingkui Gong \* and Jiajia Xu

Aerospace Information Research (AIR) Institute, Chinese Academy of Sciences (CAS), Beijing 100091, China

\* Correspondence: gongyk@aircas.ac.cn; Tel.: +86-10-8217-8896

Abstract: The scenario is very important to smartphone-based pedestrian positioning services. The smartphone is equipped with MEMS(Micro Electro Mechanical System) sensors, which have low accuracy. Now, the methods for scenario recognition are mainly machine-learning methods. The recognition rate of a single method is not high. Multi-model fusion can improve recognition accuracy, but it needs to collect many samples, the computational cost is high, and it is heavily dependent on feature selection. Therefore, we designed the DT-BP(decision tree-Bayesian probability) scenario recognition algorithm by introducing the Bayesian state transition model based on experience design in the decision tree. The decision-tree rules and state transition probability assignment methods were respectively designed for smartphone mode and motion mode. We carried out experiments for each scenario and compared them with the methods in the references. The results showed that the method proposed in this paper has a high recognition accuracy, which is equivalent to the accuracy of multi-model machine learning, but it is simpler, easier to implement, requires less computation, and requires fewer samples.

Keywords: scenario recognition; feature extraction; MEMS sensors; smartphone mode; motion mode

# 1. Introduction

With the rapid development of Micro Electro Mechanical System(MEMS)sensors, the smartphone is equipped with an inertial measurement unit (IMU), barometer and magnetometer, which provide new and cheap approaches to smartphone-based pedestrian positioning services. Now, it has penetrated all aspects of people's lives. However, because of the complex scenarios faced by the smartphone, obtaining a high-precision position based on the smartphone is still a challenge. The complex scenarios include the diversity of smartphone carrying modes, pedestrian movement modes, and the accuracy limitations of smartphone built-in sensing, which are all factors that affect the accuracy of pedestrian positioning. Contextual information is very important to the positioning system. It not only affects the types of available signals, but also provides more information for positioning, and provides an important basis for positioning methods, the selection of fusion algorithms, and failure detection. Therefore, it is necessary to identify different scenarios and choose different coping strategies for different scenarios to obtain high-precision pedestrian positioning results.

At present, pedestrian motion mode recognition is mainly divided into two research directions: one is based on image processing technology, which converts the input image or video into feature vectors, and then recognizes the motion mode [1,2]. However, it is easily infringes on personal privacy and relies heavily on light conditions [3]. The other is based on various sensors, such as accelerometers, gyroscopes, gravimeters, barometers, etc., that collect sensor data, extract various features, and classifying features. Then, various methods are used to recognize the movement pattern. Machine learning methods are mostly used for motion pattern recognition based on sensors, such as support vector machine (SVM) [4], k-nearest neighbor algorithm (KNN) [3], Gaussian naive Bayes (GNB),



Citation: Li, X.; Yuan, H.; Yang, G.; Gong, Y.; Xu, J. A Novel Algorithm for Scenario Recognition Based on MEMS Sensors of Smartphone. *Micromachines* **2022**, *13*, 1865. https://doi.org/10.3390/ mi13111865

Academic Editor: Andrea Prato

Received: 13 October 2022 Accepted: 28 October 2022 Published: 30 October 2022

**Publisher's Note:** MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



**Copyright:** © 2022 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (https:// creativecommons.org/licenses/by/ 4.0/). and artificial neural network (ANN). The average recognition success rate can reach more than 80%. For example, Sun Bingyi et al. [5] proposed a behavior recognition method based on the SC-HMM algorithm, which can classify up and down stairs and elevators with a classification accuracy of more than 80%. Jin-Shyan Lee et al. [6] proposed a threshold-based classification algorithm for carrying phone mode with an acceleration value, which is very simple and easy to achieve. Qinglin Tian et al. [7] proposed a finite state machine (FSM) to classify the smartphone mode with a classification accuracy of more than 89%.

Other scholars have tried to combine different models to improve recognition performance. Liu Bin et al. [8] combined four typical methods: k-nearest neighbor algorithm, support vector machine, naive Bayesian network, and the AdaBoost algorithm based on a naive Bayesian network to create a human activity recognition model. The optimal human activity recognition model was obtained through model decision-making, and the accuracy reached 92%. Using support vector machine (SVM) and decision tree (DT), any combination of motion state and mobile phone postures could be successfully identified [9] with an average success rate of 92.4%. Other scholars combined convolutional neural network (CNN) and long-term and short-term memory (LSTM) network to recognize walking, sitting, and lying behaviors for wearable (tied-to-the-waist) devices, with a success rate of over 96% [10,11]. In addition, some scholars used other methods to realize motion pattern recognition, such as the implicit Markov model [12,13], the sensor data interception method based on last bit matching [14], and the voting method [15]. Some scholars have also studied the influence of window length on human motion pattern recognition, to choose the optimal window length [16,17].

Ichikawa et al. [18] studied the ways people like to use mobile phones. The common locations of mobile phones are trouser pockets, clothing pockets, hand-held and so on. Scholars have explored various methods to identify the common locations, that is, to identify the location of the mobile phone. Yang et al. [19] proposed PACP (Parameters Adjustment-Corresponding-to-smartphone position), a method that is independent of the smartphone mode. It uses the SVM (support vector machine) model to identify the smartphone mode with an accuracy rate of 91%. Deng et al. [20] proposed to recognize the location of mobile phones based on accelerometer features, and tested the recognition results based on SVM, Bayesian network, and random forest. Noy et al. [21] used KNN, decision tree, and XGBoost to test and compare, and showed that XGBoost has the best recognition success rate. Wang [22] proposed a recognition method of the superimposed model, which combines the six models of AdaBoost, DT, KNN, LightGBM, SVM, and XGBoost to realize the location recognition of smartphones, and the recognition accuracy can reach 98.37%.

In general, the methods for scenarios recognition mainly focus on machine learning methods, such as SVM, CNN, KNN, etc. These methods have a low recognition accuracy rate when recognizing based on raw data, and when recognizing based on sensor features, they have a strong dependence on feature selection. The fusion of multiple models can improve the recognition accuracy, increasing the complexity of calculations and requiring a large number of samples. In addition, the calculation cost is large, and the choice of features is heavily dependent.

To solve this problem, we designed a DT-BP (decision tree-Bayesian probability) scenarios recognition algorithm by using a single model decision tree and a Bayesian state transition model, which aims at motion mode and smartphone mode. This method is more simplified, less computationally expensive, and less computationally complex, and can obtain the same recognition accuracy as the multi-model machine learning method. The contributions of this study are as follows:

We designed a decoupling analysis method to analyze the relationship between e different kinds of scenario, so as determine the identification order. As the interactions of different scenario are categorised, adverse effects on scenario recognition occur. Therefore, a decoupling relationship analysis method was designed to decouple different scenario categories and determine the sequence of scenario type identification;

- We designed a DT-BP (decision tree-Bayesian probability) scenario recognition algorithm by using a single model decision tree and a Bayesian state transition model, which aimed at motion mode and smartphone mode. This method is more simplified, less computationally expensive, and less computationally complex, and can obtain the same recognition accuracy as the multi-model machine learning method;
- We designed the corresponding decision tree criteria and probability allocation method for smartphone mode and motion mode. We carried out experiments for each scenario and compared them with the methods in the references.

The rest of this paper is organized as follows: Section 2 introduces the presented algorithm, including decoupling analysis, feature extraction and scenario recognition algorithm. Section 3 shows the experimental setup, results and discussion. And finally, Section 4 concludes the paper.

#### 2. Methodology

## 2.1. Decoupling Analysis of Scenario Category

It is necessary to analyze the decoupling relationship of different scenario categories to determine their independence and correlation. For example, if there are nk motion modes, mk smartphone modes, there are  $nk \times mk$  kinds of situations in any combination of the two kinds of contexts. It is too complicated and redundant to identify all the combined scenarios. And as the interaction of different scenario is categorized, adverse effects on scenario recognition occur. Therefore, a decoupling relationship analysis method was designed to decouple different scenario categories and determine the sequences of scenario type recognition.

The decoupling of smartphone modes and motion modes needs to be analyzed in three parts:

- The correlation coefficient of the same motion mode in different smartphone modes;
- The correlation coefficient of different motion modes in the same smartphone mode;
- The correlation coefficient between different smartphone modes and different motion modes.

We used Pearson's correlation coefficient to analyze the decoupling of the data. The calculation formula of the correlation coefficient is as follows:

$$r = \frac{\sum_{i=1}^{n} (x_i - \overline{x})(y_i - \overline{y})}{\sqrt{\sum_{i=1}^{n} (x_i - \overline{x})^2} \sqrt{\sum_{i=1}^{n} (y_i - \overline{y})^2}}, \begin{cases} \overline{x} = \sum_{i=1}^{n} x_i/n \\ \overline{y} = \sum_{i=1}^{n} y_i/n \end{cases}$$
(1)

where *r* is the correlation coefficient, *n* is the length of the data,  $x_i$ ,  $y_i$  which are two different time series.

As the data sampling length in different situation is different and the data contains the periodic behavior of pedestrian movement, it was necessary to establish a time-lag series [23]. The sequence  $(X, Y) = \{(x_i, y_i), i = 1, 2, ..., n\}$ , after moving forward and backward by m sampling points is:

$$X_{t}, Y_{t+m}) = \{(x_{i}, y_{i+m}), i = 1, 2, \dots, n-m\}, 1 \le m < n, m \in N^{+} \\ X_{t}, Y_{t-m}) = \{(x_{i}, y_{i-m}), i = m+1, m+2, \dots, n\}, 1 \le m < n, m \in N^{+}$$

$$(2)$$

If the time-shift sequence is correlated, it must exist  $m_0(1 \le |m_0| \le n, |m_0| \in N^+)$  to maximize the correlation coefficient of  $(X_t, Y_{t+m_0})$ .

To decouple different scenario categories, the following analysis method was used:

(1) To avoid dependence on feature selection, raw data were selected for data analysis;

To ensure a full analysis of different scenario categories, it was necessary to exclude other factors as far as possible, such as pedestrian differences, smartphone brand differences, etc. Therefore, the window length n shall meet the pedestrian movement cycle, generally 0.5–1.2 s. The calculation method of window length n is as follows:

$$n > 2 \times \left[\frac{\mathrm{T}_n}{\Delta \mathrm{t}}\right] \tag{3}$$

where  $T_n$  is the window period, generally bigger than 2 s.  $\Delta t$  is the sampling period, which depends on smartphone's brand and model. [] means round numbers.

(2) To ensure the integrity of pedestrian motion cycle, the forward and backward sampling points m should be selected as:

$$\left[0.5 \times \frac{T_n}{\Delta t}\right] + 1 \le m < n \tag{4}$$

(3) To ensure the analysis is not disturbed by abnormal data, the sliding window is used to calculate the correlation coefficient r, which is  $\Re = r_i(i = 1, 2, ..., k), k = [N/n]$ , and N is the sampling length of data. The decoupling analysis correlation coefficient is

$$\overline{r} = \frac{1}{k} \sum_{i=1}^{k} r_i \tag{5}$$

(4) To analyze the decoupling correlation different scenario categories, we set  $r(x_{i,j}, y_{u,v}) = \overline{r}$ . Where *i* and *u* represent the smartphone mode, *j* and *v* represent the motion mode.  $y_{u,v}$  and  $y_{u,v}$  are two kinds of scenario. To obtain the analysis result we needed to analyze three situations as follows:

$$\begin{cases}
i = u, j \neq v \\
i \neq u, j = v \\
i \neq u, j \neq v
\end{cases}$$
(6)

The test results are shown in Table 1, which gives the correlation calculation results of a total of nine scenarios composed of three motion modes and three smartphone modes. The raw data include GNSS sensor, accelerometer, gyroscope, magnetometer, barometer, and Bluetooth.

Table 1.	Pearson	correlation	coefficients	of different	t motion m	nodes and	different s	martphone	nodes.
								1	

		Wa				US			UE	
		Calling	Texting	Cloths Pocket	Calling	Texting	Cloths Pocket	Calling	Texting	Cloths Pocket
	Calling	0.77	-	-	-	-	-	-	-	-
Wa	Texting	0.56	0.64	-	-	-	-	-	-	-
	Cloths pocket	0.55	0.53	0.77	-	-	-	-	-	-
	Calling	0.35	0.18	0.21	0.68	-	-	-	-	-
US	Texting	0.27	0.34	0.25	0.55	0.80	-	-	-	-
	Cloths pocket	0.27	0.24	0.33	0.51	0.52	0.67	-	-	
	Texting	0.32	0.21	0.23	0.26	0.26	0.21	0.71	-	-
UE	Cloths pocket	0.21	0.33	0.28	0.18	0.33	0.25	0.52	0.86	-
	Texting	0.21	0.25	0.35	0.22	0.15	0.31	0.65	0.55	0.72

Wa is Walking, US is upstairs, UE is up elevator.

According to the test results in Table 1, the decoupling correlation can be summarized as follows:  $(0.2 \le r(r - tr)) \le 0.4$  submit = tr - i = r

$$\begin{cases} 0.3 < r(x_{i,j}, y_{u,v}) < 0.4, when i = u, j \neq v \\ r(x_{i,j}, y_{u,v}) > 0.5, when i \neq u, j = v \\ r(x_{i,j}, y_{u,v}) < 0.3, when i \neq u, j \neq v \end{cases}$$
(7)

From Formula (7) we know that the correlation between different motion modes has a certain correlation under the same smartphone mode. In the case of different smartphone modes, the correlation of the same motion mode is greater than 0.5. The correlation between different smartphone modes and different motion modes was very low, and less than 0.3. So, smartphone modes have little influence on motion mode recognition. On the contrary, motion modes have a great influence on smartphone mode recognition.

According to the above analysis, during scenario recognition, we can recognize the motion mode first. And when this is determined, the smartphone mode is recognized secondly.

# 2.2. Feature Extraction

In this paper, we extracted features from different sensor data in both the time domain and frequency domain, respectively. The time-domain refers to the extraction of the mathematical-statistical characteristics of the sensor measurement data in a certain window length, such as variance, mean, amplitude, etc. The frequency domain refers to the calculation of the Fourier transform and frequency domain entropy of the sensor measurement data in a certain window length. Then, the features in the frequency domain were extracted, such as dominant frequency, energy, frequency difference, etc.

As shown in Table 2, the time-domain features extracted from active sensors were used in this paper. Where  $n_1, n_2, n_3, n_4, n_5, n_6$  is the length of the data window,  $x_i, y_i, z_i$  is the sampling data,  $\overline{x}$  is the mean value, i,j,k are the times at different sampling points,  $s_k$  is the value that the data in the window length  $n_4$  is greater and smaller than the threshold value  $thre_{zero}, thre_{-zero}$ .

Feature	Definition
First-order norm	$  x  _1 =  x_i $
Peaks	$a_{\max} = \max(x_i), i = 1, \dots, n_1$
Wave	$a_{\min} = \min(x_i), i = 1, \dots, n_1$
Difference of Peaks and Wave	$a_{mm} = a_{max} - a_{min}$
Mean	$\overline{x} = \frac{1}{n} \sum_{i=1}^{n} x_i$
Variance	$\sigma^2 = \frac{1}{n_3} \sum_{i=1}^{n_3} (x_i - \overline{x})^2$
Amplitude	$\text{svm} = \sqrt{x_i^2 + y_i^2 + z_i^2}$
Zero-Crossing Rate	$p_{0} = \frac{s_{k}}{n_{4}}, \begin{cases} s_{k} = s_{k} + 1, if(x_{i} > thre_{zero}) \\ s_{-k} = s_{-k} + 1, if(x_{i} < thre_{-zero}) \end{cases}$
Gradient	$dx = x_{k+i} - x_k$
correlation coefficient	$r_k = rac{\sum_{i=1}^{n_5-k} (x_i - \overline{x}) (x_{i+k} - \overline{x})}{\sum_{i=1}^{n_5} (x_i - \overline{x})^2}$
Fourier Transform	$X(k) = \sum_{i=0}^{n-1} x(i) W_{n_6}^{ki}, k = 0, 1, \dots, n-1, W_{n_6} = e^{-j\frac{2\pi}{n_6}}$

Table 2. Feature extraction in time domain and frequency domain [24].

The height gradient value [25,26] is calculated by the raw data of barometer as:

$$dh = h - h_0 = 18400 \cdot \left(1 + \frac{t}{273.15}\right) \cdot \lg \frac{p_0}{p}$$
(8)

where *t* is the temperature, and the unit is °C.  $p_0$  is the reference air pressure and *p* is the output of the barometer.

#### 2.3. Scenario Recognition Algorithm

2.3.1. Design of Scenario Recognition Algorithm Based on DT-BP

The decision tree (DT) establishes the nodes by exploring the high-value data features in the overall data and constructs the branches of the tree according to the required research contents. With repeatedly establishing the branch nodes, the classification results and decision set contents are displayed with the tree structure [27,28]. The decision tree has the advantages of low computational complexity and is insensitive to missing content in the middle. It can handle irrelevant feature data [29]. Decision trees also have shortcomings, such as low detection accuracy and the work needed for the preprocessing of time-sequential data. In the actual environment, due to the complexity of the environment, the different interference, the performance of different devices, the error accumulation of the sensor itself, etc., the error in the identification process is so relatively large that the usability is not high. To deal with this problem, we designed a context recognition method based on the decision tree and Bayesian state transition probability (decision tree-Bayesian probability, DT-BP).

Bayes theory is a common method in model decision-making. The basic idea is to know the conditional probability density parameter expression and a priori probability, convert the formula into a posteriori probability, and finally use a posteriori probability for decision classification.

If P(A) is a priori probability or edge probability of A. the conditional probability of A after the occurrence of B is P(A|B), which is called the posterior probability of A. P(B|A) is the conditional probability of B after the occurrence of A, which is called the posterior probability of B. P(B) is the prior probability of B [30]. Then

$$P(B|A) = \frac{P(A|B)P(B)}{P(A)}$$
(9)

For the situation recognition in this paper, we suppose  $T_i$  as the situation category, and  $T_i \in \Gamma(b_1, b_2, ..., b_u)$ . Where U is the number of situations. We assume  $S_f$  as the feature quantity set, and  $S_f = \{a_1, a_2, ..., a_V\}$ . Where V is the number of features. If the features of  $S_f$  all belong to  $T_i$ , the probability is  $P(T_i | S_f)$ . When it is satisfied  $P(T_k | S_f) = \max_{i=1, \cdots, K} P(T_i | S_f)$ , it is considered  $S_f \in T_k$ , which means the recognition is successful. Therefore, we only need to calculate  $P(T_i | S_f)$  to recognize context. The Formula (9) is changed to:

$$P(T_i|S_f) = \frac{P(T_i)P(S_f|T_i)}{P(S_f)}$$
(10)

To obtain the conditional probability of the scenario  $T_i$ , we need to calculate  $P(S_f)$ ,  $P(T_i)$  and  $P(S_f|T_i)$ .

The principle of probability allocation in this article: the number of features quantities is *V*, and the probability of each feature quantity is the same, which means  $P(S_f)$  is a constant. So  $P(T_i|S_f)$  is the largest when  $P(T_i)P(S_f|T_i)$  is the largest. That is

$$P(T_i|S_f) \propto P(T_i)P(S_f|T_i)$$
(11)

where  $P(T_i)$  is the state probability. Its value at the current moment is related to the number of scenarios to be detected and the probability of the previous moment. It is independently designed according to different scenario categories and the number of scenarios  $U. P(S_f | T_i)$  is the conditional probability of the feature vector  $S_f$ , which is obtained from DT rules. The obtaining algorithm designed in this paper is as follows:

$$P\left(S_f|T_i\right) = \frac{\sum_{j=1}^{K_i} c_j}{K_i} \tag{12}$$

where  $K_i$  represents the number of features related to the category  $T_i$ .  $c_j$  is the judgment value of each feature. If the judgment condition is met, it is  $c_j = 1$ , otherwise is  $c_j = 0$ .

- 2.3.2. Recognition of Smartphone Mode Based on DT-BP
- (1) Algorithm design based on decision tree

To realize the recognition of different smartphone modes, it is necessary to detect the transformation process between different smartphone modes, which determines whether the smartphone mode is transforming or fixed. When it is in the fixed position, the different smartphone mode is determined. There are two parts of mobile phone location recognition: transformation recognition and current location recognition. In this paper, we took six common smartphone modes [18] as examples to design the specific decision tree, including texting, calling, pants front pocket, clothes pocket, pants back pocket and hand swing, as shown in Figure 1.



Figure 1. The example of decision tree for smartphone modes recognition.

According to the data characteristic analysis and feature analysis, the variance of the first-order norm of acceleration was used as the criterion for judging the position change of the mobile phone. The first-level decision criterion for the decision tree is:

$$if var(ax) > Thre \& var(ay) > Thre \& var(az) > Thre, changing = 1$$

$$others, changing = 0$$
(13)

where *Thre* is the threshold. When changing = 1, the location is changing, otherwise it is not.

When the position is fixed, it is necessary to determine whether there is periodic oscillation, which means there are other periodic motions besides walking and swinging with the pedestrian. We use the second main frequency amplitude of the acceleration amplitude to determine it. The second-level decision criterion of the decision tree is:

$$\begin{cases} if(A_{mf2} > Thre_{mf2}), per = 1; \\ others, per = 0 \end{cases}$$
(14)

where  $Thre_{mf2}$  is the threshold. When per = 1, the position of the mobile phone has periodic movement, otherwise it is not. If the smartphone location with periodic movement, the judgment criterion is as follows:

$$\begin{cases} if(F_{mf} > Thre_{mf}), swpp = 1; \\ others, swpp = 0 \end{cases}$$
(15)

where  $Thre_{mf}$  is the threshold. When swpp = 1, the position is pants pocket(pp), otherwise it is swinging. As the features of the front pants pocket(fpp) and back pants pocket(bpp) is similar, there is a new branch for them, and the decision criterion is:

$$\begin{cases} if \ \frac{1}{M}\sum_{i=1}^{M}(ax-az) > Thre_{qh}, fpp = 1\\ others, bpp = 1 \end{cases}$$
(16)

where *M* is the window length,  $Thre_{qh}$  is the threshold.

To recognize the fixed smartphone mode without periodic motion, we used the firstorder norm of acceleration, peaks, and wave as the features. The determination rule for designing a fixed smartphone mode decision tree is:

$$\begin{array}{ll} text: |ax| < Thre1, & |ay| < Thre1, & az_{\min} > Thre2\\ call: ax_{\max} < -Thre1, & ay_{\min} > Thre2, & |az| < Thre1\\ pocket: & ax_{\min} > Thre2, & ay_{\max} < -Thre1, & |az| < Thre1 \end{array}$$
(17)

where Thre1, Thre2, Thre3 are thresholds.

(2) Probabilistic design based on DT-BP method

According to the design of the DT-BP method, we needed to design  $P(T_i)$  and  $P(S_f|T_i)$  to calculate the probability distribution principle. The number of scenarios U is 7, including 6 smartphone positions and the change process.  $P(T_i)$  is designed as shown in Table 3. According to the design of the decision tree, the number of features V is 10. The design of  $K_i$  in Formula (12) is as follows in Table 4.

	Changing	Text	Call	Fpp	Pocket	Врр	Swing
Changing	1/7	1/7	1/7	1/7	1/7	1/7	1/7
Text	1/2	1/2	-	-	-	-	-
Call	1/2	-	1/2	-	-	-	-
Fpp	1/2	-	-	1/2	-	-	-
Pocket	1/2	-	-	-	1/2	-	-
Врр	1/2	-	-	-	-	1/2	-
Swing	1/2	-	-	-	-	-	1/2

Table 3. Smartphone mode transition probability allocation.

Table 4. Smartphone mode related feature quantity allocation.

	Changing	Text	Call	Fpp	Pocket	Врр	Swing
K <sub>i</sub>	3	7	7	6	7	6	5

2.3.3. Recognition of Motion Mode Based on DT-BP

(1) Design of motion mode recognition algorithm based on decision tree

In this paper, the recognition algorithm is designed by taking the motion modes of static, walking, turning, going upstairs and downstairs, escalator, and elevator as examples. According to the analysis of the extracted features, the dynamic and static are distinguished by the acceleration variance. We used positive and negative zero-crossing rates of acceleration air pressure gradients as the features of static, elevator and escalator recognition. Dynamic motion includes walking on the ground, going upstairs, going downstairs and turning. Walking, turning, going upstairs and downstairs are coupled movements, that is, turning and going up and down stairs also have walking movements. Therefore, the main work was to distinguish turning and going up and down stairs from walking. The amplitude of angular velocity is used to recognize turning. We used the auto-correlation coefficient, Fourier dominant frequency, and elevation gradient to recognize going up and down stairs and walking, as shown in Figure 2.



Figure 2. The example of decision tree for motion modes recognition.

The process of motion mode recognition based on the decision tree method is as follows:

- (a) If var(a) is greater than the threshold *Thre*<sub>*a*</sub>, it is considered dynamic, otherwise, it is static.
- (b) When it is static, it is necessary to distinguish between static and elevator and escalator. We use the zero-crossing rate of acceleration amplitude as judgment, and the judgment criterion is:

$$\begin{cases} flag_{el} = 1, if (p_{+}(a_{k}) > Thre_{+el} \& p_{-}(a_{k}) < Thre_{-el}) \\ flag_{el} = -1, if (p_{-}(a_{k}) > Thre_{+el} \& p_{+}(a_{k}) < Thre_{-el}) \\ flag_{el} = 0, otherwise \end{cases}$$
(18)

where  $a_k = svm_a - g$ ,  $svm_a$  is the amplitude of acceleration and g is the acceleration of gravity.  $p_+(a_k)$  and  $p_-(a_k)$  is the positive and negative zero-crossing rate.  $Thre_{+el}$  and  $Thre_{-el}$  are the zero-crossing judgment threshold. If  $flag_{el} \neq 0$ , it is going up or down elevator.

- (c) When the motion is the elevator, we judge whether going up or down. If  $flag_{el} = 1$  and the previous state is not the elevator, the state is going up elevator. If  $flag_{el} = 1$  and the previous state is the elevator, the state is going down elevator. If  $flag_{el} = -1$  and the previous state is not the elevator, the state is going down elevator. If  $flag_{el} = -1$  and the previous state is not the elevator, the state is going up elevator. If  $flag_{el} = -1$  and the previous state is the elevator, the state is going up elevator. If  $flag_{el} = -1$  and the previous state is the elevator, the state is going up elevator.
- (d) If  $flag_{el} = 0$ , it is further recognized whether it is an escalator, and the recognition criteria are as follows:

$$\begin{cases} flag_{es} = -1, if (p_{+}(dbro) > Thre_{+es}\&p_{-}(dbro) < Thre_{-es}) \\ flag_{es} = 1, if (p_{-}(dbro) > Thre_{+es}\&p_{+}(dbro) < Thre_{-es}) \\ flag_{es} = 0, otherwise \end{cases}$$
(19)

where *dbro* is the pressure gradient,  $p_+(dbro)$  and  $p_-(dbro)$  are the positive and negative zero-crossing rates of the pressure gradient. *Thre*<sub>+es</sub> and *Thre*<sub>-es</sub> are the thresholds. When  $flag_{es} = 0$ , it is the static state. If  $flag_{es} = 1$ , the escalator is going up. If  $flag_{es} = -1$ , it is the elevator is going down.

- (e) When pedestrians are in a dynamic state, we mainly distinguish turning, stairs, and walking. The angular velocity amplitude is used to recognize turning. When  $svm_w > Thre_w$ , it is turning.
- (f) We use the auto-correlation coefficient and Fourier transform to distinguish stairs and walking. The auto-correlation coefficient at the offset k = 2 and k = 4 as the judgment

value, and the main frequency of the Fourier transform as the judgment condition. The criterion is as follows:

$$\begin{cases} flag_{st} = 1, if\left(|R_a(2)| > |R_a(4)|\&F_a > Thre_{fa}\right) \\ flag_{st} = 0, otherwise \end{cases}$$
(20)

where  $R_a(2)$  is the correlation coefficient of the acceleration amplitude offset k = 2.  $F_a$  represents the main frequency of the Fourier transform.  $Thre_{fa}$  is the main frequency judgment threshold. If  $flag_{st} = 1$ , it is up and downstairs, otherwise it is walking on level ground.

(g) When pedestrians are going up and down the stairs, we use the height gradient value to make judgments. The criterion is as follows:

$$\begin{cases} flag_h = 1, if (dh > Thre_h) \\ flag_h = -1, if (dh < Thre_{-h}) \end{cases}$$
(21)

where  $Thre_h$  and  $Thre_{-h}$  are the judgment thresholds of the height gradient. If  $flag_h = 1$ , it is going up the stairs. If  $flag_h = -1$ , it is going down the stairs.

(2) Probabilistic design based on DT-BP method

For the nine motion modes in the above example, the probability  $P(T_i)$  of each state in the DT-BP method is determined according to the previous state. The transition probability between each motion mode is shown in Table 5. According to Figure 2 and the decision tree rules, the number of features *V* is 12, as shown in Table 6 for the design  $K_i$  in Formula (12).

	Static	Wa	US	DS	UE	DE	UC	DC	Turning
Static	1/8	1/8	1/8	1/8	1/8	1/8	1/8	1/8	-
Wa	1/9	1/9	1/9	1/9	1/9	1/9	1/9	1/9	1/9
US	1/4	1/4	1/4	-	-	-	-	-	1/4
DS	1/4	1/4	-	1/4	-	-	-	-	1/4
UE	1/3	1/3	-	-	1/3	-	-	-	-
DE	1/3	1/3	-	-	-	1/3	-	-	-
UC	1/3	1/3	-	-	-	-	1/3	-	-
DC	1/3	1/3	-	-	-	-	-	1/3	-
Turning	1/5	1/5	1/5	1/5	-	-	-	-	1/5

Table 5. Motion mode transition probability allocation.

Wa is Walking, US is upstairs, DS is down stairs, UE is up elevator, DE is down elevator, UC is up escalator, DC is down escalator.

 Table 6. Motion mode related feature quantity allocation.

	Static	W	US	DS	UE	DE	UC	DC	Turning
K <sub>i</sub>	5	5	6	6	4	4	6	6	2

# 3. Experimental

3.1. Experimental Setup

To understand the effectiveness and limitations of our proposed Scenario recognition algorithm, we conducted an implementation on Android to collect data. During the experiment, we collected data using an Android smartphone (Huawei mate 8, whose parameters are shown in Table 7), which was equipped with a three-axis accelerometer and a three-axis gyroscope. We evaluated the proposed method in six common smartphone modes (texting, calling, pants front pocket, clothes pocket, pants back pocket and hand swing) and nine natural motion modes (static, walking, turning, upstairs and downstairs, up escalator, down escalator, up elevator and down elevator). The threshold of smartphone mode and motion mode in the experimental are shown in Table 8. We compared the proposed algorithm with state-of-the-art algorithms.

 Table 7. Huawei mate 8 smartphone sensor and related parameters.

Sensors	Туре	Parameters
GNSS sensor	_	Support GPS, A-GPS, GLONASS and BDS
accelerometer	LSM330	Sensitivity: 0.0095768068 m/S <sup>2</sup> ; Measurement Range: 78.4532012939 m/S <sup>2</sup>
gyroscope	LSM330	Sensitivity: 0.0012217305 rad/s; Measurement Range: 34.9065856934 rad/s
magnetometer	AK09911	Sensitivity: 0.0625 μT; Measurement Range: 2000 μT
barometer	AirPress sensor Rohm, BM1383	Sensitivity: 0.0099999998 hPa; Measurement Range: 1100 hPa
Bluetooth	_	4.2 + BLE

Table 8. The threshold of smartphone mode and motion mode.

Smartpho	ne Mode	Motion	Mode
Threshold	Number	Threshold	Number
Thre	2	Threa	1.5
$Thre_{mf2}$	1.5	$Thre_{+el}$	0.9
$Thre_{mf}$	10	$Thre_{-el}$	0.05
Threah	1	$Thre_{+es}$	0.9
Thre1	6	$Thre_{-es}$	0.05
Thre2	6	Thre <sub>fa</sub>	14
Thre3	8	Thre <sub>h</sub>	0.1
		$Thre_{-h}$	-0.1

## 3.2. Experimental Results of Smartphone Mode Recognition

To test the algorithm of smartphone mode recognition, we collect text, calling, front pants pocket, back pants pocket, clothes pocket, and hand-held positions, respectively. There was a total of 9901 epoch data, including 568 epoch data in the changing, 2691 epoch data in texting, 1236 epoch data in calling, 2095 epoch data in the front pants pocket 1512 epochs in the back pants pockets, 1561 epochs in clothes pocket and 238 epochs in swing.

The accuracy of smartphone mode recognition is shown in Table 9. The recognition accuracy was calculated with the following formula:

$$err = \frac{Num_{reco}}{Num_{true}} \times 100\%$$
(22)

where  $Num_{reco}$  is the number of epochs where the state recognition result is the same as the state to be recognized.  $Num_{true}$  is the actual number of epochs in the state to be recognized. The results were similar to Table 9 in many tests. The average recognition accuracy was 99.06%, the lowest accuracy was 96.48%, and the recognition accuracy of all positions was greater than 96%. The main reason is that the smartphone mode changing was followed by various fixed smartphone modes. The error of various smartphone modes will be reflected in the smartphone mode changing.

The comparison of the recognition accuracy of the algorithm in this paper with other different algorithms [20,31] is shown in Table 10. From the table, the random forest has the highest accuracy when texting and calling. DT-BP has the highest success rate in hand-held. The average accuracy in this paper is slightly higher by 0.5%. Compared with other methods based on machine learning, the DT-BP proposed in this paper has a slightly higher recognition accuracy, and takes 0.51 s in total, while the random forest takes 8.34 s.

	Changing	Text	Call	Fpp	Pocket	Врр	Swing
Changing	96.48%	0.18%	0	0	2.64%	0	0.70%
Text	2.04%	97.96%	0	0	0	0	0
Call	1.13%	0	98.87%	0	0	0	0
Fpp	0.29%	0	0	99.71%	0	0	0
Pocket	0	0	0	0	100%	0	0
Врр	0.40%	0	0	0	0	99.6%	0
Swing	1.68%	0	0	0	0	0	98.32%

The algorithm in this paper greatly reduces the operation time and ensures the recognition success rate.

Tal	ble 9	9. 5	Smartpl	hone	mode	recog	gnition	result	s
-----	-------	------	---------	------	------	-------	---------	--------	---

 Table 10. Comparison of smartphone mode recognition accuracy for different methods.

	Text	Call	Pocket	Swing	Average
DT-BP	97.2%	98.0%	99.77%	98.32%	98.32%
SVM	97.96%	95.8%	95.2%	98.1%	96.765%
Bayesian Network	94.5%	97.2%	92.2%	93.6%	94.375%
Random Forest	99.0%	98.2%	97.1%	96.9%	97.8%

## 3.3. Experimental Results of Motion Mode Recognition

We collected static, walking, up and down stairs, elevator up and down, and escalator up and down data, respectively, to test DT-BP. There was a total of 6789 epoch data, including 3239 epochs in static, 2005 epochs in walking, 173 epochs in up the stairs, 165 epochs in down the stairs, 317 epochs in up the elevator, 319 epochs in down the elevator, 145 epochs in up the escalator, and 426 epochs in down the escalator. The recognition accuracy was calculated by the Equation (22).

The accuracy of motion recognition is shown in Table 11, which was similar to many tests. The average recognition accuracy of DT-BP was 97.3%, and the lowest was 88.73%, which is down escalator. The main reason is that there is a parallel stage of the escalator when getting on the escalator and preparing to get off the escalator. During this period, the speed is uniform. It is difficult to recognize as the speed is submerged in the noise of acceleration. The accuracy of the elevator going up and down and turning is the highest, mainly because the acceleration change characteristics are obvious in these states. Since the intermediate transition state of each motion process is static, it will be considered static when detecting errors in the other eight states. Therefore, there will be more states of detecting errors at rest.

	Static	W	US	DS	UE	DE	UC	DC	Turning
Static	96.10%	2.07%	0.27%	0.24%	0	0.03%	0.03%	1.25%	0
W	0.17%	99.83%	0	0	0	0	0	0	0
US	0	0	100%	0	0	0	0	0	0
DS	0	0	0	100%	0	0	0	0	0
UE	0.95%	0	0	0	99.05%	0	0	0	0
DE	1.88	0	0	0	0	98.12%	0	0	0
UC	6.46%	0	0	0	0	0	93.54%	0	0
DC	11.27%	0	0	0	0	0	0	88.73%	0
Turning	0	0	0	0	0	0	0	0	100%

Table 11. Motion mode recognition results.

To further analyze DT-BP proposed in this paper, it is compared with other algorithms [9,10,32–34], as shown in Figure 3. The recognition accuracy using a single machine learning model was relatively lower. For example, SVM and KNN are both more than

13 of 14

80%. The recognition success rate using multiple models will significantly improve with increases in algorithm complexity and calculation cost. The accuracy of DT-BP is the same as the method of machine learning using multiple models, and the computational complexity and computational cost are significantly reduced.



Figure 3. Comparison of motion mode recognition accuracy for different methods.

#### 4. Conclusions

At present, the methods for scenario recognition are mainly machine-learning methods. The recognition accuracy of a single model is not high. Multi-model fusion can improve recognition accuracy. However, the computational cost is high, and it is heavily dependent on feature selection. We mainly focused on two types of contexts: sports mode, and mobile phone location to design a DT-BP recognition algorithm by introducing the Bayesian state transition model based on experience design into the decision tree. It is more simplified and easier to implement and has less computation and lower computational complexity. In addition, it can obtain the same recognition accuracy as the multi-model machine learning method.

**Author Contributions:** Conceptualization, X.L. and H.Y.; methodology, X.L.; software, Y.G.; validation, X.L., Y.G. and G.Y.; formal analysis, X.L.; investigation, H.Y.; resources, H.Y.; data curation, Y.G.; writing—original draft preparation, X.L.; writing—review and editing, X.L. and H.Y.; visualization, X.L.; supervision, Y.G. and J.X.; project administration, G.Y. and J.X.; funding acquisition, Y.G. All authors have read and agreed to the published version of the manuscript.

**Funding:** This work was supported in part by the Space information technology innovation workstation, CX2022-04-03-02.

Conflicts of Interest: The authors declare no conflict of interest.

# References

- Simonyan, K.; Zisserman, A. Two-Stream Convolutional Networks for Action Recognition in Videos. In Proceedings of the 27th International Conference on Neural Information Processing Systems, Montreal, QC, Canada, 8–13 December 2014; pp. 568–576.
- 2. Luo, P. Research on Pedestrian Detection and Action Recognition Based on Deep Learning. Master's Thesis, Control Science and Engineering, University of Science and Technology of China, Hefei, China, 2020.
- Zhou, X.; Ran, F.; Huang, Y.; Kong, X. Improved PDR indoor location based on KNN motion pattern recognition. *Geospat. Inf.* 2019, 17, 25–28.
- Guo, J.; Wang, W.; Zhang, S. Pedestrian Motion Modes Recognition of Smart Phone Based on Support Vector Machine. Bull. Surv. Mapp. 2018, 2, 1–5.
- 5. Sun, B.; Lü, W.; Li, W.Y. Activity Recognition Based on Smartphone Sensors and SC-HMM Algorithm. J. Jilin Univ. (Sci. Ed.) 2013, 51, 1128–1132.
- Lee, J.; Huang, S. An experimental heuristic approach to multi-pose pedestrian dead reckoning without using magnetometers for indoor localization. *IEEE Sens. J.* 2019, 19, 9532–9542. [CrossRef]
- Tian, Q.; Salcic, Z.; Kevin, I.; Wang, K.; Pan, Y. A multi-mode dead reckoning system for pedestrian tracking using smartphones. *IEEE Sens. J.* 2016, 16, 2079–2093. [CrossRef]
- 8. Liu, B.; Liu, H.; Jin, X.; Guo, D. Human activity recognition based on sensors of smart phone. *Comput. Eng. Appl.* **2016**, *52*, 188–193.
- 9. Wang, B.; Liu, X.; Yu, B.; Jia, R.; Gan, X. Pedestrian Dead Reckoning Based on Motion Mode Recognition Using a Smartphone. *Sensors* **2018**, *18*, 1811. [CrossRef] [PubMed]

- 10. Chen, X.; Zhang, T.; Zhu, X.; Mo, L. Human behavior recognition method based on fusion model. *Transducer Microsyst. Technol.* **2021**, *40*, 142–145, 149.
- Chen, F.; Chen, H.B.; Wang, W.G. Human Motion Recognition Method Based on CNN-LSTMs Hybrid Model. Inf. Technol. Informatiz. 2019, 4, 32–34.
- 12. Li, F.; Pan, J.K. Human Motion Recognition Based on Triaxial Accelerometer. J. Comput. Res. Dev. 2016, 53, 621–631.
- Xiong, H.; Guo, S.; Zheng, X.; Zhou, Y. Indoor Pedestrian Mobile Activity Recognition and Trajectory Tracking. *Geomat. Inf. Sci. Wuhan Univ.* 2018, 43, 1696–1703.
- 14. Su, B.; Zheng, D.; Tang, Q.; Sheng, M. Human daily short-time activity recognition method driven by single sensor data. *Infrared Laser Eng.* **2019**, *48*, 282–290.
- 15. Qian, L.P.; Li, P.H.; Huang, L. An Adaptive Algorithm for Activity Recognition with Evolving Data Streams. *Chin. J. Sens. Actuators* 2017, *37*, 909–915.
- 16. Wang, G.; Li, Q.; Wang, L.; Wang, W.; Wu, M.; Liu, T. Impact of Sliding Window Length in Indoor Human Motion Modes and Pose Pattern Recognition Based on Smartphone Sensors. *Sensors* **2018**, *18*, 1965. [CrossRef]
- 17. Banos, O.; Galvez, J.-M.; Damas, M.; Pomares, H.; Rojas, I. Window size impact in human activity recognition. *Sensors* **2014**, *14*, 6474–6499. [CrossRef]
- Ichikawa, F.; Chipchase, J.; Grignani, R. Where's the phone? A study of mobile phone location in public spaces. In Proceedings of the International Conference on Mobile Technology, Nice, France, 2–4 September 2009; pp. 1–8.
- Yang, R.; Wang, B.W. PACP: A Position-Independent Activity Recognition Method Using Smartphone Sensors. *Information* 2016, 7, 72. [CrossRef]
- 20. Deng, Z.-A.; Wang, G.; Hu, Y.; Cui, Y. Carrying Position Independent User Heading Estimation for Indoor Pedestrian Navigation with Smartphones. *Sensors* **2016**, *16*, 677. [CrossRef]
- 21. Noy, L.; Bernard, N.; Klein, I. Smartphone Mode Recognition During Stairs Motion. Proceedings 2019, 42, 65.
- Wang, Q.; Ye, L.; Luo, H.; Men, A.; Zhao, F.; Ou, C. Pedestrian Walking Distance Estimation Based on Smartphone Mode Recognition. *Remote Sens.* 2019, 11, 1140. [CrossRef]
- 23. Jiang, G.X.; Wang, W.J. Correlation Analysis in Curve Registration of Time Series. J. Softw. 2014, 25, 2002–2017.
- 24. Elhoushi, M.; Georgy, J.; Noureldin, A.; Korenberg, M.J. Motion Mode Recognition for Indoor Pedestrian Navigation Using Portable Devices. *IEEE Trans. Instrum. Meas.* **2016**, *65*, 208–221. [CrossRef]
- Liu, K.; Wang, Y.; Wang, J. Differential Barometric Altimetry Assists Floor Identification in WLAN Location Fingerprinting Study. In *Principle and Application Progress in Location-Based Services*; Springer International Publishing: Cham, Switzerland, 2014; pp. 21–29.
- Zheng, L.; Zhou, W.; Tang, W.; Zheng, X.; Peng, A.; Zheng, H. A 3D indoor positioning system based on low-cost MEMS sensors. Simul. Model. Pract. Theory 2016, 19, 45–56. [CrossRef]
- 27. Yu, D.B. Pedestrian Behavior Pattern Recognition and Analysis of Indoor Location Data. Ph.D. Dissertation, Photogrammetry and Remote Sensing, Wuhan University, Wuhan, China, 2019.
- Schafer, P. The BOSS is Concerned with Time Series Classification in the Presence of Noise. *Data Min. Knowl. Discov.* 2014, 29, 1505–1530. [CrossRef]
- 29. Harrington, P. Machine Learning in Action; Manning Publications: New York, NY, USA, 2012; p. 33.
- 30. Jing, X.N.; Li, X.J. Application of Naïve Bayesian Method in Girl's Figure Discrimination. J. Text. Res. 2017, 38, 124–128.
- Chen, G.L.; Cao, X.X. Method of Pedestrian's Behavior Recognition Based on Built-in Sensor of Smartphone in Compartment Fires. J. Tongji Univ. (Nat. Sci.) 2019, 47, 414–420.
- Anguita, D.; Ghio, A.; Oneto, L.; Parra, X.; Reyes-Ortiz, J.L. Training computationally efficient smartphone-based human activity recognition models. In Proceedings of the 2013 Artificial Neural Networks and Machine Learning, Sofia, Bulgaria, 10–13 September 2013; pp. 426–433.
- Reyes-Ortiz, J.L.; Ghio, A.; Parra, X.; Anguita, D.; Cabestany, J.; Catala, A. Human activity and motion disorder recognition: Towards smarter interactive cognitive environments. In Proceedings of the 21th International European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning, Espoo, Finland, 24–26 April 2013; pp. 403–412.
- 34. Wu, Y.; Tian, X.S.; Zhang, Z.X. An Improved GRU-InFCN for Human Behavior Recognition Model. *Comput. Appl. Softw.* 2020, 37, 205–210.