sensors

MDPI

Article

A Novel Energy-Efficient Approach for Human Activity Recognition

Lingxiang Zheng [1,*], **Dihong Wu** [1], **Xiaoyang Ruan** [1], **Shaolin Weng** [1], **Ao Peng** [1] **, Biyu Tang** [1], **Hai Lu** [1], **Haibin Shi** [1] **and Huiru Zheng** [2]

[1] School of Information Science and Engineering, Xiamen University, Xiamen 361005, China;
    xmuwudh@stu.xmu.edu.cn (D.W.); ruanxiaoyang@stu.xmu.edu.cn (X.R.);
    23320141153268@stu.xmu.edu.cn (S.W.); pa@xmu.edu.cn (A.P.); tby@xmu.edu.cn (B.T.);
    luhai@xmu.edu.cn (H.L.); shihaibin@xmu.edu.cn (H.S.)
[2] School of Computing, Ulster University, Newtownabbey, CO Antrim BT37 0QB, UK; h.zheng@ulster.ac.uk
*   Correspondence: lxzheng@xmu.edu.cn; Tel.:+86-135-1596-7334

Abstract: In this paper, we propose a novel energy-efficient approach for mobile activity recognition system (ARS) to detect human activities. The proposed energy-efficient ARS, using low sampling rates, can achieve high recognition accuracy and low energy consumption. A novel classifier that integrates hierarchical support vector machine and context-based classification (HSVMCC) is presented to achieve a high accuracy of activity recognition when the sampling rate is less than the activity frequency, i.e., the Nyquist sampling theorem is not satisfied. We tested the proposed energy-efficient approach with the data collected from 20 volunteers (14 males and six females) and the average recognition accuracy of around 96.0% was achieved. Results show that using a low sampling rate of 1Hz can save 17.3% and 59.6% of energy compared with the sampling rates of 5 Hz and 50 Hz. The proposed low sampling rate approach can greatly reduce the power consumption while maintaining high activity recognition accuracy. The composition of power consumption in online ARS is also investigated in this paper.

Keywords: activity recognition; low power consumption; low sampling rate; energy-efficient classifier

1. Introduction

Human activity recognition plays a crucial role in pervasive computing. Many applications for healthcare, sports, security agencies and context-aware services applications have emerged [1,2]. For example, life logs collected by smart mobile phone sensors (such as accelerometers) have been used to provide personalized health care [3]. Vermeulen et al. [4] developed a smartphone-based falls detection application to help elderly people. Zhou et al. [5] implemented a phone system for indoor pedestrian localization. Google Now is one of the emerging smart applications that provide context-aware services. It calculates and pushes relevant information automatically to mobile users based on their current locations [6].

The history of human activity recognition can be traced back to the late 1990s [7]. Four sensors (accelerometers) were placed on different positions of body to detect human activities (lying, sitting, sitting/talking, sitting/operating, standing, walking, upstairs, downstairs, and cycling). Randle and Muller [8] used a single wired biaxial accelerometer to classify six activities (sitting, standing, walking, running, upstairs, and downstairs) in 2000. However, the early systems are not easy to use.

Thanks to the development of microelectronics and computer systems, the sensors and mobile devices are now with higher computational capability, smaller size and more acceptable usability. The studies on activity recognition systems (ARS), especially smartphone based activity recognition system (ARS), have been set off a booming in recent years [9–13]. The accelerometers and gyroscopes

embedded in smart phones have been used to collect raw activity data in ARS. Smart phones have become one of the most indispensable parts of life when comparing with other special devices [13,14]. It is now a relative low-cost device for both developers and users.

The smart phone based ARS can be divided into two types. One is online activity recognition systems, i.e., data collection, data processing and classification are carried out locally on the mobile phones [11,15]. The other is offline activity recognition systems, i.e., the classification is carried out non-real-time, or offline. Similar to other research [16], we consider an ARS in which the classification is carried out in a remote server or cloud as an offline ARS because the classification becomes non-real-time when the phone has no internet connection.

An online ARS can recognize the user's behavior and provide the feedback in real time to support user's daily life [17]. A number of studies on online ARS have been carried out. Anjum et al. [18] developed an application for recognizing a number of activities, including driving and cycling, with an average accuracy of greater than 95%. Kose et al. [15] investigated the performance of different classifiers and used accelerometer of the smartphone to classify four activities (sitting, standing, walking, and running). Schindhelm et al. [19] explored the capability of using smartphone (HTC hero) sensors for the detection of steps and movement/activity types. Martín et al. [20] presented the work of using smartphone for activity recognition without interfering user's life.

Although the previous research work has achieved good accuracy in activity detection, there are few reports on power consumption. The power consumption is one of the main challenges [11], especially for the online ARS. The mobile phones are usually used for making phone calls or Internet surfing, so the power consumption of the online ARS must be reduced as low as possible. In order to solve this issue, one straightforward method is to reduce the number of sensors—for example, turning off the Global Positioning System (GPS) while the user is indoors or applying some energy-efficient sensors (such as accelerometers, gyroscopes) [10,21,22] instead. The other approach is to lower the sampling rate. However, in most studies [13], the sampling rates were still high because they followed the Nyquist theorem, i.e., the sampling rate must be equal to or higher than twice the signal frequency so that no actual information will be lost during the sampling process.

In this paper, we propose an energy-efficient ARS that uses low sampling rate and can still achieve high accuracy. A theoretically proof of the rationale of using low sampling rate in ARS is presented. A novel classifier is also proposed and developed to improve the performance of activity recognition. The proposed system consists of three components: (a) sensors using the proposed low sampling rate for data collection; (b) feature extraction for training and classification; (c) the proposed classifier which integrates hierarchical support vector machine (H-SVM) with context-based classification (HSVMCC) to detect user's activities.

The rest of the paper is organized as follows. In Section 2, we briefly describe the related work. Section 3 details the proposed energy-efficient system, including data collection using low sampling rate, feature extraction method, the proposed HSVMCC algorithm, and the composition of power consumption in online ARS. The discussion of experiments and results is presented in Section 4. The paper is concluded by the summary of merits, limitations and future work in Section 5.

2. Related Work

The common approach of saving energy consumption in online ARS is to detect the mobile status and user's location or activities and then turn on/off some unused sensors [22–24]. Wang et al. [24] presented a novel design framework for an Energy Efficient Mobile Sensing System (EEMSS). Using a hierarchical sensor management strategy to recognize user states and detect state transitions, the EEMSS significantly improved battery life of the device.

Adopting one tri-axial accelerometer, Lius et al. [13] detected activities (including walking, jumping, immobile, running, up, down, cycling and driving) using the sampling rate varying from 32 Hz to 50 Hz. An average accuracy of 98% was achieved. Discrete variables were used to reduce

the calculation costs and save the energy. However, the sampling rate is still too high to maximize energy savings.

Reddy et al. [25] employed the GPS and accelerometer to detect activities (including stationary, walking, running, biking or motorized transport). Although the sampling rate was set to 1 Hz only, the GPS is not an energy-efficient sensor and cannot be used indoors.

Similarly, aiming to detect human mobility states, Oshin et al. [10] used an accelerometer at a sampling rate of 4 Hz. The results showed an overall average accuracy of 92%. However, it failed to detect some regular indoor activities, such as climbing stairs, in contrast to other studies [12,18,26]. The rationale of using the sampling rate of 4 Hz was not presented in the paper.

Applying a tri-accelerometer at the sampling rate of 2 Hz, Liang et al. [27] managed to obtain the average accuracy of 89% detecting human activities (standing, sitting, lying, walking, running, jumping, ascending, descending, cycling and driving). However, no justification was provided for the choice of sampling rate. It also lacked the accurate evaluation of power performance.

Activity recognition plays an important role in the area of pervasive healthcare. Liang et al. [28] proposed a hierarchical method to recognize user activities based on a single tri-axial accelerometer in smart phones for health monitoring.

Li et al. [29] proposed to leverage machine learning technologies for improving the energy efficiency of multiple high-energy-consuming context sensors by trading off the sensing accuracy.

For the purpose of utilizing available energy efficiently while achieving a desired activity recognition accuracy, Zappi et al. [30] investigated the benefits of dynamic sensor selection. It introduced and characterized an activity recognition method with the help of an underlying run-time sensor selection scheme.

Mortazavi et al. [31] presented a multiple model approach to classifying movements in exergame environment with fine-grain motions. Expert knowledge was applied to identify similar movements. Each submodel was modeled using a one to many support vector machine (SVM) with nonlinear kernel. Although the multiple model approach achieved a good classification performance, the study didn't consider the power consumption either in algorithms or in data sampling, where a sampling rate of 50 Hz was used.

In our previous work [32], we have experimentally tested and confirmed that the sampling rate of 1 Hz could achieve high performance for detecting activities (including sitting, standing, walking, and running) in an offline ARS.

Considering the above-related work, we have carried out a theoretical analysis on why using a low sampling rate in ARS can also achieve high performance. Experiments based on the smartphone have also been undertaken to evaluate the power consumption of the online ARS with different sampling rates. Furthermore, the recognition of climbing stairs activities (upstairs and downstairs) is also included in the proposed ARS.

3. Energy-Efficient Activity Recognition System

The aim of this research is to build a user-independent and energy-efficient online ARS with high accuracy. The proposed system, as shown in Figure 1, includes data collection, feature extraction, and the training and classification (H-SVM and context-based classification). The system does not contain data processing before the feature extraction because the data obtained have been preprocessed by the phone's built-in filters.
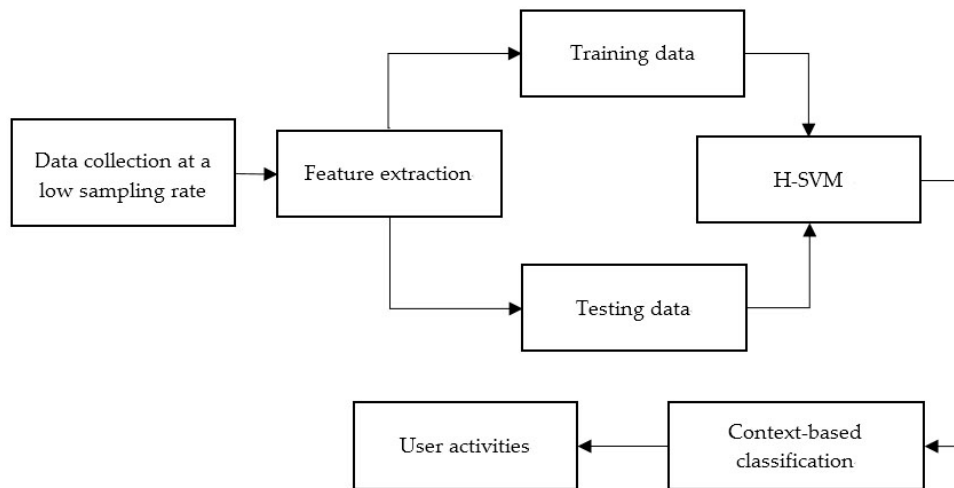
Figure 1. System architecture.

3.1. Data Collection at a Low Sampling Rate

The types of sensors and the sampling rates are two factors that must be considered during data collection in ARS. The barometer, accelerometer and gyroscope are the sensors usually used in ARS.

Barometer is the sensor for measuring altitude or height. Using low sampling rate of barometer can detect whether the user is climbing stairs or not.

Inertial measurement units (IMUs), such as accelerometer and gyroscope, are used to measure the user motion. In previous studies [1], the sampling rate of these sensors (such as accelerometers) in ARS was set between 10 Hz and 100 Hz. It is a general view that a high sampling rate can avoid the information loss of signals. Some research also claimed that high sampling rate could achieve high accuracy of recognition [33]. However, the higher the sampling rate is, the more energy consumed. The trade-off between the sampling rate and the power consumption has become one important concern in most energy-efficient ARS.

In our research, we proposed a solution to solve the contradiction between the sampling rate and the power consumption, that is, using a low sampling rate of IMU in ARS to achieve a similar recognition accuracy, compared with using high sampling rates.

For human activity recognition, the purpose of sampling is not to restore the raw signals of activities, but to detect different activities according to the statistical properties of signals, such as means, variance, and maximum. It is considered that using high sampling rate can capture all the details of the person's movements, and this would benefit the recognition of activity [34]. However, the signal information would not be lost if the sampling rate agreed with the Nyquist theorem, which means that the statistical properties of using low or high sampling rate are consistent. When the sampling rate is less than the frequency required by the Nyquist theorem, we suggested that adding more sampling periods can acquire the consistent statistical properties, which is demonstrated as follows:

Set the frequency of activity as F_a, the sampling rate as F_S, and the sampling period as T.

For different sampling rates of F_{S_1} and F_{S_2}, if they agree with the following conditions:

$$F_{S_1} \% F_{S_2} \neq 0 \tag{1}$$

There exists Equation (2):

$$F_{S_1} \times T_1 = F_{S_2} \times T_2$$
$$T_1 \in \{1, 2, \ldots, n\}, F_{S_1} \times T_1 \in \{1, 2, \ldots, n\} \tag{2}$$

where T_1 and T_2 are sampling periods.

The elements of dataset $X_1 = \{x_1, x_2, \ldots, x_n\}$ obtained at the sampling rate of F_{S_1} and sampling period of T_1 are the same with dataset $Y_1 = \{y_1, y, \ldots, y_n\}$ obtained at the sampling rate of F_{S_2} and sampling period of T_2. Thus, the statistical properties of the dataset X_1 and the dataset Y_1 are the same, if the sampling period is long enough.

If sampling rate F_{S_3} is less than the frequency required by the Nyquist theorem F_a, which is:

$$F_{S_3} < 2F_a \tag{3}$$

Human activity signal is a non-strict period, so F_a is not a determined value, but a fluctuating value. The relation between $2F_a$ and F_{S_3} satisfied the Equation (1). The same as above, the elements of dataset $D = \{d_1, d_2, \ldots, d_n\}$ obtained at the sampling rate of $2F_a$ and the sampling period of T has the same statistical properties as the dataset $D' = \{d'_1, d'_2, \ldots, d'_n\}$ obtained at the sampling rate of F_{S_3} and the sampling period of T_3.

Combined with the formulas above, time period T_3 is calculated as Equation (4):

$$\frac{T_3 = (2F_a \times T)}{F_{S_3}} \tag{4}$$

Therefore, when we use a low sampling rate that does not agree with the Nyquist theorem in ARS, we can add the sampling time to ensure the same statistical properties.

3.2. Hierarchical Support Vector Machine (H-SVM)

Support vector machine (SVM) is a supervised learning algorithm. The basic SVM model is the probability of a binary classification. In order to deal with multiple classes, Liu et al. [35] proposed an adaptive hierarchical multi-class SVM classification scheme at the training stage.

In this paper, the *k*-means clustering algorithm was used in training the H-SVM classifier.

The training algorithm is summarized in Algorithm 1.

Algorithm 1 Training algorithm

1: Construct a feature set ($\{f_1, f_2, \ldots, f_n, (f_{i_1}, f_{j_1}), (f_{i_2}, f_{j_2}), \ldots, (f_1, f_i, \ldots, f_m)\}$). The feature set is the combination of all features.

2: Sorting features according to the power consumption of the sensor and the computational cost of feature extraction. The lower power consumption of the sensor ranks the higher. Features with lower computational cost have in higher priority when the sensor is the same.

3: Selecting *m* top features with the higher priority from the feature set.

4: For each of these *m* features, set the whole dataset as DataSets_In.

5: Input DataSets_In. do

6: {

7:      Set parameter of clustering $k = 2$

8:      Use *k*-means clustering algorithm to get two cluster A and B. The two clusters are the subsets of DataSets_In. Evaluate the performance of *k*-means clustering, select the optimal ones according the accuracy and the equilibrium of the two subset.

9:      Input subsets A and B

10:     Training a binary classifier SVM

11:     SVM_*n* = SVM (n = 1, 2, 3 … …)

12:     DataSets_In = subsets A or B respectively

13: } while DataSets_In only contains data from a single class

14: The resulting classifier (i.e., final classifier) is a one-node SVM classifier, as an *N*-class classification needs an *N-1* node classifier.

3.3. Context-Based Classification

During the study, we found that there are two reasons that some errors may occur in the activity recognition: (1) features were similar between two activities; (2) some measurement errors. To correct these errors, we proposed a context-based classification approach. Context-based classification is a method that combining the previous variables' information and the following variables' information during the analysis of the current variable. It can effectively eliminate the individual errors.

Human activities are continuous processes. Therefore, the previous recognition results and the following recognition results can be used to check and correct the current recognition result. The process is archived by using a sliding window (the window length is *2k + 1*) to correct the result at time *t*, as shown in Figure 2.
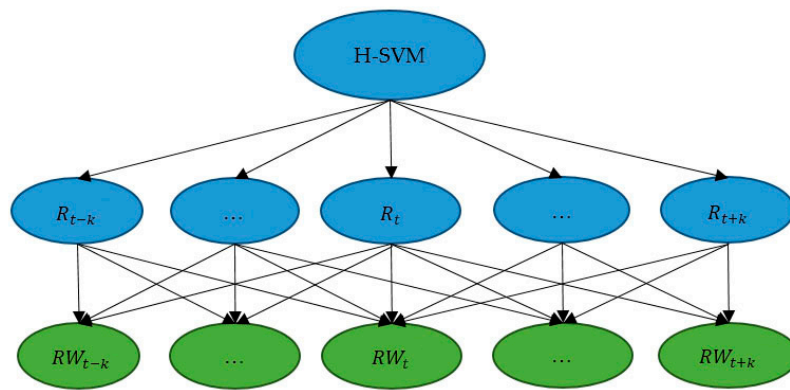

Figure 2. The context-based classification.

In Figure 2, the variable R_t is the result of the H-SVM model at time t, and RW_t is the corrected result after context-based classification. RW_t is defined as the mode (the most frequent result) among $\{R_{t-k}, R_{t-k+1}, \ldots, R_{t+k}\}$. We assume that the probability of recognition errors ψ is independent identically distributed, the accuracy at time *t* ($Accuracy_t$) with values of *k* is showed as follows:

$$Accuracy_t = (1 - \psi)^{2k+1} + C_{2k+1}^1 (1 - \psi)^{2k} \psi + \ldots + C_{2k+1}^{2k} (1 - \psi) \psi^{2k} + \psi^{2k+1} \tag{5}$$

The algorithm is summarized in Algorithm 2:

Algorithm 2 Context-based classification Algorithm

1: Initialize the data buffer $\{Result_1, Result_2, \ldots, Result_{2k+1}\}$
2: while R is input from H-SVM do
3: {
4:    If $n < 2k + 2$ then
5:      $Result_n = R(n = 1, 2, \ldots, 2k + 1)$
6:      $RW_T = Rt$
7:    else
8:      $\{Mode_1, Mode_2, \ldots\} = MajorityOf\{Result_1, Result_2, \ldots, Result_{2k+1}\}$
9:        if $Result_{k+1} \in \{\{Mode_1\}, \{Mode_2\}, \ldots\}$ then
10:          $RW_t = Result_{k+1} = R_t$
11:        else
12:          $RW_t = Mode_1$
13: }
14: end while

Firstly, variables $\{Result_1, Result_2, \ldots, Result_{2k+1}\}$ are initilised as data buffer. Then, for each R received, it is placed to the data buffer $\{Result_1, Result_2, \ldots, Result_{2k+1}\}$. For the first R (which equals to R_{t-k}) input from H-SVM, the RW_t is set as R_t. For the following R, RW_t does not change. When the $2k + 1$ of R are all input, the majority $\{Mode_1, Mode_2, \ldots\}$ of the data buffer is calculated. If $Result_{k+1}$ (here, $Result_{k+1}$ equals to R_t) belongs to $\{Mode_1, Mode_2, \ldots\}$, and the RW_t is set as $Result_{k+1}$; otherwise, the RW_t is set as $Mode_1$. Then, the data buffer is shifted right by one to store the next R_t. For the next RW, the steps of calculating RW are the same as RW_t. The algorithm is stopped when R is no longer received.

Time delay is the main defect of the context-based classification algorithm. It is equal to the value of k. It increases with the growth of the sliding window size. Thus, it is important to choose a suitable sliding time window size (that is, the values of $2k + 1$) in an online ARS.

4. Experiments and Discussion

The experiments include four parts: (1) data collection; (2) parameter selections; (3) classification performance; and (4) the power consumption of the proposed energy-efficient online ARS.

4.1. Data Collection

For data collection, a smartphone (Nexus 5, Google Inc., Mountain View, CA, United States of America) was placed in the right-front pocket of the pants, showed in Figure 3. The sensors used in the experiments were barometer and accelerometer inside the smartphone. As shown in Table 1, four independent data collections were carried out separately in the study.

Figure 3. Illustration of the placement of the phone on the participant's body.

Table 1. Data collection of four experiments.

Data Type	Number of Volunteers	Sampling Rate	Time Window	Overlap Time
Data collection 1	1 (male, 23 years old)	1 Hz	5 s	0 s
Data collection 2	20 (14 males, 6 females, 22–25 years old)	1 Hz	5 s	0 s
Data collection 3	5	5 Hz	1 s	0 s
Data collection 4	5	10 Hz/50 Hz	1 s	0 s

Data collection 1 (Training datasets): the sampling rate was set to 1 Hz, the time window was 5 s, without overlap. One volunteer (male, 23 years old, healthy student) was asked to perform six types of activities: sitting, standing, walking, running, climbing upstairs and going downstairs. The volunteer sat and stood indoors, walked in the corridor or in the room, ran on a treadmill at 9 km/h, climbed upstairs and went downstairs in our lab building, which is six-floors. Each activity was carried out 15 times (15 samples collected). In total, 90 samples were collected as the training data-sets.

Data collection 2: The sampling rate was set to 1 Hz, the time window was 5 s, without overlap. Twenty volunteers (14 males and six females, ages between 22 to 25) participated in the data collection. The volunteers were asked to undertake the six activities as described in the Data collection 1. Each activity (sitting, standing, walking and running) was consecutively carried out for 5 min. The volunteers were asked to climb upstairs from the second floor to the sixth floor and go downstairs from the sixth floor to the second floor, five times repeatedly. All these data collected in Data collection 2 are only used for testing, as shown in Table 2. For each activity, we removed the data of the first time window (5 s) and the last time window (5 s) to ensure that the data obtained only contained one type of activity.

Table 2. Testing datasets collected from 20 participants (Data collection 2).

Activity Type	Phone Model	Number of Data Time Windows
Sitting	LG Nexus5	1234
Standing	LG Nexus5	1246
Walking	LG Nexus5	1375
Running	LG Nexus5	1154
Upstairs	LG Nexus5	1324
Downstairs	LG Nexus5	1256

To compare the accuracy of activity recognition at different sampling rates, five volunteers (from the above 20 volunteers in the Data collection 2) participated again in the following two new data collections.

Data collection 3: The purpose of this data collection is to verify that using a low sampling rate (1 Hz) can also achieve high accuracy of recognition, compared with the sampling rate agreed with the Nyquist theorem. Thus, the sampling rate was set as 5 Hz, which agreed with the Nyquist theorem and was close to twice the frequency of human activity obtained by phone sensor. The time window was 1 s.

Data collection 4: The aim of this data collection is to verify that different sampling rates that agreed with the Nyquist theorem achieve almost the same accuracy. Activity data were collected at the sampling rates of 10 Hz and then 50 Hz. The time window of different sampling rates was 1 s.

4.2. The K-Means Clustering and H-SVM

In general, the feature extraction aims to identify the main characteristics that accurately represented the original data [36]. The process is to find the most useful, valid and meaningful information to recognize activities with high accuracy. In previous studies [1,10,13], the common features include time domains and frequency domains, such as means, standard deviation, magnitude of acceleration and FFT (Fast Fourier Transform). There are no fixed features that are suitable for all ARS.

In this paper, we firstly constructed a feature set. The feature set is the combination of all features, that is P_d (pressure difference), $P_{d_{abs}}$ (absolute value of pressure difference), $X_{means}, Y_{means}, Z_{means}$ (the means of X-/Y-/Z-axis accelerometer values), and T_{waves} (the sum of root mean squares of the difference of adjacent points in a time window).

After constructing the feature set, algorithm 1 was applied to feature selections and classification. Based on the power consumption of the sensor and the computational cost of feature extraction, m features ($P_d, P_{d_{abs}}, Y_{means}, T_{waves}$) with the higher priority from the set of optimal features were selected.

(1) Pressure difference Pd: The difference of pressure value is measured by barometer built-in the mobile phone, as shown in Equation (6). The barometer value is considered as height changing. When the altitude increases, the pressure value decreases, and vice versa:

$$P_d = p_n - p_0 \tag{6}$$

where p_n is the last pressure value and p_0 is the first pressure value in the time window (sampling period). The P_d value is negative when the user climbs upstairs, and it is positive when going downstairs.

(2) The absolute value of pressure difference ($P_{d_{abs}}$): The $P_{d_{abs}}$ is calculated as follows:

$$P_{d_{abs}} = abs(P_d) \tag{7}$$

(3) *X*-/*Y*-/*Z*-axis accelerometer value ($X_{means}, Y_{means}, Z_{means}$): This is the means of the *X*-/*Y*-/*Z*-axis accelerometer values. The values of tri-axial accelerometer we got from the smartphone (Android API) contained the gravity values. The following is the calculation for $X_{means}, Y_{means}, Z_{means}$:

$$\begin{aligned} X_{means} &= \frac{x_1+x_2+x_3}{n} \\ Y_{means} &= \frac{y_1+y_2+y_3}{n} \\ Z_{means} &= \frac{z_1+z_2+z_3}{n} \end{aligned} \tag{8}$$

(4) The wave of three-axis accelerometer (T_{waves}): this is the sum of the RMS (Root Mean Square) of the difference of adjacent points in a time window, and can be calculated using Equation (9):

$$T_{wave} = \sum_{i=0}^{n} \sqrt{\left(AccX_{i+1} - AccX_i\right)^2 + \left(AccY_{i+1} - AccY_i\right)^2 + \left(AccZ_{i+1} - AccZ_i\right)^2} \tag{9}$$

where $AccX_i, AccY_i, AccZ_i$ are the three-axis values of accelerometer at time stamp *i*, respectively.

The training carried out on the whole dataset (Data collection 1) using algorithm 1. As shown in Figure 4, for the feature P_d (Figure 4a,b), the whole training dataset was divided into subset A (downstairs) and B (upstairs, sitting, standing, walking, running). For the feature $P_{d_{abs}}$ (Figure 4c,d), the whole training dataset was divided into two subsets A (upstairs and downstairs) and subset B (sitting, standing, walking, running) using the *k*-means clustering algorithm. For the feature Y_{means} (Figure 4e,f), the whole training dataset was divided into subset A (sitting) and B (downstairs, upstairs, standing, walking, running). For the feature T_{waves} (Figure 4g,h), the whole training dataset was divided into subset A (running) and B (downstairs, upstairs, sitting, standing, walking).

As shown in Figure 4, the accuracy and partition degree of *k*-means clustering of different features were assessed. The equilibrium of two subsets was also considered. Figure 4b,d,f shows the results that *k*-means clustering can get good performance using the selected features, but only the feature $P_{d_{abs}}$ can meet the equilibrium requirement. Thus, the selected feature $P_{d_{abs}}$ is the optimal feature for training the first SVM classifier (SVM1). In the Algorithm 1, the data were randomly divided into 80% for training and 20% for testing in order to select the optimal features and build the SVM classification models.
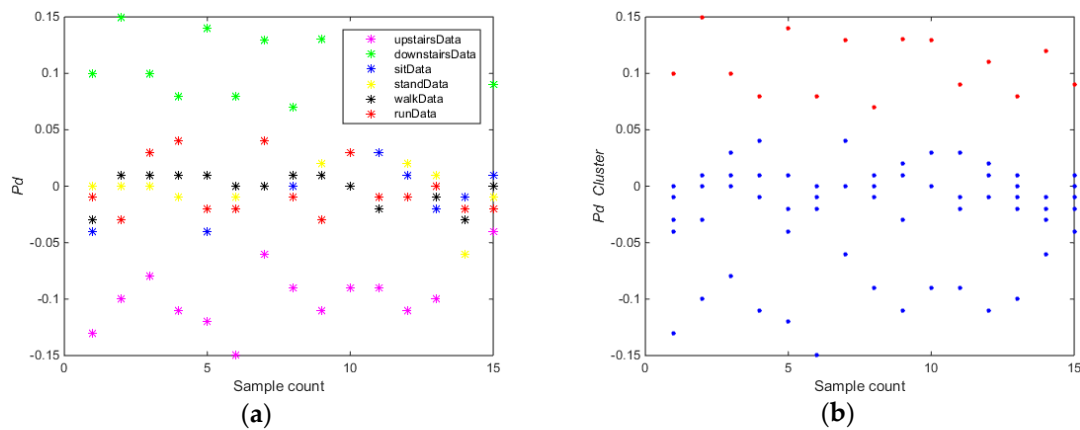

(a)   (b)
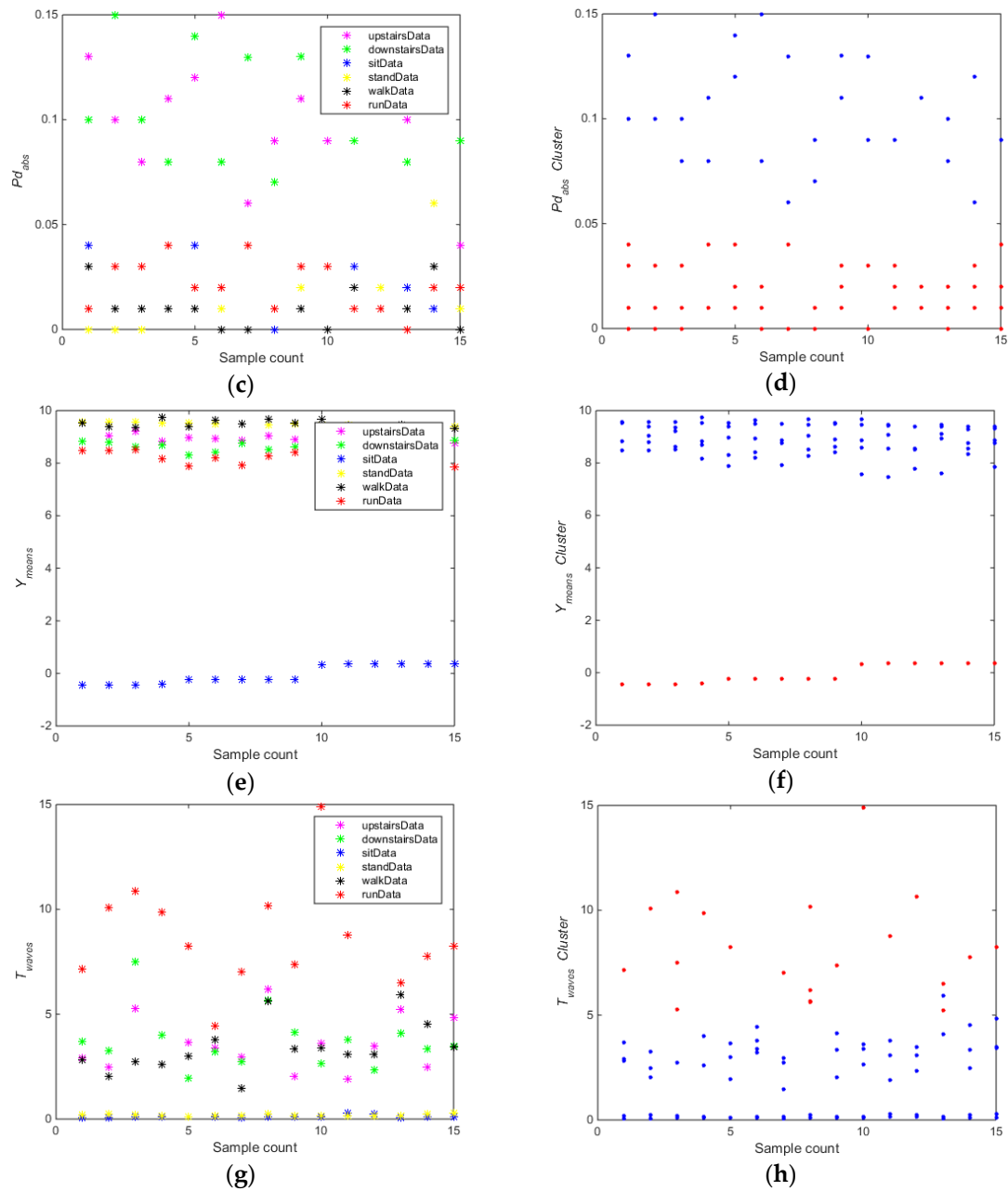
Figure 4. *Cont.*

Figure 4. (**a**) P_d value; (**b**) results of *k*-means clustering of P_d feature; (**c**) $P_{d_{abs}}$ value; (**d**) results of *k*-means clustering of $P_{d_{abs}}$ feature; (**e**) Y_{means} value; (**f**) results of *k*-means clustering of Y_{means} feature; (**g**) T_{waves} value; (**h**) results of *k*-means clustering of T_{waves} feature.

According to Algorithm 2, for subset (upstairs or downstairs), feature P_d is the optimal feature for training SVM classifier (SVM2) to partition the upstairs or downstairs because it is the most accuracy ones. For subset (sitting, standing, walking, running), feature Y_{means} is suitable for classification (SVM3), dividing dataset (sitting, standing, walking, running) into subset (sitting) and subset (standing, walking, running) with the highest accuracy. In addition, Y_{means} is also the optimal feature for dividing dataset (standing, walking, running) into subset (standing, walking) and running (SVM4). Finally, the T_{waves} is used for partition standing and walking (SVM5). The whole H-SVM classification model is shown as follows.

The whole training dataset (Data collection 1) contains 90 samples. After training, the five-node SVM classifier was built. As illustrated in Figure 5, the dataset is divided into two sets whether the

activity is climbing stairs or not. If the classification result of classifier SVM1 is climbing stairs, classifier SVM2 is used to judge if the activity is climbing upstairs or going downstairs. If the result of SVM1 is not stair climbing, classifier SVM3 is applied to classify sitting or standing, walking, running and then classifier SVM4 will contribute to recognize standing, walking or running. Finally, classifier SVM5 is used to differentiate standing or walking. Considering the *k*-means clustering results discussed before, $P_{d_{abs}}$ can be used as the input feature for SVM1 to detect climbing stairs or not. Furthermore, P_d, T_{waves} can be used as input feature for SVM2 and SVM5, respectively, and Y_{means} can be input as feature for SVM3 and SVM4. These may reflect the body movement efforts and acceleration patterns when carrying out different types of activities, that is: (1) the pressure used in climbing stairs activities is different from activities on flat ground; (2) changing of acceleration on the *Y*-axis can be used to differentiate the sitting status from standing/walking/running and further differentiate running from walking/standing; (3) information of changes in all three directions needs to be taken into account in order to classify standing from walking.
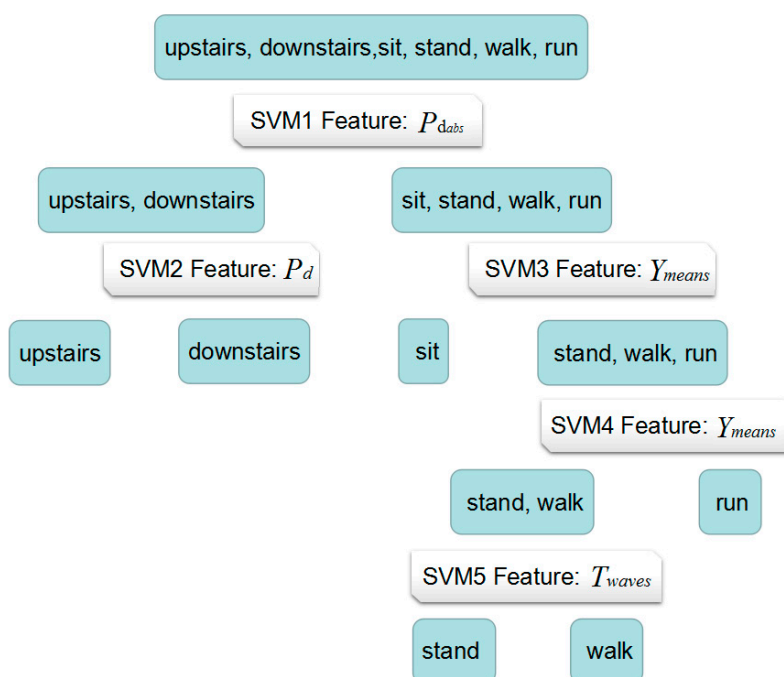

Figure 5. The whole H-SVM classification model.

4.3. The Parameter Settings of Proposed ARS.

The sampling rate and time window of accelerometer during data collection and sliding window size of context-based classification are three crucial parameters that may affect the power consumption and accuracy of proposed ARS.

The frequency of human activity is about 2 Hz. For example, the frequency of going downstairs with fast speed is less than 2 Hz, and the step time of fast walking is 0.35 s/step [37]. The time windows were usually 1 s in previous studies [38]. In our research, the sampling rate of accelerometer was 1 Hz. According to Equation (4), the time window was about 5 s.

In Section 3.3, we proposed context-based classification to improve the accuracy of recognition. For different values of the probability of recognition error ψ (0.3, 0.2, 0.1, and 0.05), the $Accuracy_t$ of different sliding window size ($2k + 1$) is shown in Figure 6:
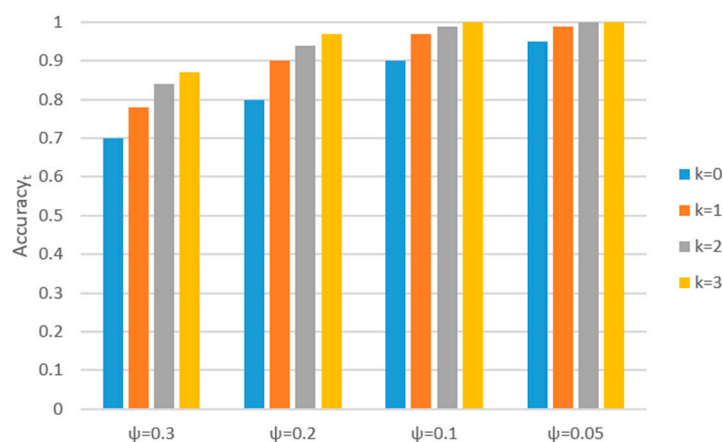
Figure 6. The $Accuracy_t$ of different sliding window length k.

Figure 6 shows that the $Accuracy_t$ has improved with the increase of k-values. For example, the $Accuracy_t$ has improved 8% with the change of value k from 0 to 1 when $\psi = 0.3$. However, the time delay will also increase, which may be harmful to the online ARS. Especially when the recognition error ψ is becoming closer to 0, with the increase of value k, the improvement of $Accuracy_t$ is becoming smaller, but the time delay is becoming greater.

The classification performance of our proposed ARS shows that the largest recognition error ψ is less than 0.2 and the average recognition error is less than 0.1. As shown in Figure 6, no matter $\psi = 0.2$ or $\psi = 0.1$ or $\psi = 0.05$, the accuracy is improved quickly when k is increased from 0 to 1. However, the improvement of $Accuracy_t$ slows down when $k \geq 1$, but the time delay became greater. Therefore, the slide window size is set as 3 ($k = 1$).

4.4. The Classification Performance of Proposed ARS

In this section, we assess the performance of proposed classifier and compare it with other classifiers. The classification accuracy of different sampling rates is also discussed.

4.4.1. Performance of Different Classifiers

In our research, we used the H-SVM model and context-based classification. In order to analyze the performance of H-SVM, we used the training dataset (Data collection 1), testing dataset (Data collection 2) and features obtained from the mobile phone. The training and classification were carried out in Matlab 2014a (MathWorks Inc., Natick, MA, USA), using Libsvm library [39]. The training used the linear kernel, cost and without cross-validation. The features of H-SVM and the parameters of SVM used in Matlab were the same as those on the phone. The classification results are shown in Table 3.

Table 3. The performance of the H-SVM classification.

Activity	Accuracy
Sitting	97.6%
Standing	94.5%
Walking	91.1%
Running	89.1%
Upstairs	83.8%
Downstairs	89.7%
Average accuracy	90.9%

As shown in Table 3, the average accuracy of six activities is 90.9% and the weakest performance (the accuracy is only 83.8%) occurred when recognizing climbing upstairs. Upon close examination,

we found that it was caused by the noise of the signals, which led to the misclassification in some discrete time windows. We randomly selected 200 continuous recognition results of climbing upstairs shown in Figure 7. In Figure 7, it can be seen that some activities of climbing upstairs were misclassified as other activities, such as standing, walking and running.
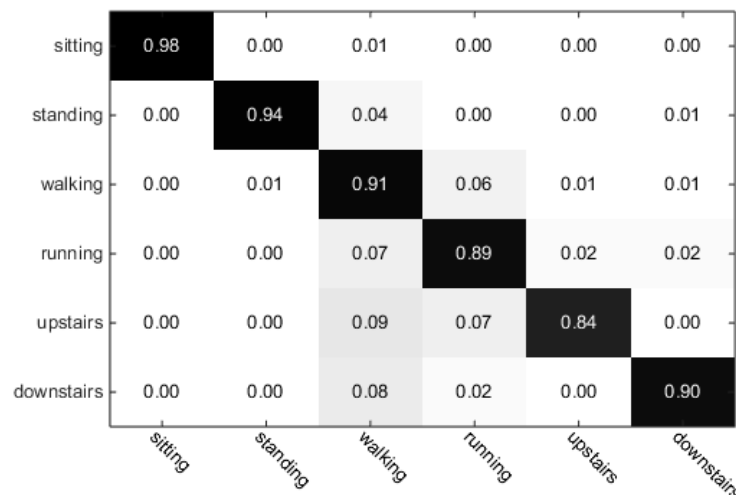

Figure 7. The classification results of climbing upstairs activity after H-SVM.

To reduce the impact of the noise and to improve the accuracy, we applied context-based classification after H-SVM. The results are shown in Table 4. The process of data collection, processing, training, and classification are all done by the phone. The accuracy values of six activities are increased by 1.8%, 3.1%, 5.5%, 4.7%, 8.3% and 6.6%, respectively, and the average accuracy of six activities is increased by 5.1%. The average accuracy of six activities of the proposed ARS is 96.0%, which is high enough for most applications.

Table 4. The performance of H-SVM and Context-based classification (HSVMCC).

Activity	Accuracy
Sitting	99.4%
Standing	97.6%
Walking	96.6%
Running	93.8%
Upstairs	92.1%
Downstairs	96.3%
Average accuracy	96.0%

We compared our method with other classification algorithms such as J48 Naive Bayes (NB) and Random Forest (RF). The machine learning tool weka [40] was used in the study and the results are shown in Figure 8. We used the same training datasets (Data collection 1) described before to obtain the model of other classification algorithms. The universal parameters were selected for these classification algorithms. For J48, the parameter C was set as 0.25 and M was set as 2. For Random Forest, the parameter I was set as 100, K was set as 0 and S was set as 1. Then, we used all testing datasets (Data collection 2) in Table 2 to test the classifiers.
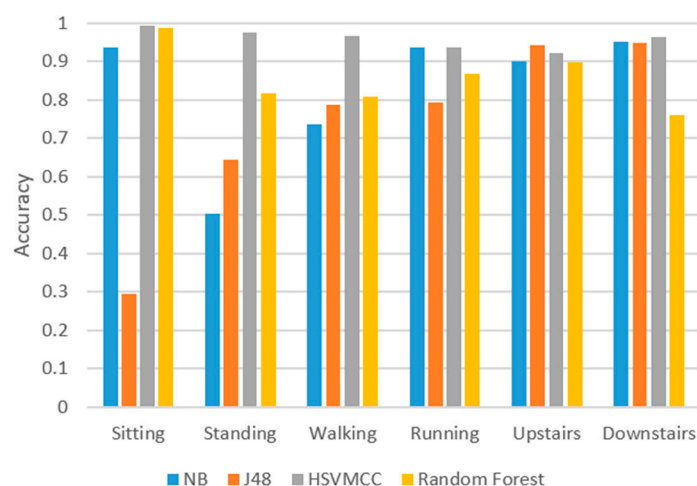
Figure 8. Comparison of the proposed ARS vs. classification models, J48, NB and Random Forest.

Figure 8 shows that the accuracies of the proposed method (HSVMCC) are more than 90% for all six activities. However, the accuracies of other algorithms vary between different activities. For sitting, the Random Forest (RF) achieves a high accuracy of 98.9%, while J48 obtained the lowest accuracy of 29.5%, but, for the 'going downstairs' activity, the accuracy of J48 is 94.8%, while the accuracy of Random Forest only achieves 76.1%.

Figure 9 shows the average accuracy of six activities of HSVMCC in comparison to Naive Bayes (NB), J48, and Random Forest (RF). The average classification accuracy for HSVNCC, NB, J48 and RF are 96%, 82.6%, 73.9%, and 85.6%, respectively. It can be concluded that the proposed HSVMCC outperformed other classifiers in terms of the average accuracy.
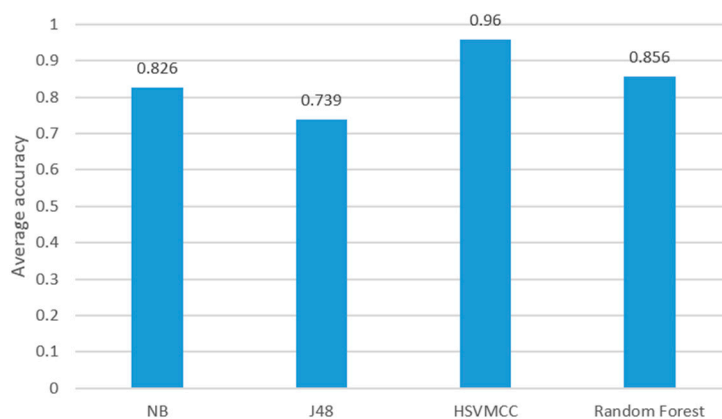

Figure 9. The comparison of average accuracy of the proposed AR system vs. other classifications.

4.4.2. The Accuracy of Different Sampling Rates

As mentioned before, we can use the sampling rate, which is less than the frequency required by the Nyquist theorem. Figure 10 shows the recognition results of ARS using the sampling rate of 1 Hz (less than the frequency required by the Nyquist theorem), 5 Hz (agreed with the Nyquist theorem), 10 Hz and 50 Hz. It can be observed that the accuracy of using 1 Hz sampling rate and using 5 Hz sampling rate are comparable, or similar. This means that, if the sampling rate is less than the frequency required by the Nyquist theorem, we can add the sampling period to achieve the similar accuracy of using the higher sampling rate that agrees with the Nyquist theorem.
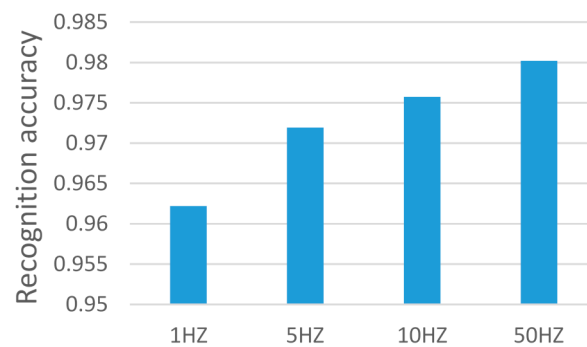
Figure 10. Recognition accuracy of activities at different sampling rates.

Figure 10 also shows the accuracy has only improved slightly with the increase of the sampling rate from 1 Hz to 50 Hz, i.e., (1 Hz: 96.2%, 5 Hz: 97.2%, 10 Hz: 97.6%, 50 Hz: 98.0%). The accuracy of 1 Hz (96.2%) is sufficiently high for practical applications.

4.5. The Power Consumption of the Energy-Efficient ARS

The research about the compositions of energy consumption in ARS can help us to assess whether the proposed energy-efficient strategies are effective or not. Furthermore, the analysis of the composition of energy consumption in ARS can provide guidance for the researchers in energy-efficient fields.

An online ARS consists of data collection, data processing and activity recognition. Thus, the main composition of energy consumption in ARS can be divided into three parts. The first part is the power consumption used by the sensors. In our research, this part does not contain the data collection. The second part is the power consumption used in data processing, including data collection, feature extraction and data storage. The last part is the power consumption used by the activity recognition algorithm.

In our previous work [32], we proposed that the low sampling rate can decrease the power consumption. Power consumption for ARS is caused by the sensor running [10] or the total power consumption [13]. In this paper, we carried out the experiments to analyze the composition of power consumption in ARS.

We use the other mobile phone (Nexus 5, with an Android 4.4.2 system) for experiments. Firstly, we restored the phone to factory data to avoid power consumption caused by other applications, and we installed the requiring applications in the phone. Then, we put the phone in a shaker to do the experiments.

The experiments can be divided into two categories.

Category 1: we carried out the experiments in the shaker with the setting of 5 mm amplitude and 5–10 Hz variant-frequency vibration and a total of 17 experiments were undertaken.

Category 2: we carried out the experiments in the shaker with the static state and a total of 17 experiments were undertaken.

The purpose of contrasting two states (shaker and static state) is to simulate the real situations. We used the shaker to simulate the status of moving such as walking. Similarly, the static state was used to simulate standing and sitting status.

For each category, we conducted four experiments (sampling rate of 1 Hz, 5 Hz, 10 Hz, 50 Hz respectively) with the setting of running the whole ARS, four experiments with the setting of only running sensors, four experiments with the setting of running the ARS without activity recognition and result processing, four experiments with the setting of running the ARS without result processing and one experiment when the phone was on standby. The details are listed in Table 5.

Table 5. Experiment setting of each case of study.

Placing State	Other Experiment Settings
Static	Running the whole ARS
Static	Only running Sensors
Static	Running the ARS without activity recognition and result processing
Static	Running the ARS without result processing
Static	The phone on standby
Shaker	Running the whole ARS
Shaker	Only running Sensors
Shaker	Running the ARS without activity recognition and result processing
Shaker	Running the ARS without result processing
Shaker	The phone on standby

For each experiment, we fixed the mobile phone in the shaker (Figure 11a) and connected an external signal generator (3.8 V) to the mobile phone (Figure 11b), and then connected the signal generator with computer to collect data of current (the time is set as 20 min). We turned on the phone and started the application (five experiment settings shown in Table 5) under the experiment condition (shaker or static state). In the end, we clicked the button ("start to save data") to collect the data of current.
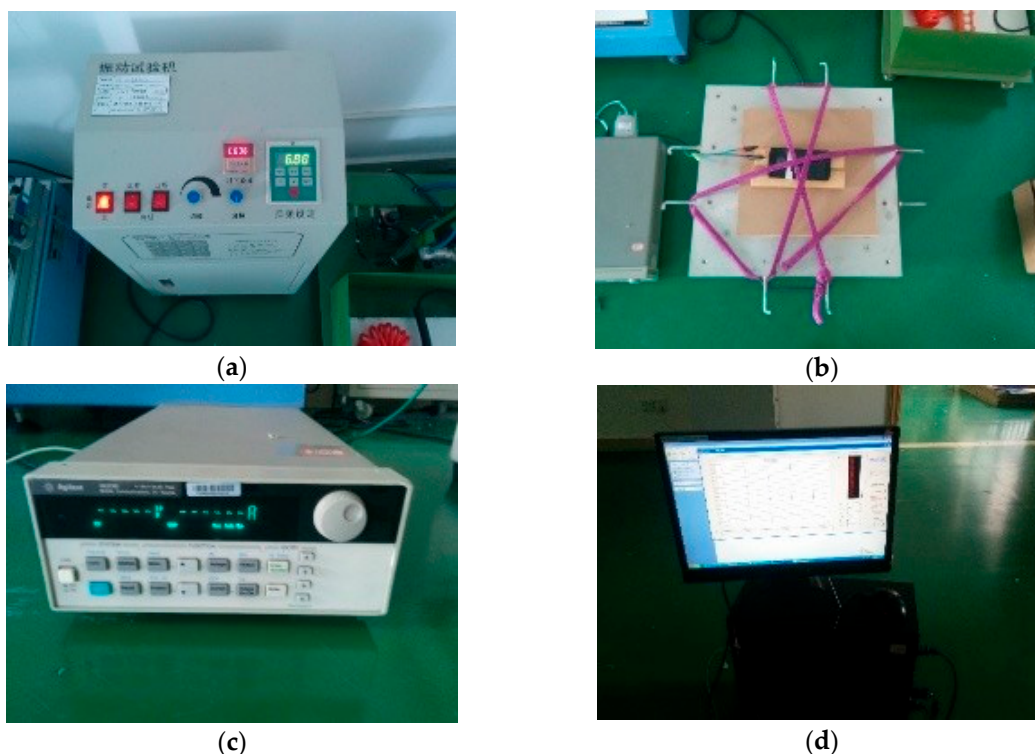

(a)

(b)

(c)

(d)

Figure 11. (**a**) Power test environments; (**b**) the phone was fixed in the shaker; (**c**) the external power; (**d**) computer used to control the experiments.

Figure 12 illustrates the average current of ARS at different sampling rates. It shows that the average current increases with the increase of the sampling rate. The average current is 20.3 mA at 1 Hz, and it is 42.7 mA at 50 Hz when the phone is in the shaker state. The average current is 20.1 mA at 1 Hz, and it is 41.1 mA at 50 Hz when the phone is in the static state. It also infers that the power consumption of ARS at rate of 10 Hz has slightly increased compared with 5 Hz. There is a large increase of power consumption when the sampling rate changes from 10 Hz to 50 Hz. There are two main reasons. One reason is that the sensor running has a great increase when the rate changes

from 10 Hz to 50 Hz (as shown in Figure 13). The other reason is the amount of data increase greatly when the sampling rate increases from 10 Hz to 50 Hz, which causes more power consumption in data processing.
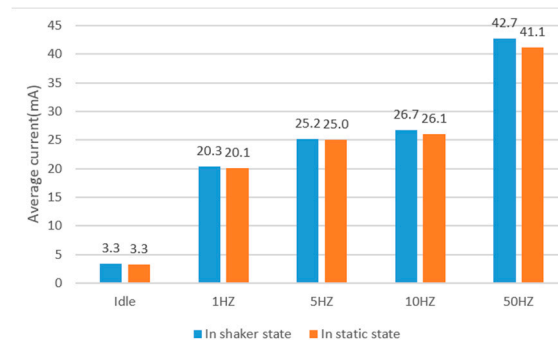

Figure 12. The average currents of the ARS at different sampling rates.

Figure 13 also shows the average current of different parts in ARS. The data processing consumes most of the power in the online ARS. The second large power consumption is the sensor running. The power consumption of the proposed recognition algorithm is very small and can even be negligible. With the decrease of the sampling rate, the energy is saved in the sensor running and data processing for the reason that the amount of data is smaller.
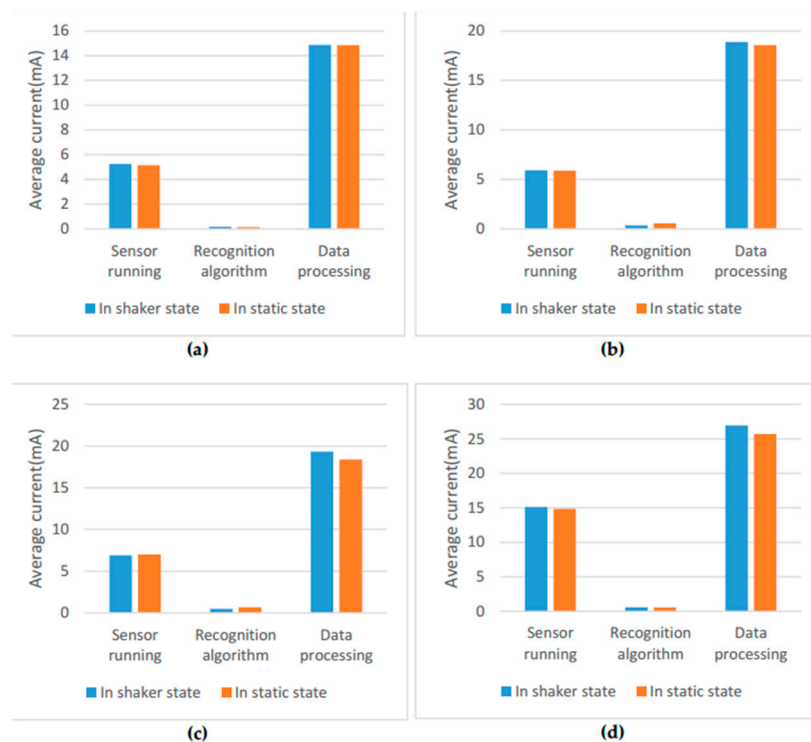

Figure 13. The average current in the sensor running, data processing and activity recognition. (**a**) ARS with a sampling rate of 1 Hz; (**b**) ARS with a sampling rate of 5 Hz; (**c**) ARS with a sampling rate of 10 Hz; (**d**) ARS with a sampling rate of 50 Hz.

We carried out another experiment to evaluate the power performance with different sampling rates. We turned on the phone when the phone was fully charged, and started phone application at four different sampling rates (1 Hz, 5 Hz, 10 Hz, 50 Hz) or idle state, respectively. Then, put the phone

statically on the table, unplugged the charging cable and turned off the screen. After 24 h, we turned on the screen, stopped the application and recorded the data. Experiments on each sampling rate and the idle state were repeated four times. We also installed an external application called Battery Monitor Weight [41] on the phone to record the battery states (the recording interval was 2 min).

Another metric for evaluation Power Consumption Ratio (PCR) is introduced in this paper:

$$PCR = \frac{PC_{f_s}}{PC_{50HZ}} \tag{10}$$

where PC_{f_s} represents the power consumption at the sampling rate of f_s after 24 h. When the phone is in idle state, f_s is set as 0.

Figure 14 shows the tendency of power consumption at different sampling rates and the idle state. From the chart, we found that our ARS at the sampling rate of 1 Hz consumed 21% energy in 24 h. The ARS at the sampling rate of 5 Hz consumed 30% battery. The ARS at the sampling rate of 10 Hz and 50 Hz consumed 35% and 52% power, respectively. The battery was expended 5% when the phone was in an idle state. It can be concluded that the lower sampling rate is, the less power the phone consumed.
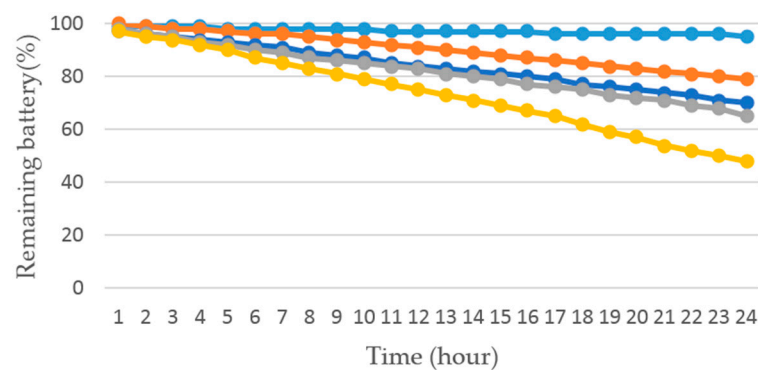

Figure 14. Remaining battery at different sampling rates and in idle state.

Table 6 summarizes the PCR in different experiment conditions. From the results presented in Figure 11 and Table 6, we can conclude that the proposed ARS of using the sampling rate of 1 Hz can save 17.3% power than the rate of 5 Hz, and there is no significant difference of accuracy achieved in the activity recognition. Comparing the sampling rates of 5 Hz or 10 Hz or 50 Hz, it can be concluded that the power consumption becomes higher with the increase of the sampling rate, but there is little improvement of accuracy. In particular, when the sampling rate increases from 10 Hz to 50 Hz, the power consumption increases 32.7%, but the accuracy only increases 0.03%. Furthermore, the ARS by using the sampling rate of 1 Hz can save 59.6% power than ARS by using the sampling rate of 50 Hz. The working time of ARS by using the sampling rate of 1 Hz is almost twice more than that of using 50 Hz.

Table 6. PCR in different experiment conditions.

Experiment Conditions	PCR
Phone in idle state	9.6%
HSVMCC by using a sampling rate of 1 Hz	40.4%
HSVMCC by using a sampling rate of 5 Hz	57.7%
HSVMCC by using a sampling rate of 10 Hz	67.3%
HSVMCC by using a sampling rate of 50 Hz	100%

5. Conclusions

This work presents a user-independent and energy-efficient ARS with high accuracy using the sampling rate lower than what is required by the Nyquist theorem. It can achieve an average accuracy of 96.0% for the recognition of six activities (sitting, standing, walking, running, climbing upstairs, and going downstairs) using the low sampling rate of 1 Hz. We also theoretically analyze the using of low sampling rate, which disagrees with the Nyquist theorem. We conclude that using a large sampling time window and a low sampling rate (such as 1 Hz) can obtain the same signal statistical properties as using a high sampling rate that meets the requirements of the Nyquist theorem.

Using a low sampling rate can save power consumption. It not only reduces the power consumption of the sensor running, but also reduces the power consumption of data processing and activity recognition. Our research shows that the ARS can save 17.3% power when using the sampling rate of 1 Hz than that of using 5 Hz, and it can save 59.6% power than that of using 50 Hz. The research has great significance for practical applications.

The H-SVM and context-based classification (HSVMCC) was proposed in our research for activity recognition. The H-SVM is an improvement of SVM algorithm, which is more efficient and achieves higher accuracy. The context-based classification is a method that uses the previous recognition results and the following recognition results to correct the current recognition results. With the integration of context-based classification presented in the paper, the average accuracy has increased by 5.1%. In comparison to the one against many multi-class approach proposed by Mortazavi et al. [31] for the activity recognition, our approach is more energy effective. It needs $N \times (N-1)$ SVM node classifiers for the N class classification in the one against the many multi-class approach. The HSVMCC approach only requires $(N-1)$ SVM node classifiers for N class classification, and, therefore, consumes less power. The use of SVM with nonlinear kernel in Mortazavi et al. [31] also requires more computational cost in comparison to the proposed HSVMCC. Additionally, feature selections prior to the training and testing in our approach can greatly reduce the dimensions of the H-SVM and thereby further reduce the energy consumption.

In this paper, we also discussed the power consumption of online ARS in detail. We found that the most power consumption in the online ARS is the data processing, and the power consumption of sensor running is in a second-largest place. The power consumption used by recognition algorithms is relatively low and has very little impact on the power consumption of the entire system.

In this study, we have not yet explored whether the position of the phone can be placed arbitrarily or not, which is an important question to be further investigated in the future.

Author Contributions: Lingxiang Zheng, Dihong Wu and Xiaoyang Ruan conceived of and developed the algorithm, in addition to performing the experiments. Shaolin Weng contributed to the design and performed the simulation, experimental model and test. Ao Peng provided guidance and direction for the implementation, and development and evaluation of the research. Biyu Tang, Hai Lu, and Haibin Shi supervised and analyzed the experimental and simulation tests. Huiru Zheng provided guidance on study design, supervised the experimental results and collaborated with the paper review.

References

1. Lara, O.D.; Labrador, M.A. A Survey on Human Activity Recognition Using Wearable Sensors. *IEEE Commun. Surv. Tutor.* **2013**, *15*, 1192–1209. [CrossRef]
2. Chen, L.; Hoey, J.; Nugent, C.D.; Cook, D.J.; Yu, Z. Sensor-Based Activity Recognition. *IEEE Trans. Syst. Man Cybern. Part C* **2012**, *42*, 790–808. [CrossRef]
3. Kwon, Y.; Kang, K.; Bae, C.; Chung, H.-J.; Kim, J.H. Lifelog Agent for Human Activity Pattern Analysis on Health Avatar Platform. *Healthc. Inform. Res.* **2014**, *20*, 69–75. [CrossRef] [PubMed]
4. Vermeulen, J.; Willard, S.; Aguiar, B.; De Witte, L.P. Validity of a Smartphone-Based Fall Detection Application on Different Phones Worn on a Belt or in a Trouser Pocket. *Assist. Technol.* **2015**, *27*, 18. [CrossRef] [PubMed]

5. Zhou, B.; Li, Q.; Mao, Q.; Tu, W.; Zhang, X. Activity Sequence-Based Indoor Pedestrian Localization Using Smartphones. *IEEE Trans. Hum.-Mach. Syst.* **2017**, *45*, 562–574. [CrossRef]

6. Chen, R.; Chu, T.; Liu, K.; Liu, J.; Chen, Y. Inferring Human Activity in Mobile Devices by Computing Multiple Contexts. *Sensors* **2015**, *15*, 21219–21238. [CrossRef] [PubMed]

7. Foerster, F.; Smeja, M.; Fahrenberg, J. Detection of Posture and Motion by Accelerometry: A Validation Study in Ambulatory Monitoring. *Comput. Hum. Behav.* **1999**, *15*, 571–583. [CrossRef]

8. Randell, C.; Muller, H. Context Awareness by Analyzing Accelerometer Data. In Proceedings of the Fourth IEEE International Symposium on Wearable Computers, Atlanta, GA, USA, 16–17 October 2000; IEEE Computer Society: Washington, DC, USA, 2000; p. 175.

9. Shoaib, M.; Scholten, H.; Havinga, P.J.M. Towards Physical Activity Recognition Using Smartphone Sensors. In Proceedings of the 2013 IEEE 10th International Conference on Ubiquitous Intelligence and Computing and 2013 IEEE 10th International Conference on Autonomic and Trusted Computing, Vietri sul Mere, Italy, 18–21 December 2013; IEEE Computer Society: Washington, DC, USA, 2013; pp. 80–87.

10. Oshin, T.O.; Poslad, S.; Zhang, Z. Energy-Efficient Real-Time Human Mobility State Classification Using Smartphones. *IEEE Trans. Comput.* **2015**, *64*, 1680–1693. [CrossRef]

11. Shoaib, M.; Bosch, S.; Incel, O.D.; Scholten, H.; Havinga, P.J.M. A Survey of Online Activity Recognition Using Mobile Phones. *Sensors* **2015**, *15*, 2059–2085. [CrossRef] [PubMed]

12. Kwapisz, J.R.; Weiss, G.M.; Moore, S.A. Activity Recognition Using Cell Phone Accelerometers. *ACM SIGKDD Explor. Newsl.* **2011**, *12*, 74–82. [CrossRef]

13. Morillo, L.M.S.; Gonzalez-Abril, L.; Ramirez, J.A.O. Low Energy Physical Activity Recognition System on Smartphones. *Sensors* **2015**, *15*, 5163–5196. [CrossRef] [PubMed]

14. Frank, K.; Munoz Diaz, E.; Robertson, P.; Sanchez, F.J.F. Bayesian Recognition of Safety Relevant Motion Activities with Inertial Sensors and Barometer. In Proceedings of the 2014 IEEE/ION Position, Location and Navigation Symposium-PLANS 2014, Monterey, CA, USA, 5–8 May 2014.

15. Kose, M.; Incel, O.D.; Ersoy, C. Online human activity recognition on smart phones. In Proceedings of the 2nd International Workshop on Mobile Sensing, Beijing, China, 16 April 2012; pp. 11–15.

16. Paniagua, C.; Flores, H.; Srirama, S.N. Mobile Sensor Data Classification for Human Activity Recognition using MapReduce on Cloud. *Procedia Comput. Sci.* **2012**, *10*, 585–592. [CrossRef]

17. Coskun, D.; Incel, O.D.; Ozgovde, A. Phone position/placement detection using accelerometer: Impact on activity recognition. In Proceedings of the 2015 IEEE Tenth International Conference on Intelligent Sensors, Sensor Networks and Information Processing, Singapore, 7–9 April 2015; pp. 1–6.

18. Anjum, A.; Ilyas, M.U. Activity recognition using smartphone sensors. In Proceedings of the 2013 IEEE 10th Consumer Communications and Networking Conference, Las Vegas, NV, USA, 11–14 January 2013; pp. 914–919.

19. Schindhelm, C.K. Activity recognition and step detection with smartphones: Towards terminal based indoor positioning system. In Proceedings of the 2012 IEEE 23rd International Symposium on Personal, Indoor and Mobile Radio Communications, Sydney, NSW, Australia, 9–12 September 2012; pp. 2454–2459.

20. Martín, H.; Bernardos, A.M.; Iglesias, J.; Casar, J.R. Activity logging using lightweight classification techniques in mobile devices. *Pers. Ubiquitous Comput.* **2013**, *17*, 675–695. [CrossRef]

21. Gordon, D.; Czerny, J.; Beigl, M. Activity Recognition for Creatures of Habit. *Pers. Ubiquitous Comput.* **2014**, *18*, 205–221. [CrossRef]

22. Yu, J.M.; Cho, S.B. A Low-Power Context-Aware System for Smartphone Using Hierarchical Modular Bayesian Networks. In *Hybrid Artificial Intelligent Systems*; Springer International Publishing: Basel, Switzerland, 2015; pp. 543–554.

23. Langdal, J.; Godsk, T. EnTracked: Energy-efficient robust position tracking for mobile devices. In Proceedings of the 7th International Conference on Mobile Systems, Applications, and Services, Kraków, Poland, 22–25 June 2009; pp. 221–234.

24. Wang, Y.; Lin, J.; Annavaram, M.; Jacobson, Q.A.; Hong, J.; Krishnamachari, B.; Sadeh, N. A framework of energy efficient mobile sensing for automatic user state recognition. In Proceedings of the 7th International Conference on Mobile Systems, Applications, and Services, Kraków, Poland, 22–25 June 2009; pp. 179–192.

25. Reddy, S.; Min, M.; Burke, J.; Estrin, D.; Hansen, M.; Srivastava, M. Using Mobile Phones to Determine Transportation Modes. *ACM Trans. Sens. Netw.* **2010**, *6*, 1–27. [CrossRef]

26. Dernbach, S.; Das, B.; Krishnan, N.C.; Thomas, B.L.; Cook, D.J. Simple and Complex Activity Recognition through Smart Phones. In Proceedings of the 2012 Eighth International Conference on Intelligent Environments, Guanajuato, Mexico, 26–29 June 2012; pp. 214–221.

27. Liang, Y.; Zhou, X.; Yu, Z.; Guo, B.; Yang, Y. Energy Efficient Activity Recognition Based on Low Resolution Accelerometer in Smart Phones. In *Advances in Grid and Pervasive Computing*; Springer: Berlin/Heidelberg, Germany, 2012.

28. Liang, Y.; Zhou, X.; Yu, Z.; Guo, B. Energy-Efficient Motion Related Activity Recognition on Mobile Devices for Pervasive Healthcare. *Mob. Netw. Appl.* **2014**, *19*, 303–317. [CrossRef]

29. Li, X.; Cao, H.; Chen, E.; Tian, J. Learning to Infer the Status of Heavy-Duty Sensors for Energy-Efficient Context-Sensing. *ACM Trans. Intell. Syst. Technol.* **2012**, *3*, 35. [CrossRef]

30. Zappi, P.; Lombriser, C.; Stiefmeier, T.; Farella, E.; Roggen, D.; Benini, L.; Tröster, G. Activity Recognition from On-Body Sensors: Accuracy-Power Trade-Off by Dynamic Sensor Selection. In *Wireless Sensor Networks*; Springer: Berlin/Heidelberg, Germany, 2008; pp. 17–33.

31. Mortazavi, B.J.; Pourhomayoun, M.; Lee, S.I.; Nyamathi, S.; Wu, B.; Sarrafzadeh, M. User-Optimized Activity Recognition for Exergaming. *Pervasive Mob. Comput.* **2016**, *26*, 3–16. [CrossRef]

32. Weng, S.; Xiang, L.; Tang, W.; Yang, H.; Zhneg, L.; Lu, H.; Zhneg, H. A Low Power and High Accuracy MEMS Sensor Based Activity Recognition Algorithm. In Proceedings of the 2014 IEEE International Conference on Bioinformatics and Biomedicine, Belfast, UK, 2–5 November 2015; pp. 33–38.

33. Krause, A.; Ihmig, M.; Rankin, E.; Leong, D.; Gupta, S.; Siewiorek, D.; Smailagic, A.; Deisher, M.; Sengupta, U. Trading off Prediction Accuracy and Power Consumption for Context-Aware Wearable Computing. In Proceedings of the Ninth IEEE International Symposium on Wearable Computers, Osaka, Japan, 18–21 October 2005; IEEE Computer Society: Washington, DC, USA, 2005; pp. 20–26.

34. Zhang, M.; Sawchuk, A.A. Human Daily Activity Recognition with Sparse Representation Using Wearable Sensors. *IEEE J. Biomed. Health Inform.* **2013**, *17*, 553–560. [CrossRef] [PubMed]

35. Liu, S.; Yi, H.; Chia, L.T.; Rajan, D. Adaptive hierarchical multi-class SVM classifier for texture-based image classification. In Proceedings of the 2005 IEEE International Conference on Multimedia and Expo, Amsterdam, The Netherlands, 6–8 July 2005.

36. Krishnan, N.C.; Juillard, C.; Colbry, D.; Panchanathan, S. Recognition of Hand Movements Using Wearable Accelerometers. *J. Ambient Intell. Smart Environ.* **2009**, *1*, 143–155.

37. Saeedi, S. Context-Aware Personal Navigation Services Using Multilevel Sensor Fusion Algorithms. Ph.D. Thesis, University of Galgary, Calgary, AB, Canada, 2013.

38. Sun, L.; Zhang, D.; Li, B.; Guo, B.; Li, S. Activity Recognition on an Accelerometer Embedded Mobile Phone with Varying Positions and Orientations. In Proceedings of the International Conference on Ubiquitous Intelligence and Computing, UIC 2010, Xi'an, China, 26–29 October 2010; pp. 548–562.

39. Chang, C.C.; Lin, C.J. LIBSVM: A Library for Support Vector Machines. *ACM Trans. Intell. Syst.Technol.* **2011**, *2*. [CrossRef]

40. Hall, M.; Frank, E.; Holmes, G.; Pfahringer, B.; Reutemann, P.; Witten, I.H. The WEKA data mining software: an update. *ACM SIGKDD Explor. Newsl.* **2009**, *11*, 10–18. [CrossRef]

41. 3C Battery Monitor Widget. Available online: https://play.google.com/store/apps/details?id=ccc71.bmw&hl=zh-CN (accessed on 1 January 2016).