Journal of Reliable Intelligent Environments (2022) 8:285–298
https://doi.org/10.1007/s40860-022-00181-6

ORIGINAL ARTICLE

# Feature selection and human arm activity classification using a wristband

Shumei Zhang[1,2] · Vic Callaghan[3] · Xiangping An[1] · Liying Duan[1] · Hua Liu[1]

Received: 12 October 2021 / Accepted: 10 June 2022 / Published online: 19 July 2022

**Abstract**

Activity monitoring is a core application for wristbands and, consequently, all the top selling brands (Xiaomi, Apple, Huawei, Fitbit, and Samsung) incorporate accelerometers as a core movement sensor. Applications range from sports to fitness, supported by algorithms that analyze the sensor data. Thus, there are significant benefits to be accrued from improving the activity classification performance of wrist-worn activity monitors, a goal that this study seeks to address. Further to achieving this goal, this paper presents research which investigates the potential for improving strategies and algorithms used in data pre-processing and model training/testing, for wrist-worn accelerometer sensing. To those ends we investigate different techniques for data sampling frequency, feature ranking, feature scaling and sub-feature sets selection, as well as model selection strategies based on a set of neural network, support vector machine, and Gaussian Naïve Bayes classification algorithms. We explore the effects of different model training and testing strategies, and compare three models trained with different datasets organized by personalization, partial mixing, and full mixing from multiple subjects. Their relative performance is then compared based on different test datasets, which are personalized, mixed with pre-specified training subjects, and non-pre specified (unseen/new) subjects, respectively. Moreover, a novel plurality voting mechanism was explored as a means to adjust the prediction result during the model testing stage. Finally, the paper concludes by presenting the main finding of the research which are that the most robust and reliable performance for human activity classification can be obtained by combining a personalized model with a plurality voting mechanism.

**Keywords** Data pre-processing · Wearable technology · Activity classification · Feature selection · Plurality voting

## 1 Introduction

According to the International Data Corporation (IDC), one of the world's largest market intelligence organizations, in 2019, over 34 million wrist-worn wearables (wristbands and smartwatches) were shipped, creating a market valued at about USD 27 million which is expected to rise to USD 1252 billion by 2025 [1, 2]. Activity monitoring is a core application for wristbands, with all the top selling brands (Xiaomi, Apple, Huawei, Fitbit, and Samsung) incorporating accelerometers as their core movement sensor [3]. Applications for human activity recognition cover a wide range of research areas including surveillance, healthcare, sports training, robotics, smart homes, and human computer interaction [4, 5]. Human posture recognition involves automatically detecting and analyzing human activities based on data acquired from sensors such as cameras, wearable inertial sensors. Thus, there are significant benefits to be accrued from improving human activity classification performance of wrist-worn activity classifiers, a goal that this study seeks to address.

Technological advances have allowed Wristbands, containing embedded accelerometers, to be realized in unobtrusive, light-weight, low-cost and power-efficient mobile forms [6, 7]. To enable wristbands to provide a reliable posture recognition system there is a need to research data pre-processing and optimal model selection techniques. To convey the ideas underlying and motivating this work, this study takes as an example scenario, an imaginary crowd control steward, who is policing a sports event where there

✉ Shumei Zhang
may.z111@yahoo.com

[1] Shijiazhuang University, Shijiazhuang, China

[2] Hebei Key Laboratory of IoT Blockchain Integration, Shijiazhuang, China

[3] School of Computer Science and Electronic Engineering, Essex University, Essex, UK

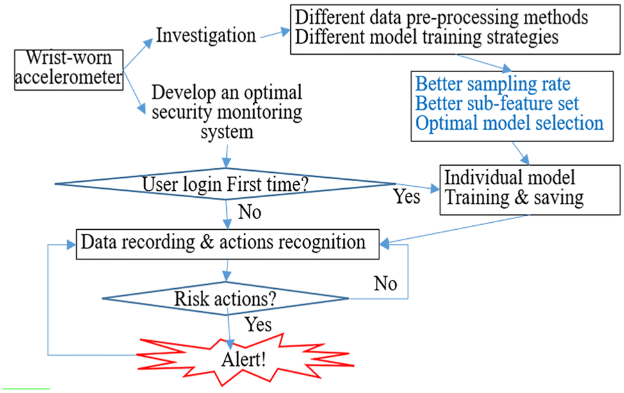

Wrist-worn accelerometer
Investigation
Different data pre-processing methods
Different model training strategies
Develop an optimal security monitoring system
Better sampling rate
Better sub-feature set
Optimal model selection
User login First time?
Yes
Individual model Training & saving
No
Data recording & actions recognition
No
Risk actions?
Yes
Alert!

**Fig. 1** The purpose of this study and main contributions (colored in blue)

is a large unruly crowd but (fortunately) on an occasion where he is wearing a smart wristband containing movement analysis monitoring algorithms (which this work will investigate) that can detect if the Stewart was 1—walking, 2—running, 3—waving, 4—punching, 5—clenching his/her fist, 6—slapping, 7—throwing something, or 8-standing still, so that assistance might be provided, if necessary. The scenario is somewhat artificial but its intention is simply to act as a conceptual vehicle to group together and convey various challenging armband activities to motivate the AI research which will be described over the course of the paper.

Figure 1 visually depicts the aim and main contributions to computer science arising from the work described in this paper. In summary, this paper seeks to advance knowledge by presenting:

- An investigation of feature selection methods: the research explores performance consequences arising from design choices relating to 6 sampling rates, five feature selection techniques and three feature scaling methods, as well as 14 sub-feature sets that were compared and selected using three classification algorithms: a Gaussian Naïve Bayes (Bayes), a Support Vector Machine (SVM) with a radial basis function (RBF) kernel function, and a Neural Network (NN).
- A design of optimal model selection strategies: three types of models (data collected from personalization, part of combined subjects and full combined subjects) were trained and compared based on three testing sets (collected from personalized training subjects, 5 training subjects, and 12 new subjects). In addition, a plurality voting mechanism was used to adjust the original prediction result during the model testing.

## 2 Related work

Activity data tracking, data pre-processing methods and optimal model selection algorithms are important mechanisms for human activity recognition. Raw activity data sets, tracked from wearable sensors (e.g. accelerometer, gyroscope), are normally not used directly for activity classification, rather data processing algorithms are sometimes used for improving data usability. Moreover, feature selection and feature scaling are generally performed during the data pre-processing step. The selected sub-feature sets used for models training and testing can affect the model's performance significantly. These issues are discussed in more detail in the following paragraphs.

### 2.1 Data pre-processing

Feature selection aims to identify a smaller feature space by choosing those features from a dataset that contribute most to the performance of the activity classification model. For instance, irrelevant or partially relevant features can have a negative impact on the accuracy of many models [8]. There are several potential benefits to feature selection: reduced overfitting, improved accuracy and reduced training time, as well as facilitating better data visualization and understanding [9, 10]. Generally, feature selection has two main objectives; maximizing classification accuracy and minimizing the number of features [11].

There are many feature selection techniques, each with different (dis)advantages and computational costs. These can be categorized into three groups: filter, wrapper and embedded methods.

Filter methods apply a statistical measure to calculate the feature relevance scores for each of features, removing low-scoring features. For example, the 'Pearson correlation coefficient' (Corr.) approach measures linear correlation between two variables, with the resulting value lying in the range [− 1, 1]. The value + 1 means a perfect positive correlation, − 1 means a perfect negative correlation (as one variable increases, the other decreases) and 0 means no linear correlation between the two variables [12]. A higher absolute value of the correlation coefficient indicates a stronger relationship between variables. Therefore, features with higher scores should be selected for inclusion.

Wrapper methods utilize a predictor to search the optimal feature subset. In this method, different combinations of features are prepared and evaluated by assigning a score based on predictor accuracy, comparing the result of each combination to the others to find the best set [13]. A popular method is recursive feature elimination (RFE), which selects features by recursively considering smaller and smaller sets of features, based on an external predictor which assigns weights to features [14].

Embedded methods select the best contributing features based on the accuracy of the model, during the model creation stage (i.e. the best feature subset is embedded into the model). The least absolute shrinkage and selection operator (Lasso) and Ridge regression are two examples of this approach. The Lasso performs both feature selection and regularization, being also called L1 regularization [15]. L1 feature scores are in the range [0,1]. The L1 regularization causes the regression coefficient of the least contributing variable to shrink to zero or near zero. Thus, only some features with higher scores should be selected. A variation of Ridge regression, called L2, adds a penalty to the loss function (L2-norm), which forces the coefficient values to be spread out more equally [16].

Feature scaling/normalization aims to make comparisons between different features. Most machine learning algorithms will require proper scaling of features if the attributes are measured using different scales. In this case, feature scaling can enhance the performance of machine learning algorithms [17]. In other cases, the feature normalization could reduce the training time significantly, although it may not greatly affect in the prediction output. In addition, the gradient descent converges much faster when it includes feature scaling than without it [18]. Many techniques are used for feature scaling, for example, the Min–Max method and the standardization method.

The Min–Max normalization method can normalize all feature values to the range [0, 1] or [− 1, 1]. Selecting the target range depends on the nature of the data [19]. The standardization normalization (also called $Z$-score) method sets the values of each feature distribution to $N$ (0, 1) (normal distribution with zero mean and unit-variance) and assumes that initial features follow a normal distribution [20].

The work of Gao et al. [21] has investigated how sampling frequency impacts the performance of classifiers, by increasing the sampling rate from 10 to 200 Hz in 10 Hz increments. Their experimental results have revealed that the recognition accuracy was not sensitive to the sampling rate (only increasing by 1% increasing above 20 Hz and stabilizing beyond 50 Hz). However, the high sampling rate can lead to increasing the computational load and power usage requirements. To choose a better sampling rate, it is necessary to balance the classification accuracy against the computational load. This is especially important for wearable devices because their processors are not very powerful. Thus, in this paper, we have chosen to expand the sampling rate range and analyze the correlation between classifier performance and sampling rate by comparing classification accuracy using 6 datasets collected at 6 different sampling rates from 12.5 to 400 Hz.

Tchuente et al. [22] have identified a combination of feature selectors and classification models for aggressive activity classification using smartwatches. They used the WEKA tool to run 6 machine learning classifiers (RF, kNN, MLP, SVM, Bayes, DT) coupled with three feature selectors (ReliefF, InfoGain, Correlation). Ten activities were separated into aggressive behavior (punching, shoving, slapping, shaking) and non-aggressive behaviors (clapping hands, waving, opening/closing a door, handshaking, typing on a keyboard). Their experimental results demonstrated that the combination of kNN and ReliefF obtained the best performance for differentiating between aggressive and non-aggressive actions, with 99.6% accuracy, 98.4% sensitivity, 99.8% specificity and 98.9% precision. However, their algorithms were performed offline, not-in real-time meaning that the selected models might in practical applications yield different and (probably) lower performance metrics [23]. This work is also relevant to our study, since we also propose a human arm activity recognition method that combines a better sub-feature set selection method based on a hybrid classifier.

## 2.2 Activity classification

Classification aims to use trained models to assign a class label to each sample in a given dataset based on the values of features. Commonly, classification uses supervised machine learning algorithms to build a model using training datasets with associated class labels. Classification algorithms have a wide range of applications such as human activity recognition, image identification, social network analysis etc. There are several types of classification techniques, for a given classification problem, the choice of algorithms is normally decided based on the classification accuracy and computational efficiency required. The most popular algorithms reported in research literature for activity classification are Naïve Bayes (Bayes), k-nearest neighbor (kNN), C4.5 decision trees (DT), neural networks (NN), SVM and rule-based algorithms [24]. For example, Yang [25] classified human activities using accelerometers built into smartphones. In this work six daily activities: walking, sitting, standing, driving, running and cycling were classified. In addition, the classification results were compared based on different features, such as mean and standard deviation extracted from sensed acceleration dataset. The research evaluated and compared four classifiers, namely DT, kNN, Bayes and LibSVM using the cross-validation method. Their results showed that the DT algorithm achieved the best overall performance for an acceptable computational complexity and load. kNN and SVM obtained good classification performance based on the selected magnitude features, but at cost of computational load/time. From this body of earlier work, it was evident that feature extraction is computational demanding, which presents particular problems to portable and wearable devices, such as phones, which have less powerful processors. Therefore, efficient computing algorithms and strategies are important for wristband applications. Namsrai et al. [26] proposed a method for building an ensemble of classification models using a feature selection schema. Several classifiers were built, based on different feature subsets, which combined the classifiers by adopting a voting approach to form a classification ensemble. Experimental results illustrated that this method can improve the classification accuracy in high dimensional datasets. Ngo et al. [27] proposed an 'inter-class relation algorithm' to deal with similar activity classes. In their work, gait recognition performance was improved overall using a waist-worn inertial sensor. Jadhav and Channe [28] compared three classification algorithms, including DT, kNN and Bayes based on the same dataset using the WEKA tool. Their study aimed to analyze the advantages and disadvantages of one classification algorithm over the other. Their experimental results showed that kNN was effective for small data sets, but slow for big data sets; DT was shown to produce highly accurate results for some cases, but suffered from overfitting in many others. Although Bayes had lower accuracy, it was able to handle noisy datasets and large amounts of data relatively efficiently; considering the ability to deal with big data and noisy data, this paper chose Bayesian as one of the three algorithms for comparison.

To select optimal models, some studies have investigated different model training methods. For example, Weiss et al. [29] evaluated three types of models: impersonal, personal, and hybrid based on datasets collected using a smartphone. Their experimental results demonstrated that impersonal models performed much more poorly than personal models, while the performance of hybrid models was able to approach that of personal models. They also described that a personalized model can achieve nearly perfect results even using only a small amount of user specific training data. Additionally, since hybrid models also require user specific training data, one may as well just use personal models instead. Therefore, they strongly suggested building personal models whenever possible. Tapia et al. [30] experimented with the C4.5 DT algorithm, presenting performance data for physical activity recognition, using five triaxial wireless accelerometers (attached to each limb and hip). They also placed a heart rate monitor on the chest for discriminating between the intensity of activities. In their experiments they compared the performance of the C4.5 DT algorithm using subject-dependent and subject-independent training based on datasets consisting of 30 physical gymnasium activities collected from a total of 21 people. Their experimental results revealed a recognition accuracy of 94.6% using a subject-dependent model and 56.3% using a subject-independent model. The addition of heart rate data improved the recognition accuracy by only 1.2% for the subject-dependent model and by 2.1% for the subject-independent model.

Ferrari et al. [31] provided a very informative survey of the main stages of the human activity classification process, including data collection, data pre-processing, data segmentation, feature extraction, and activity classification. They noted that, in the data acquisition step, accelerometers, gyroscopes, and magnetometers are the most commonly used sensors, and that the sampling rate was an important factor to consider; In the data segmentation step, they argued that window features are affected by window type, window size, and overlap of adjacent windows; Considering the feature extraction stage, explained how it aimed to extract the most meaningful parts of information from raw data to feed into classification algorithms; Finally, in the classification stage, they described how machine learning and deep learning algorithms were popular approaches to activity recognition, but due to the problem of population diversity, that it was difficult for these methods to achieve good performance for new unseen users. The concluded that one solution was to use personalized machine learning methods to overcome the problem of natural differences between user activity patterns [32].

The literature study of related work demonstrated that each algorithm has its advantages and disadvantages. No one classifier works best for every problem. Therefore, there are many factors to consider such as the size and structure of the dataset in a given application. The consequences for researchers working in this area are that their remains a need to experiment with many different algorithms to evaluate performance and for system designers to be able to select the appropriate classifier for a given application.

Thus, in this study, we have investigated different techniques for data sampling frequency, feature ranking, feature scaling and sub-feature sets selection, as well as model selection strategies based on a set of NN, SVM, and Bayes classification algorithms. Finally, informed by the literature and our own experimental results, we designed a hybrid classifier for human arm activity classification that combined a personal model together with a plurality voting mechanism.

## 3 Methodologies

The goal of this section is to describe the methodology we used for each task in the human arm action classification process. First, the system configuration and sensor specifications used in this study are introduced, then we introduce the sampling rate selection, followed by feature extraction and optimal sub-feature set selection; Finally, there is a hybrid classifier design.

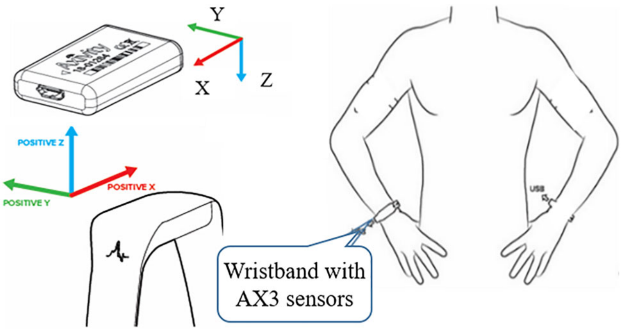


**Fig. 2** System configuration

## 3.1 System configuration

To collect data on human arm activities, the Axivity wristband was used to support this study, which embedded an AX3 accelerometer [33]. AX3 features a state-of-the-art MEMS 3-axis accelerometer and Flash based on-board memory. The device incorporates a real-time quartz clock, 512 MB of built-in memory, temperature and light sensors. Its sample rate is configurable in a range from 12.5 to 3200 Hz, battery life is 30 days at 12.5 Hz.

The wristband was attached to the right wrist as shown in the right of Fig. 2, and the orientation of the AX3 axis is shown in the left of Fig. 2.

## 3.2 Comparison of different sampling frequency

To identify a reasonable sampling frequency, we were first guided by the literature which indicated the optimum sample rate was in the range 1–400 samples per second [34]. Using this information, we set up an experiment to assess the performance of three classifiers for 6 sampling rates (12.5 Hz, 25 Hz, 50 Hz, 100 Hz, 200 Hz, 400 Hz) as described below.

First, we configured six AX3 sensors, for six different sampling rates in the list above. Then one subject was asked to wear all 6 sensors on their right arm as shown in Fig. 3.

The subject was instructed to undertake 8 activities [*walk, run, wave, punch, fistClench, slap, throw, still*] for 3 min, while 6 datasets were collected and saved. Figure 4 visualized the Ay signals from the 6 datasets concerned.

Figure 4 shows that the Ay signals for 6 different sampling rates are similar in shape but significantly different in number of samples, going from about 2000 samples (12.5 Hz) to about 60,000 samples (400 Hz) for the same data acquisition time.

The 6 sample rates were evaluated and compared using a cross-validation technique with three classifiers: a Gaussian Naïve bayes (Bayes), a support vector machine (SVM) with a description of the RBF kernel function, and the neural network (NN) used in our experiments. Their classification accuracy for the 6 datasets is shown in Table1.

**Table 1** Comparing the performance of three classifiers (Bayes, SVM and NN) based on six datasets collected at six different sampling rates

| Frequency | 12.5 Hz | 25 Hz | 50 Hz | 100 Hz | 200 Hz | 400 Hz |
|---|---|---|---|---|---|---|
| Classifier | | | | | | |
| NN | 0.83 | **0.85** | 0.85 | 0.87 | 0.87 | **0.89** |
| SVM | 0.77 | **0.81** | 0.82 | 0.83 | **0.86** | 0.76 |
| Bayes | 0.68 | **0.72** | 0.73 | 0.73 | **0.74** | 0.73 |

Table 1 indicates that the datasets for the six sampling rates, applied to the same classifier, produced similar classification accuracy although the three classifiers delivered differing levels of performance. For example, the classification accuracy ranged from 83 to 89% for the NN, from 77 to 86% for the SVM and from 68 to 74% for the Bayes approach, based on sample frequencies of 12.5 Hz to 400 Hz. From the results in Table1, there is no significant difference in classification accuracy between 25 and400 Hz. This result is similar to the Gao et al. study [21], although we widen the sampling rate range.

Leaving aside the slightly weaker processors in wearables, 200 Hz may be the best option based on the performance of the three classifiers in Table 1. However, it is well known that higher sampling rate produce a higher data load and thus, lower data processing throughput. Finally, we select this sampling rate 25 Hz as a good balance between data processing load and classification accuracy.

Furthermore, the experimental results in Fig. 4 and Table 1 show that the data acquisition results are not affected by the exact location of the wristband (near the elbow or wrist), but only by the sampling rate.

## 3.3 Data pre-processing

### 3.3.1 Feature extraction

Feature extraction is defined as the process of extracting a new set of features from the original dataset through some functional mapping [35]. Using a set of features instead of raw data can potentially improve classification performance [36]. In this study, raw 3D acceleration datasets were collected using the AX3 accelerometer and organized as ($t$, $A_x$, $A_y$, $A_z$). Then more features ($A_{xyz}$, $\Delta A$, $\varphi$, $\theta$) were extracted using the following equations from Eqs. (1) to (4), respectively.

$$A_{xyz}(t) = \sqrt{A_x(t)^2 + A_y(t)^2 + A_z(t)^2} \tag{1}$$

$$\Delta A(\mathrm{t}) = \left|A_{xyz}(t) - A_{xyz}(t-1)\right| \tag{2}$$

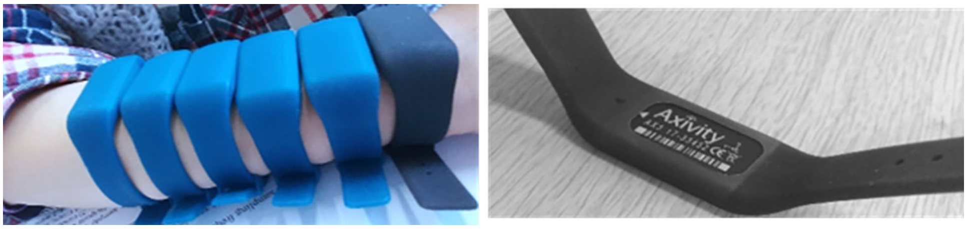


**Fig. 3** Six sensors were setting up in 6 different sample rates and collecting 6 datasets at the same time

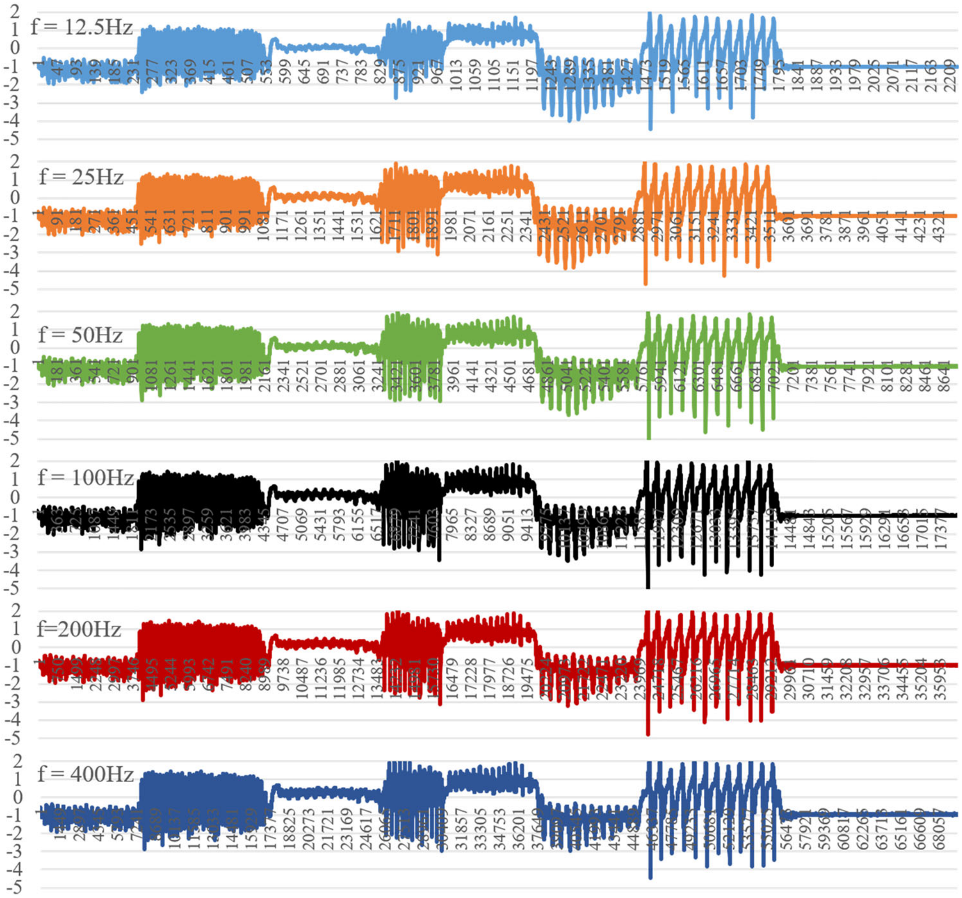


**Fig. 4** Comparison of the 6 datasets (12.5 Hz, 25 Hz, 50 Hz, 100 Hz, 200 Hz, and 400 Hz), using their Ay signals for 3 min

$$\phi(\mathrm{t}) = Atan(\frac{A_y(t)}{A_z(t) + 0.00001}) \times \frac{180^{\circ}}{\pi} \tag{3}$$

$$\theta(\mathrm{t}) = Atan\left(\frac{-A_x(t)}{\sqrt{A_y(t)^2 + A_z(t)^2} + 0.00001}\right) \times \frac{180^{\circ}}{\pi} \tag{4}$$

where $A_x, A_y, A_z$ are the acceleration components in the $X$, $Y$, and $Z$ directions, respectively; $A_{xyz}$ is the three-dimensional acceleration; ΔA is the absolute $A_{xyz}$ change between time points $t$ and $(t-1)$; $\varphi$ and $\theta$ are the sensor rotation angles around the $X$ and $Y$ axis, respectively. The number 0.00001 in Eqs. (3) and (4) is an adjustable constant pragmatically introduced via our experiments, to avoid a zero divisor. There are seven features ($A_x$, $A_y$, $A_z$, $A_{xyz}$, $\Delta A$, $\varphi$, $\theta$) in the dataset used for further feature selection ($t$ is simply used to record time points).

### 3.3.2 Features ranking and sub-feature sets selection

The feature ranking is performed by five different feature selection methods at the same time, which includes the Pearson correlation coefficient (Corr.), the least absolute shrinkage and selection operator (Lasso), ridge, the recursive feature elimination (RFE), and stability selection.

The experimental results are shown in Table2, which demonstrated that each of the seven features has a different score for the five different feature selection methods. For example, the feature $\Delta A$ obtained the highest score for all of the five methods; the feature $A_y$ received one highest score 1 for REF method, but has lower scores for other four meth-

**Table 2** The feature ranking score obtained from 5 different feature selection methods

| Features | $A_x$ | $A_y$ | $A_z$ | $A_{xyz}$ | $\Delta A$ | $\varphi$ | $\theta$ |
|---|---|---|---|---|---|---|---|
| Method | | | | | | | |
| Corr | 0.13 | 0.02 | 0.25 | 0.07 | **0.91** | 1 | 0 |
| Lasso | 0.12 | 0 | 0.55 | 0 | **1** | 0.01 | 0 |
| Ridge | 0.71 | 0.03 | 0.54 | 0.36 | **1** | 0 | 0 |
| RFE | 0.4 | 1 | 0.6 | 0.8 | **1** | 0 | 0.2 |
| Stability | 0.14 | 0 | 0.52 | 0 | **1** | 1 | 0 |

**Table 3** Selected potential sub-feature combination based on feature ranking scores from different methods

| Methods | Selected potential sub-feature sets |
|---|---|
| Corr | Fset1 ($A_x$, $A_y$, $\triangle A$, $\varphi$), Fset2 ($A_z$, $\Delta A$, $\varphi$), Fset3 ($\Delta A$, $\varphi$) |
| Lasso and ridge | Fset4 ($A_z$, $\Delta A$), Fset5 ($A_x$, $A_z$, $\Delta A$), Fset6 ($A_x$, $A_z$, $A_{xyz}$, $\Delta A$) |
| REF | Fset7 ($A_x$, $A_y$,$A_z$, $A_{xyz}$, $\Delta A$), Fset8 ($A_y$,$A_z$, $A_{xyz}$, $\Delta A$)<br>Fset9 ($A_y$,$A_{xyz}$, $\Delta A$), Fset10 ($A_x$, $A_y$, $A_z$, $A_{xyz}$, $\Delta A$, $\theta$) |
| Stability | Fset11 ($A_x$, $A_z$, $\Delta A$, $\varphi$), Fset12 ($A_z$, $\Delta A$, $\varphi$) |
| All | Fset13 ($A_x$, $A_y$, $A_z$, $A_{xyz}$, $\Delta A$, $\varphi$, $\theta$) |
| Original | Fset14 ($A_x$, $A_y$,$A_z$) |

ods; the feature $\varphi$ received two higher scores 1 for methods of Corr. and Stability, but also got three 0 scores for other methods; $A_x$ and $A_z$ were ranked in similar position, only receiving some medium scores; $A_{xyz}$ got one high score 0.8 for REF method, one medium score 0.36 for Ridge method and three lower scores; $\theta$ ranked in the lowest position, it had more zero scores. Therefore, different combinations of sub-features were chosen based on the feature ranking scores from different methods.

Finally, 14 sub-feature sets are selected, also labeled from Fset1 to Fset14, as listed in Table 3, which will be used for optimal sub-feature selection in the next section.

### 3.3.3 Feature scaling and the best sub-feature set selection

The values for each of the features use different scales. For example, accelerations have values in the range from − 3 g to 3 g, however the rotation angles have values in the range from 0° to 360°. Hence feature scaling is necessary for some machine learning algorithms. Three feature scaling methods (original, standardization, and minmax) were compared using the above 14 sub-feature sets. The training and testing datasets were organized in three groups ($X_{\text{orig}}$, $X_{\text{stand}}$, and $X_{\text{minmax}}$), respectively, as shown in Eqs. (5), (6) and (7).

$$X_{\text{orig}} = (Ax, Ay, Az, Axyz, \Delta A, \varphi, \theta) \tag{5}$$

$$X_{\text{stand}} = \frac{X_{orig} - \mu}{\sigma}, \begin{cases} \mu = \frac{1}{N}\sum_{i=1}^{N} X_{\text{orig}}(i) \\ \sigma = \sqrt{\frac{1}{N}\sum_{i=1}^{N}\left(X_{\text{orig}}(i) - \mu\right)^2} \end{cases} \tag{6}$$

$$X_{\text{minmax}} = \frac{X_{\text{orig}} - \min(x_{\text{orig}})}{\max(x_{\text{orig}}) - \min(x_{\text{orig}})} \tag{7}$$

In these equations, $X_{\text{orig}}\ (i)$ is the $i$th original feature vector, $\mu$ is the mean of that feature vector, $N$ is the number of samples, and $\sigma$ is its standard deviation.

Each of the three group datasets were classified, based on the 14 sub-feature sets, using three algorithms (Bayes, SVM, and NN). The experimental results are presented in Table 4.

Table 4 demonstrates that the 'stand' scaling dataset displays the best performance and that the minmax scaling dataset shows the worst performance for all three algorithms.

For the Bayes algorithm, the 'stand' scaling dataset shows very similar results to the original dataset. This means that 'stand' scaling is necessary when NN and SVM classifiers are used for activity classification.

Table 4 also illustrated that two sub-feature sets Fset7 ($A_x$, $A_y$, $A_z$, $A_{xyz}$, $\Delta A$) and Fset13 ($A_x$, $A_y$, $A_z$, $A_{xyz}$, $\Delta A$, $\varphi$, $\theta$) achieved highest performance (90%).

From analyzing these results, Fset7 was selected as the best sub-feature set in this study for further model training and testing, since it was the best balance between delivering higher classification accuracy and requiring least features.

## 3.4 Model design with plurality voting algorithm

To improve activity classification performance, a hybrid classifier was designed for arm action recognition. The hybrid classifier combined an original classification algorithm (originalA) with plurality voting algorithm, and was named pluralityVA.

The originalA could be based on any selected machine learning algorithm such as NN, SVM or Bayes. Because the classifiers operate on the testing set point-by-point, it is possible that they return different class labels for a same action, during a given period of time. To reduce the number of possible misclassified points, a plurality voting algorithm was designed based on a data segmentation technique, to assign the same class label to identical actions that may have more than one classification. Data segmentation splits the signal into smaller data segments, also known as windows ($w$), which helps the algorithm deal with high volumes of data and facilitates simpler and less time-consuming data analysis (i.e., lowers the computational load). Three main types of approaches are used in human activity recognition, including activity-defined windows, event-defined windows, and sliding windows [37]. Sliding windowing is the most

**Table 4** Comparison of activity classification results based on three feature scaling methods (original, minmax and stand scaling) for each of the selected 14 sub-feature sets, using three algorithms (Bayes, SVM and NN)

| Methods | Bayes | | | SVM | | | NN | | |
|---|---|---|---|---|---|---|---|---|---|
| | Origi | mixmax | stand | Origi | mixmax | stand | Origi | mixmax | Stand |
| Features | | | | | | | | | |
| Fset1 | 0.73 | 0.73 | 0.73 | 0.74 | 0.72 | 0.77 | 0.73 | 0.70 | **0.83** |
| Fset2 | 0.67 | 0.66 | 0.66 | 0.70 | 0.67 | 0.73 | 0.71 | 0.66 | **0.80** |
| Fset3 | 0.61 | 0.63 | 0.63 | 0.57 | 0.54 | 0.60 | 0.57 | 0.53 | **0.65** |
| Fset4 | **0.63** | 0.58 | 0.62 | 0.52 | 0.51 | 0.56 | 0.58 | 0.49 | 0.62 |
| Fset5 | 0.66 | 0.64 | 0.66 | 0.62 | 0.58 | 0.66 | 0.67 | 0.55 | **0.73** |
| Fset6 | 0.67 | 0.66 | 0.67 | 0.71 | 0.62 | 0.74 | 0.67 | 0.57 | **0.80** |
| Fset7 | 0.75 | 0.74 | 0.75 | 0.78 | 0.72 | 0.82 | 0.82 | 0.68 | ***0.90*** |
| Fset8 | 0.71 | 0.70 | 0.71 | 0.72 | 0.66 | 0.76 | 0.77 | 0.64 | **0.86** |
| Fset9 | 0.68 | 0.62 | 0.68 | 0.65 | 0.60 | 0.69 | 0.70 | 0.57 | **0.76** |
| Fset10 | 0.74 | 0.73 | 0.74 | 0.79 | 0.74 | 0.83 | 0.69 | 0.67 | **0.89** |
| Fset11 | 0.74 | 0.73 | 0.73 | 0.77 | 0.74 | 0.81 | 0.75 | 0.70 | **0.88** |
| Fset12 | 0.69 | 0.69 | 0.68 | 0.74 | 0.71 | 0.77 | 0.70 | 0.66 | **0.82** |
| Fset13 | 0.76 | 0.76 | 0.75 | 0.78 | 0.74 | 0.83 | 0.76 | 0.72 | ***0.90*** |
| Fset14 | 0.72 | 0.71 | 0.72 | 0.69 | 0.64 | 0.69 | 0.79 | 0.60 | **0.80** |

Bold values indicate better accuracy and best feature scaling method (stand); Bold italics values indicate best accuracy and best sub-feature sets (Fset7 and Fset13).

widely employed segmentation technique in activity recognition [38].

In this paper, the pluralityVA determines a relevant majority class value for each sliding window *P(w)*, here the relevant majority class is calculated based on the predicted result from the original classification algorithm (originalA). Details of this pluralityVA algorithm is described below.

1) There exists a classifier that defined a list of class labels $C = [c_1 \dots c_m]$, where $m$ is the total class number.
2) The classifier predicts a result of a testing set using the orignalA algorithm, and the result denotes as a list $P = [p_1, \dots, p_n]$, where for all $p_i$ belong to $C$, and $n$ is the sample number of the testing set in total ($n$ = sample rate $f$ times data collecting time $t$), as shown in Eq. (8).
3) A window ($w$) size is set as *1 s* period of time (25 samples in this study). Then, count the number of each class ($N_{ci}$) for every sliding window ($w$) from the predicted result $P(w)$ as shown in Eq. (9).
4) Obtain the relevant majority class label key, and use this key value to replace all values in $P(w)$, as shown in Eq. (10).

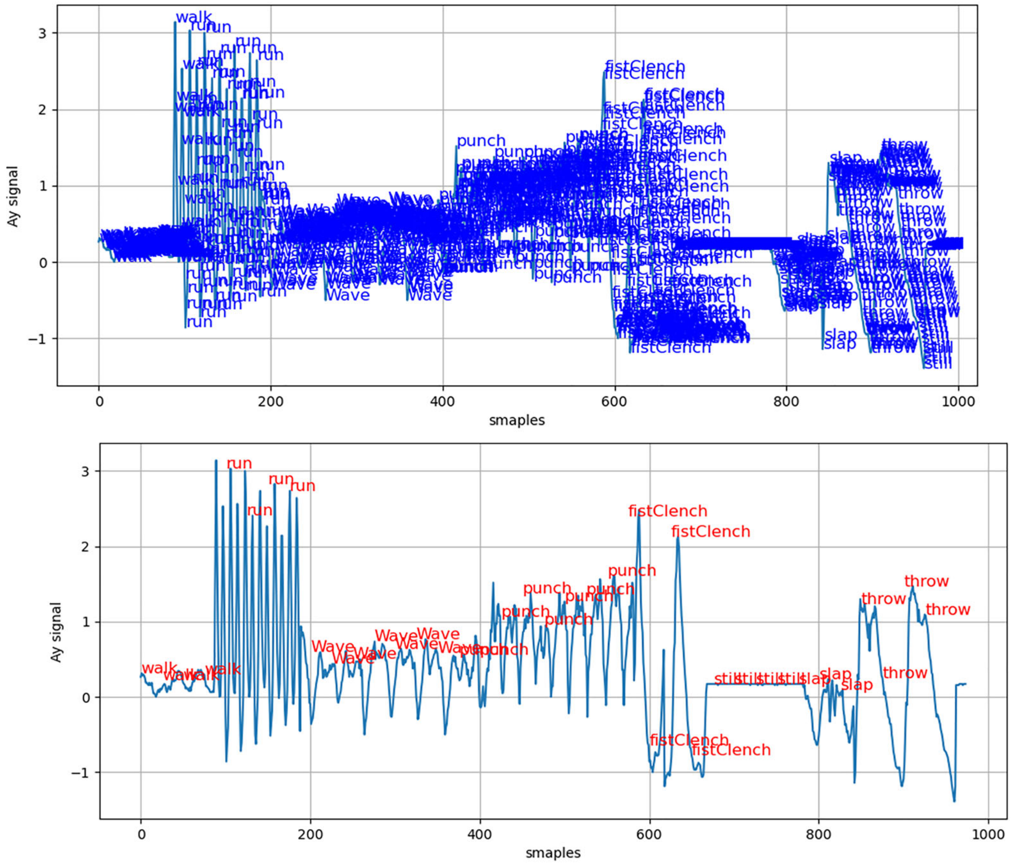
Compared with the orignalA method, the pluralityVA algorithm not only improved the classification accuracy, but also clearly illustrated the classification results through offering a diagrammatic view of the acceleration signal, as shown in Fig. 5a, b. For example, in the "run" time period in Fig. 5a, some points were incorrectly classified as "walk", however, in Fig. 5b, the wrongly classified points were corrected for each sliding window.

The detailed performances of pluralityVA and originalA algorithms will be compared in the Experiments section.

## 4 Experiments

Data were collected from 17 subjects. All subjects performed 8 actions corresponding to 8 classes: {*walking, running, waving, punching, fistClenching, slaping, throwing, still*}. The experimental results were validated against synchronized videos, recorded with 3 cameras installed on a ceiling or high up on a wall. All 17 subjects performed 8 actions based on a crowd safety scenario simulated in an AI lab environment.

$$pluralityVA = \begin{cases} \exists \left\{ \begin{array}{ll} C = [c_1, \dots, c_m], & n = t \times f \\ \quad P = [p_1, \dots, p_n], \forall p_i \in C & \end{array} \right. & (8) \\ N = counter([p_i, \dots, p_{i+w}] | C) & (9) \\ key = c_i : \max(N), P(w) = [key] * w & (10) \end{cases}$$



(a) Based on the orignalA algorithm (pointwise). Not clear, there are misclassified points in the same action time period

(b) Based on the pluralityVA algorithm (sliding windows). It is clear and reduces the number of possible misclassified points

**Fig. 5** Comparison of the visual classification results of arm actions using the Ay signal between the orignalA algorithm (pointwise) and the pluralityVA algorithm (sliding windows)

### 4.1 Experiments protocols

The experimental protocols were performed as follows: first, subject 1 (sub1) performed all 8 actions in a stated order and the associated dataset was collected and saved into a file as *personTrain*. Subsequently the 17 subjects (sub1 ~ sub17) were organized into two groups (one comprising 5 people, the other 12 people), and each of the 2 groups performed all 8 actions 2 times, the first in the prescribed order and the second in a random order. In total 35 datasets (1 + 17*2) were collected (collected separately for each dataset), and organized as different types of training and testing sets as shown in Fig. 6, three datasets were collected from sub1 and two datasets were collected from each of the other subjects (sub2 ~ sub17).

Three types of models (personM, pComM, fComM) were trained based on three corresponding training sets, then tested using three testing sets. From this the following training sets and testing sets were gathered.

- *personTrain*: A personal training set, collected from only one subject such as subject 1 (sub1).
- *partComTain*: A 'partial combination' training set based on 50 samples for each of the 8 classes from subject 2 (sub2) to subject 5 (sub5), appended it to *personTrain.*
- *fullComTrain*: A 'full combination' training set collected from sub1 to sub5.
- *personTest*: Test-set from a personal training subject (sub1).
- *trainSubTest*: Test-set from training subjects that combines 5 training subjects' unseen datasets.
- *newSubTest*: Test-set from new subjects that combines all untrained subjects' datasets (sub6–sub17).

### 4.2 Experimental results

#### 4.2.1 Comparison of originalA and pluralityVA

The three types of testing sets (personTest, trainSubTest, and newSubTest) were classified by each of the 9 models based on the two model design strategies (originalA and pluralityVA), respectively. The experimental results are shown in Fig. 7, which indicates that the classification accuracy is improved using the plurality voting mechanism pluralityVA based on the performance of original algorithms originalA, compared to each of the 9 models. For example, the NN personM for the

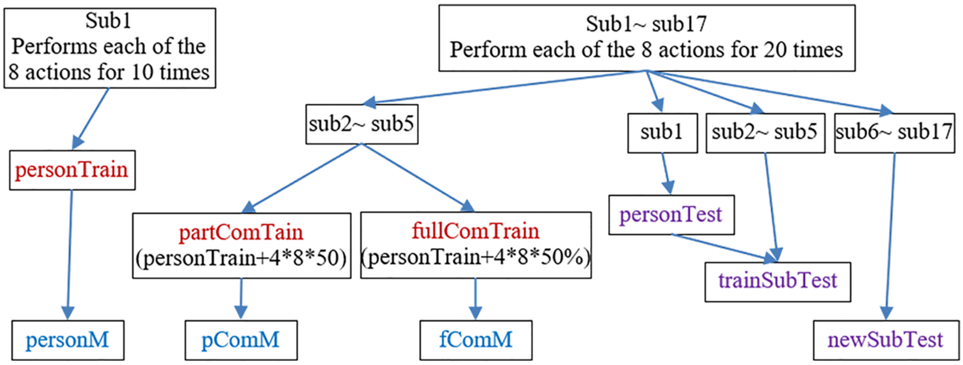


**Fig. 6** Dataset organization. The 35 datasets were collected from 17 subjects and organized as three types of training sets (shown in red) with three types of models (shown in blue) and three types of testing sets (shown in purple)

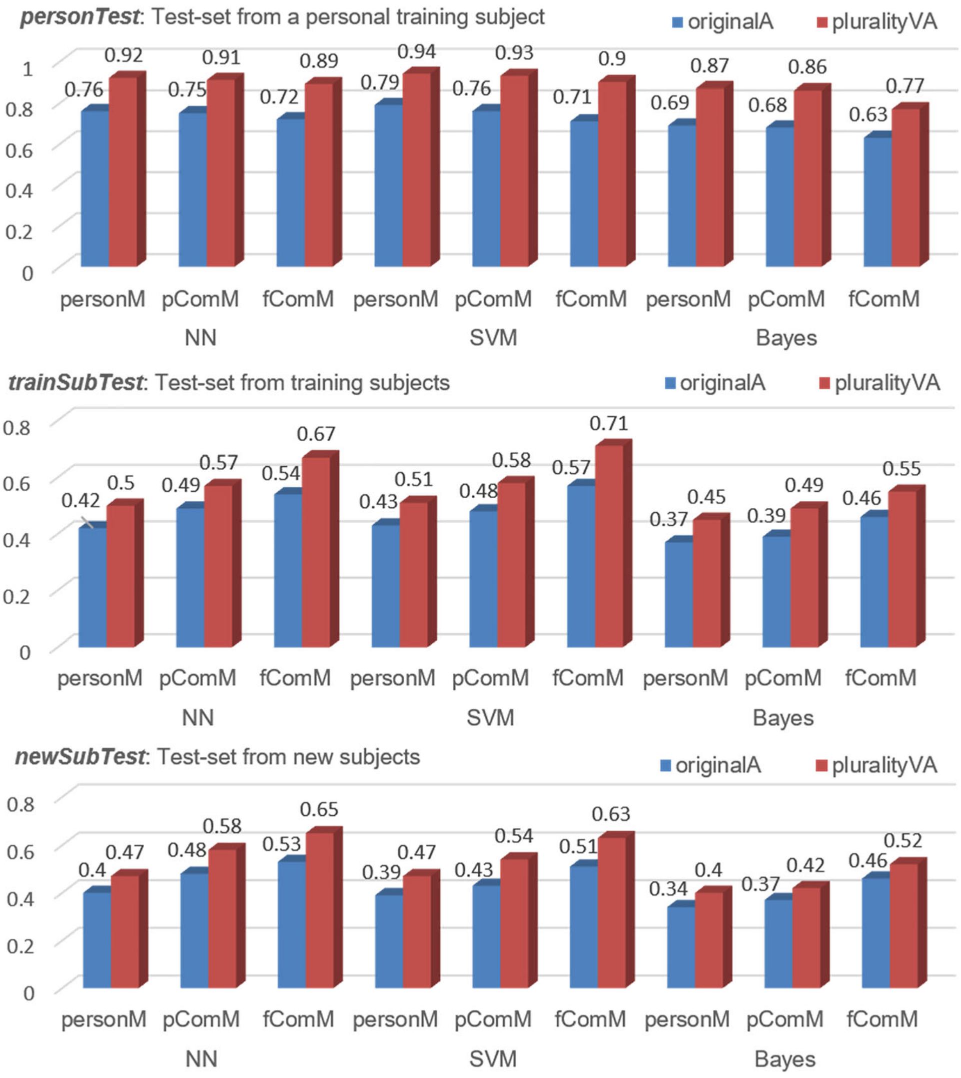


**Fig. 7** Comparison to (1) the activity classification accuracy using the two model design strategies: originalA and pluralityVA for each of the 9 models; (2) the performances of three types of models (personM, pComM, fComM) based on three types of testing sets: personTest(top), trainSubTest(middle), and newSubTest(bottom)

personTest set yields 92% classification accuracy using the pluralityVA vs. 76% classification accuracy using the originalA as shown at the top of Fig. 7, and for the SVM fComM for the trainSubTest set yields 71% classification accuracy using the pluralityVA vs. 57% classification accuracy using the originalA as shown at the middle of Fig. 7. In addition, for the newSubTest dataset, the NN fComM with pluralityVA has the better performance of all 9 models, as shown at the bottom of Fig. 7.

### 4.2.2 Comparison of the three types of models

The Full Combined Model fComM displayed better performance than the model of pComM and personM for the two testing sets trainSubTest and newSubTest, however the personal model personM delivered its best result for the personal testing set personTest. Figure 7 illustrated that the three models all works well for the personalized training subject personTest set, but had poor prediction ability for the unseen datasets (newSubTest and trainSubTest), which suggests training a personalized model for every user could improve the activity classification accuracy and robustness, compared to using a non-personalized model.

To verify the robustness of the personalized modeling approach, subject 1 was asked to randomly perform all 8 activities for 5 min, and then the newly acquired personalized dataset was classified using NN personM with both versions of originalA and pluralityVA. Their classification performance was compared on precision to recall and F-score for each class as shown in Table5.

- Precision is also referred to as the positive predictive value, since it measures the ratio between true positives versus all positives.
- Recall is also known as sensitivity, since it measures the accuracy of the classifier in identifying true positives.
- F1 score takes into account both precision and recall and is based on a balance between the two, with the best score being 1 and worst 0. The F1 score will be low if either precision or recall is low.

Table 5 shows that the personalized modeling method is reasonably robust. For example, testing on a newly collected personalization dataset (label imbalance, see support numbers), the pluralityVA classifier yielded average precision, recall and F1 with a score of 89%, 88% and 88%, respectively. Furthermore, the classification performance of the pluralityVA was significantly higher than that of the originalA classifier, which yielded average precision, recall and F1 with scores of 78%, 75% and 76%, respectively. The reason for this was that pluralityVA is a hybrid model that combined the original model originalA with sliding window segmentation and the plurality voting (PV) algorithm to revise (and improve) the classification results from the originalA for each of the sliding windows, as shown in Fig. 8. This can reduce greatly misclassified samples.

Figure 8 demonstrates that the NN personM model, with the plualityVA scheme, offers very good performance for training subjects using a posture-based adaptive segmented signal. This is because some misclassified samples will be automatically corrected by the plurality voting algorithm, if the number of errored samples is less than the number of correct samples within a given signal segmentation.

### 4.2.3 Optimal model selection

Our experimental results have indicated that the personalized model personM provided the best result of the arm activity classification activities for the personal training subject. For example, Table5 illustrated that four classes (2-run, 4-punch, 6-slap, and 8-still) got the best precision and F1 score (more than 90%); three classes (1-walk, 4-punch, 8-still) obtained the best recall (more than 90%). While, class 7-throw had lowest precision (81%), and class 3-waving suffered the lowest recall (80%) as well as the lowest F1 score (82%). However, the personalized model personM provided the worst result for the unseen subjects.

According to the analysis of the experimental results in Fig. 7, the SVM fComM outperformed the NN and Bayes fComM for the mix-training objects (71% vs. 67% vs. 55%), but the NN fComM outperformed all other models for unseen objects (65% vs.63% vs. 52%). If the system stability for unseen subjects is of concern, then from the results it's clear that NN fComM should be chosen, despite its poor performance. Figure 9 illustrates the challenge of using a common model for activity classification on unseen subjects.

Figure 9 shows experimental results for a 'punching action' scenario, based on the accelerometer Ay signals collected from four subjects. These signals illustrate that all models performed poorly for unseen datasets, which we argue was due to the different subjects have widely different behavior, even though they performed the same action.

As we discussed above, the personalized model training strategy delivers the best performance of the approaches studied. Therefore, we suggest that it is necessary to train a personalized model for each of the new users, especially for business systems, which can improve the accuracy and robustness of activity classification. This result is similar to the Weiss et al. study [29], although we classified different activities using different algorithms.

## 5 Conclusion and future work

In this study, we investigated different techniques for data sampling rate, feature ranking, feature scaling and subfeature sets selection, as well as model selection strategies.

The experimental results demonstrated that the higher sampling rates were not always associated with greater classification accuracy and, in addition, incurred a higher data load which may be important if the activity monitor was embedded into a small wristband device with limited computational resource. We also observed that the RFE feature ranking method worked well compared to others in this study. For example, the 'stand' scaling showed better performance for NN and SVM algorithms; whereas, with the original dataset, it showed very similar results to Bayes. Also, if there are con-

**Table 5** Comparison of classification reports from the two classifiers evaluated using the same testing set collected from a personal training subject

| Classifier | Originala | | | Pluralityva | | | Support numbers |
|---|---|---|---|---|---|---|---|
| | Prec | Recall | F-score | Prec | Recall | F-score | |
| Classes | | | | | | | |
| Walk | 0.73 | 0.89 | 0.80 | 0.85 | 0.93 | 0.89 | 1124 |
| Run | 0.84 | 0.66 | 0.74 | 0.96 | 0.84 | 0.90 | 834 |
| Wave | 0.71 | 0.64 | 0.67 | 0.85 | 0.80 | 0.82 | 715 |
| Punch | 0.78 | 0.58 | 0.67 | 0.90 | 0.92 | 0.91 | 488 |
| fistClench | 0.69 | 0.75 | 0.72 | 0.86 | 0.83 | 0.85 | 597 |
| Slap | 0.87 | 0.77 | 0.81 | 0.90 | 0.88 | 0.89 | 788 |
| Throw | 0.71 | 0.82 | 0.76 | 0.81 | 0.89 | 0.84 | 1002 |
| Still | 0.91 | 0.92 | 0.92 | 0.96 | 0.94 | 0.95 | 1871 |
| Avg/total | 0.78 | 0.75 | 0.76 | **0.89** | **0.88** | **0.88** | 7149 |

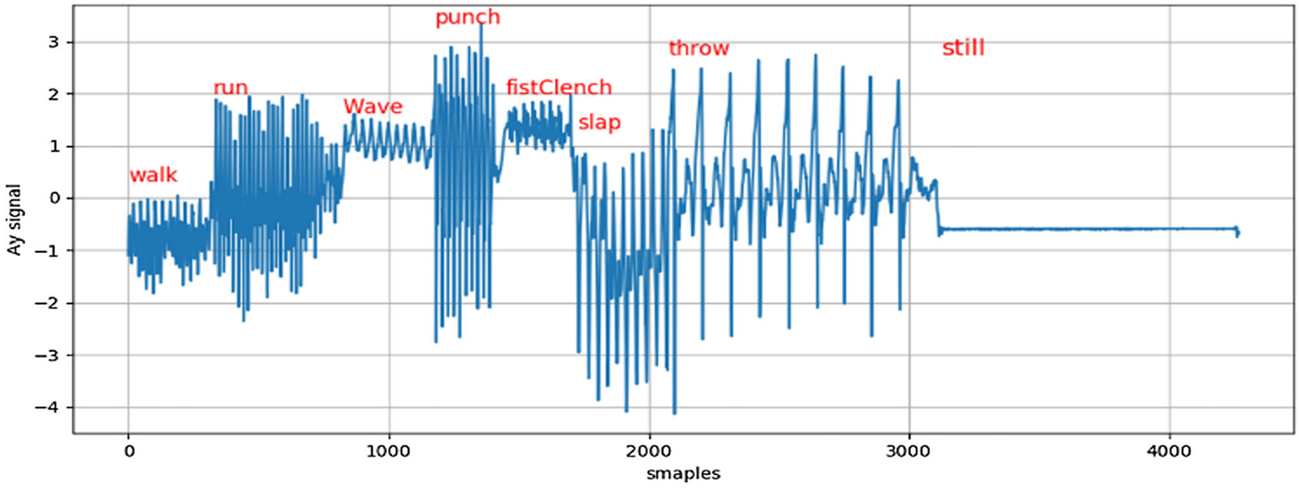
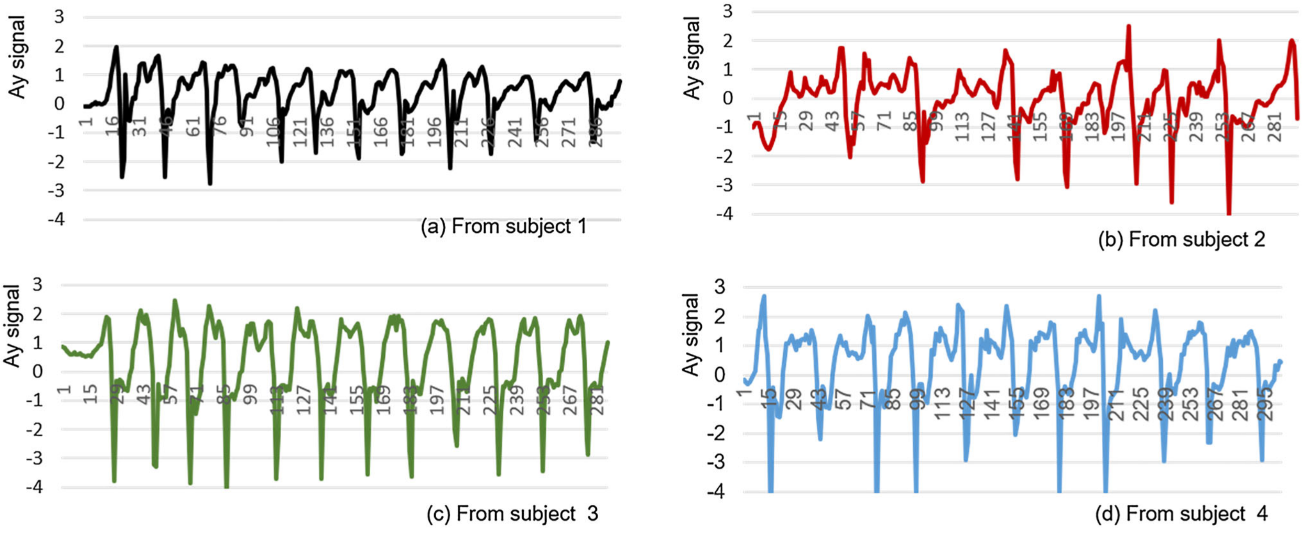


**Fig. 8** Visualized arm activity classification results for sub1 using the NN personM model with a posture-based adaptive signal segmentation algorithm and the pluralityVA approach. The Ay signal and class labels are automatically drawn based on its predicted results by the arm activity classification system



(a) From subject 1

(b) From subject 2

(c) From subject 3

(d) From subject 4

**Fig. 9** The accelerometer Ay signals collected from 4 subjects for a 'punching action' scenario

cerns about the robustness of the system for unseen subjects, NN fComM should be the best choice. In addition, our work determined (maybe unsurprisingly), that a personal model tailored to each individual user would work best. While, in the longer term, it may be possible to build in a form of lifelong learning to the scheme, so it gradually individuals to a given user, there would still remain issues for how to initialize the system without incurring unacceptable time overheads. Also, it would be challenging to incorporate such sophisticated AI into computationally small devices that typify wristbands. Therefore, balancing all such practical issues against performance we concluded that the NN fComM model offered the best solution for arm activity classification systems at this time. Of course, the choice of the classification algorithm can greatly influence the classification accuracy, but, to date, the research literature has yet to offer definitive advice on the best classifier, for human activity recognition, leaving that question as a remaining challenge for the research community [31].

By way of some final comments, during the course of this work we realized just how complicated differentiating between seemingly different human arm activities is, since every person can have a somewhat different behavior for a same action. For example, at times, there was very little difference between light punching and heavy slapping as, even a real-person might have difficult differentiating between these behaviors. From that perspective (and what we said earlier about lifelong learning) this area is challenging to AI research and, in-turn, an excellent vehicle for research which we intend to continue researching. Finally, in writing this paper, we hope to encourage other researches to pick up the gauntlet that some of our findings have thrown up.

**Funding** This work was supported by Hebei Science and Technology Department, Innovation Capability Improvement Plan Project, Grant Numbers 21557611K and Hebei Scientific Research Platform Construction project, Grant Nos. SG20182058 and SZX2020033.

## Declarations

**Conflict of interest** The authors have no competing interests to declare that are relevant to the content of this article.

## References

1. Chelsey D (2019) IDC: Q2 wrist-worn wearable shipments up 29%. Circuits Assembly Online Magazine. https://circuitsassembly.com/ca/editorial/menu-news/32081-idc-q2-wrist-worn-wearable-shipments-up-29.html. Accessed 26 June 2021
2. Ramon L (2019) Wrist-worn wearables maintain a strong growth trajectory in Q2 2019, According to IDC. https://www.businesswire.com/news/home/20190912005263/en/Wrist-Worn-Wearables-Maintain-Strong-Growth-Trajectory-Q2. Accessed 26 June 2021
3. Aroganam G, Manivannan N, Harrison D (2019) Review on wearable technology sensors used in consumer sport applications. Sensors 19(9):1983
4. Chen C, Kehtarnavaz N, Jafari R (2014) A medication adherence monitoring system for pill bottles based on a wearable inertial sensor. In: Engineering in medicine and biology society (EMBC), 36th annual international conference of the IEEE, pp 4983–4986
5. Chen C, Liu K, Jafari R, Kehtarnavaz N (2014) Home-based senior fitness test measurement system using collaborative inertial and depth sensors. In: Engineering in medicine and biology society (EMBC), international conference of the IEEE, pp 4135–4138
6. Cornacchia M, Ozcan K, Zheng Y, Velipasalar S (2017) A survey on activity detection and classification using wearable sensors. IEEE Sens J 17(2):386–403
7. Mukhopadhyay SC (2015) Wearable sensors for human activity monitoring: a review. IEEE Sens J 15(3):1321–1330
8. Rostami M, Berahmand K, Nasiri E, Forouzande S (2021) Review of swarm intelligence-based feature selection methods. Eng Appl Artif Intell 100:104210
9. Tubishat M, Ja'afar S, Alswaitti M, Mirjalili S, Idris N, Ismail MA, Omar MS (2021) Dynamic salp swarm algorithm for feature selection. Expert Syst Appl 164:113873
10. Quiroz JC, Banerjee A, Dascalu SM, Lau SL (2017) Feature selection for activity recognition from smartphone accelerometer data. Intell Autom Soft Comput 1–9
11. Xue B, Zhang M, Browne WN, Yao X (2016) A survey on evolutionary computation approaches to feature selection. IEEE Trans Evol Comput 20(4):606–626
12. Rodgers J, Nicewander WA (1988) Thirteen ways to look at the correlation coefficient. Am Stat 42(1):59–66
13. Ma L, Li M, Gao Y, Chen T, Ma X, Qu L (2017) A novel wrapper approach for feature selection in object-based image classification using polygon-based cross-validation. IEEE Geosci Remote Sens Lett 14(3):409–413
14. Yan K, Zhang D (2015) Feature selection and analysis on correlated gas sensor data with recursive feature elimination. Sens Actuators B Chem 212:353–363
15. Kukreja SL, Löfberg J, Brenner MJ (2006) A least absolute shrinkage and selection operator (LASSO) for nonlinear system identification. IFAC Proc Vol 39(1):814–819
16. Pedregosa F, Varoquaux G, Gramfort A, Michel V, Thirion B, Grisel O, Blondel M, Prettenhofer P, Weiss R, Dubourg V, Vanderplas J (2011) Scikit-learn: machine learning in Python. J Mach Learn Res 12(Oct):2825–2830
17. Grus J (2015) Data science from scratch: first principles with Python. O'Reilly Media, Inc.
18. Ioffe S, Szegedy C (2015) Batch normalization: accelerating deep network training by reducing internal covariate shift. In: International conference on machine learning, pp 448–456
19. Busso C, Mariooryad S, Metallinou A, Narayanan S (2013) Iterative feature normalization scheme for automatic emotion detection from speech. IEEE Trans Affect Comput 4(4):386–397
20. Mohamad IB, Usman D (2013) Standardization and its effects on K-means clustering algorithm. Res J Appl Sci Eng Technol 6(17):3299–3303
21. Gao L, Bourke AK, Nelson J (2014) Evaluation of accelerometer based multi-sensor versus single-sensor activity recognition systems. Med Eng Phys 36(6):779–785
22. Tchuente F, Baddour N, Lemaire ED (2020) Classification of aggressive movements using smartwatches. Sensors 20(21):6377
23. Eibe F, Hall MA, Ian H (2016) The WEKA Workbench. Online appendix for "data mining: practical machine learning tools and techniques", 4th edn. Morgan Kaufmann
24. Kotsiantis SB (2007) Supervised machine learning: a review of classification techniques. Informatica 31:249–268

25. Yang J (2009) Toward physical activity diary: motion recognition using simple acceleration features with mobile phones. In: Proceedings of the 1st international workshop on Interactive multimedia for consumer electronics, pp 1–10
26. Namsrai E, Munkhdalai T, Li M, Shin J, Namsrai O, Ryu KH (2013) A feature selection-based ensemble method for arrhythmia classification. J Inf Process Syst 9(1):31–40
27. Ngo TT, Makihara Y, Nagahara H et al (2015) Similar gait action recognition using an inertial sensor. Pattern Recogn 48(4):1289–1301
28. Jadhav SD, Channe HP (2016) Comparative study of K-NN, naive Bayes and decision tree classification techniques. Int J Sci Res (IJSR) 5(1):1842–1845
29. Weiss GM, Lockhart J (2012) The impact of personalization on smartphone-based activity recognition. In: Workshops at the twenty-sixth AAAI conference on artificial intelligence
30. Tapia EM, Intille SS, Haskell W, Larson K, Wright J, King A, Friedman R (2007) Real-time recognition of physical activities and their intensities using wireless accelerometers and a heart rate monitor. In: 2007 11th IEEE international symposium on wearable computers, pp 37–40
31. Ferrari A, Micucci D, Mobilio M, Napoletano P (2021) Trends in human activity recognition using smartphones. J Reliab Intell Environ 7(3):189–213
32. Burns DM, Whyne CM (2020) Personalized activity recognition with deep triplet embeddings. http://arxiv.org/abs/2001.05517
33. Axiviy X3 accelerometers website, https://axivity.com/. Accessed 20 June 2021
34. Murphy C (2017) Choosing the most suitable MEMs accelerometer for your application—part 2. Analog Dialogue 51(11):1–6
35. Liu H, Motoda H (1999) Feature extraction construction and selection: a data mining perspective. J Am Stat Assoc 94(448):014004
36. Ferrari A, Micucci D, Marco M, Napoletano P (2019) Hand crafted features vs residual networks for human activities recognition using accelerometer. In: Proceedings of the IEEE international symposium on consumer technologies (ISCT), pp 153–156.
37. Quigley B, Donnelly M, Moore G, Galway L (2018) A comparative analysis of windowing approaches in dense sensing environments. In: Multidisciplinary digital publishing institute proceedings, vol 2, no 19, p 1245
38. Banos O, Galvez JM, Damas M, Pomares H, Rojas I (2014) Window size impact in human activity recognition. Sensors 14(4):6474–6499

**Publisher's Note** Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.