

# Extreme learning machine-based device displacement free activity recognition model

Yiqiang Chen · Zhongtang Zhao · Shuangquan Wang · Zhenyu Chen

Published online: 7 February 2012  
© Springer-Verlag 2012

**Abstract** Activity recognition based on mobile device is an important aspect in developing human-centric pervasive applications like gaming, industrial maintenance and health monitoring. However, the data distribution of accelerometer is heavily affected by varying device locations and orientations, which will degrade the performance of recognition model. To solve this problem, we propose a fast, robust and device displacement free activity recognition model in this paper, which integrates principal component analysis (PCA) and extreme learning machine (ELM) to realize location-adaptive activity recognition. On the one hand, PCA is employed to reduce the dimensionality of feature space and extract robust features for recognition. On the other hand, in the online phase ELM is applied to classify the activity and adapt the recognition model to new device locations based on high confident recognition results in real time. Experimental results show that, with

robust features and fast adaptation capability, the proposed model can adapt the classifier to new device locations quickly and obtain good recognition performance.

**Keywords** Extreme learning machine · Activity recognition · Principal component analysis · Machine learning

## 1 Introduction

Automatically recognizing motion activities, such as standing still, walking, running, etc, allows many applications in areas such as healthcare, elderly care and energy expenditure estimation. Because of its inherent property, accelerometers are most often employed in human activity recognition to measure the movement characters of human body. With the development of micro-electrical mechanical systems (MEMS), the accelerometers are miniaturized so that they can be embedded into small mobile devices.

Compared with traditional wearable activity recognition (Roggen et al. 2011) which fixes accelerometers on specific locations of human body, activity recognition based on mobile device faces a new problem of varying device locations and orientations because the users can put the device at different places as they prefer, such as hand, pocket, bag, etc. When the device is deployed at varying locations but with the same orientation, the embedded accelerometer may exert different forces because the movement patterns of different body parts are distinct, even when the user is doing the same activity. When the device is deployed with varying orientations but at the same location, the accelerometer will exert the same force. However, the sensor readings of the three axes will be different because the decomposed force along the

---

Y. Chen (✉) · Z. Zhao · S. Wang · Z. Chen  
Institute of Computing Technology,  
Chinese Academy of Sciences, Beijing 100190, China  
e-mail: yqchen@ict.ac.cn

Z. Zhao  
e-mail: zhaozhongtang@ict.ac.cn

S. Wang  
e-mail: wangshuangquan@ict.ac.cn

Z. Chen  
e-mail: chenchenyu@ict.ac.cn

Z. Zhao  
Graduate University of the Chinese Academy of Sciences,  
Beijing 100049, China

Z. Zhao  
Zhengzhou Institute of Aeronautical Industry Management,  
Zhengzhou 450015, China

coordinates of the device are related with the angles between the synthesized force and the three axes.

In this paper, a fast, robust and device displacement free activity recognition model is proposed to deal with the problem of varying device locations and orientations. First, the readings along three axes are synthesized and the magnitude of synthesized acceleration is used to extract features. This can eliminate the orientation difference of the mobile device, at the cost of losing direction information. Second, based on a lot of features, principal component analysis (PCA) is used to eliminate noise feature, extract robust features for recognition. Third, extreme learning machine (ELM) is used to adapt the recognition model to new locations in the online phase. The high confident recognition results will be selected and added into the training dataset. Then, the recognition model will be retrained through taking the advantage of fast learning speed of ELM. Experimental results show the validity and high performance of the proposed model.

The rest of this paper is organized as follows. In Sect. 2, some related work will be illustrated. Section 3 introduces the proposed model. Experiments and results will be presented in Sect. 4. Finally, we conclude this paper in Sect. 5.

## 2 Related work

As to varying device orientations, Reddy et al. (2008) propose to recognize transportation modes using the features extracted from the series of acceleration magnitude, which is the value of acceleration synthesization of three axes. Mizell (2003) shows that the average of the accelerometer signal over a reasonable time period can produce a good estimation of the gravity-related component, which in turn can be used to estimate the vertical and horizontal component of dynamic acceleration. Yang (2009) uses these two methods to recognize the user's activities including sitting, standing, walking, running, driving and bicycling. Experimental results show that acceleration decomposition-based method performs a little better than acceleration synthesization-based method. However, Wang et al. (2010) show that the acceleration synthesization-based method outperforms the acceleration decomposition-based method for recognizing six typical transportation modes, including biking, busing, driving, stay, taking subway and walking. In addition, they demonstrate that the gravity estimation error will degrade the performance of acceleration decomposition-based method.

As to varying device locations, Kunze et al. (2005) propose a location recognition method which firstly identifies time periods where the user is walking and then

leverages the specific characters of walking motion to determine the location of the body-worn sensor, including wrist, head, trouser's pocket and breast pocket. However, for mobile device-based activity recognition, this method may not be feasible because it requires well-defined fixed positions. Kunze and Lukowicz (2008) demonstrate that acceleration caused by rotational motion is location sensitive and combining gyroscope with accelerometer is helpful to reduce the sensitivity.

To address the varying location and orientation issues simultaneously, Sun et al. (2010) attempt to extract features that are independent from or insensitive to orientation change. They build a support vector machine (SVM) classifier for all physical activities in all pocket positions. However, they assume that most of the people put their mobile phones in one of their six pockets around the waist, which simplifies this problem to a predefined scenario and may not be feasible in the real daily life.

As to the recognition model adaption, Lai et al. (2010) uses the body posture analysis flowchart to judge which bodily behavior is occurring. In order to adapt the detection algorithm to individual habits, the authors apply subtractive clustering method (SCM) to calculate the center position of the habitual inclination angles for each pose, which is used as the threshold for pose judgement. Zhao et al. (2011) propose a transfer learning-based algorithm which integrates a decision tree and the k-means clustering for personalized activity recognition model adaptation. However, above papers only take the model adaption across different users into consideration. To our knowledge, there is no existing work on the recognition model adaptation across different locations.

ELM is an efficient and practical learning mechanism for the single-hidden-layer feed-forward neural networks. The recent studies reveal that ELM can be successfully applied to fuzzy integral determination (Wang et al. 2011), fuzzy rule learning (Jun et al. 2011), image recognition (Chacko et al. 2011), and other real-world applications (Huang et al. 2011). The study on generalization performance of learning system has caused wide public concern over the recent years. One novel idea (Wang and Dong 2009; Wang et al. 2011) is to improve the generalization ability by maximizing the uncertainty inherent in the learning system. Such illumination is indeed helpful and useful to the discussion of ELM's generalization. An ELM-based activity recognition algorithm is presented in the current manuscript. The latest explorations also provide some new human-centric pervasive applications with machine learning technologies (Xiao et al. 2011; Zhang et al. 2011). In this paper, based on the characteristics of ELM, we built a fast and robust model to classify the daily activity.

### 3 The ELM-based device displacement free activity recognition model

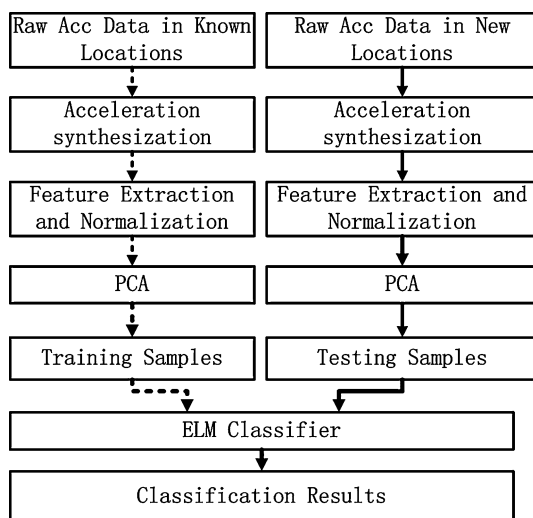
The proposed model contains two steps:

**Step 1** Offline classification model construction and online activity recognition, as shown in Fig. 1. For offline classification model construction, firstly the readings of three axes are synthesized into magnitude series to get rid of orientation dependence. Statistic and frequency-domain features are extracted from magnitude series of synthesized acceleration. Then, PCA is used to reduce the dimension of feature space and reserve useful and robust features. With the characters of fast learning speed and high generalization capability, ELM is used to build the classification model. For online activity recognition, the unlabeled testing sample is generated with the same method as that used in the offline phase. Then the sample is classified by the ELM classifier and the classification result can be obtained.

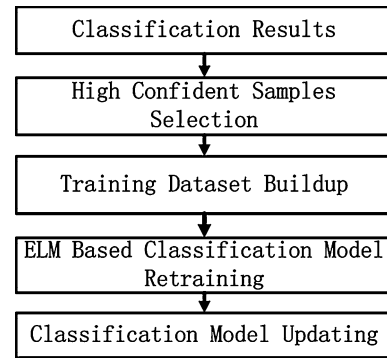
**Step 2** Activity recognition model retraining and updating, as illustrated in Fig. 2. Based on the classification results, the confidence that a sample is correctly classified is estimated. The samples whose confidences are greater than a threshold,  $\eta$ , are selected to build up new training dataset, together with the training samples in Step 1. Then, the ELM classification model will be retrained and updated. As the new training samples may be collected from new device location, the updated model would adapt to unknown locations gradually.

#### 3.1 Acceleration synthesisization

Accelerometer detects and transforms changes in capacitance into an analog output voltage, which is proportional



**Fig. 1** The overview of ELM-based activity recognition. The dotted line represents the offline classification model construction and the solid line represents the online activity recognition



**Fig. 2** Process of activity recognition model retraining and updating

to acceleration. For triaxial accelerometer, the output voltages can be mapped into acceleration along three axes,  $a_x$ ,  $a_y$ ,  $a_z$ . As  $a_x$ ,  $a_y$ ,  $a_z$  are the orthogonal decompositions of real acceleration, the magnitude of synthesized acceleration can be expressed as:

$$a = \sqrt{a_x^2 + a_y^2 + a_z^2} \quad (1)$$

$a$  is the magnitude of real acceleration, but has no directional information. Therefore, the acceleration magnitude-based activity recognition model is orientation independent.

#### 3.2 Acceleration feature extraction and normalization

Based on the acceleration magnitude series, 17 statistic features (Figo et al. 2010) are extracted from a sliding window of 256 samples with 50% overlapping between consecutive windows. These features are mean, standard deviation, energy, mean-crossing rate, maximum value, minimum value, first quartile, second quartile, third quartile, four amplitude statistic features and four shape statistic features of the power spectral density (PSD) (Wang 2004). In addition, based on FFT transformation of the acceleration magnitude series, all frequency components from 1 to 128 Hz are extracted and added into the feature vector, totally 145 features.

PSD is defined as the Fourier transform of the autocorrelation of the time series signal and describes the energy distribution of a signal in the frequency domain. The amplitude statistics is defined as:

$$\text{Amplitude: } \mu_{\text{amp}} = \frac{1}{N} \sum_{i=1}^N C(i) \quad (2)$$

$$\text{Std: } \sigma_{\text{amp}} = \sqrt{\frac{1}{N} \sum_{i=1}^N [C(i) - \mu_{\text{amp}}]^2} \quad (3)$$

$$\text{Skewness: } \gamma_{\text{amp}} = \frac{1}{N} \sum_{i=1}^N \left[ \frac{C(i) - \mu_{\text{amp}}}{\sigma_{\text{amp}}} \right]^3 \quad (4)$$

$$\text{Kurtosis: } \beta_{\text{amp}} = \frac{1}{N} \sum_{i=1}^N \left[ \frac{C(i) - \mu_{\text{amp}}}{\sigma_{\text{amp}}} \right]^4 - 3 \quad (5)$$

where  $C(i)$  is the PSD magnitude for the  $i$ th frequency bin, and  $N$  is the number of the frequency bins. Similarly, the shape statistics is defined as:

$$\text{Mean: } \mu_{\text{shape}} = \frac{1}{S} \sum_{i=1}^N iC(i) \quad (6)$$

$$\text{Std: } \sigma_{\text{shape}} = \sqrt{\frac{1}{S} \sum_{i=1}^N (i - \mu_{\text{shape}})^2 C(i)} \quad (7)$$

$$\text{Skewness: } \gamma_{\text{shape}} = \frac{1}{S} \sum_{i=1}^N \left( \frac{i - \mu_{\text{shape}}}{\sigma_{\text{shape}}} \right)^3 C(i) \quad (8)$$

$$\text{Kurtosis: } \beta_{\text{shape}} = \frac{1}{S} \sum_{i=1}^N \left( \frac{i - \mu_{\text{shape}}}{\sigma_{\text{shape}}} \right)^4 C(i) - 3 \quad (9)$$

where  $S = \sum_{i=1}^N C(i)$ .

To eliminate the scaling effects among different features, all the features are normalized using the z-score normalization algorithm (Han and Kamber 2000).

### 3.3 PCA-based dimension reduction

Suppose the normalized feature matrix  $X$  is of dimension  $M \times N$ , where  $M$  is the number of samples,  $N$  is the number of extracted features. The PCA transformation can be represented as  $Y_{M \times K} = X_{M \times N} \cdot B_{N \times K}$ , where  $K < N$ .  $B = [b_1, b_2, \dots, b_k]$  is a set of basis vectors which are linearly independent and orthogonal, and can be calculated as follows (Wang 2004):

*Step 1* Calculate the covariance matrix of the original feature matrix as

$$S = \frac{1}{M} (X - \mu)^T (X - \mu) \quad (10)$$

where  $\mu$  is the mean vector of the feature set. Because the normalized feature vectors have zero mean, the calculation of the covariance matrix can be simplified as

$$S = \frac{1}{M} X^T X \quad (11)$$

*Step 2* Calculate the eigenvalues and the corresponding eigenvectors of the covariance matrix  $S$ .

*Step 3* Suppose the eigenvectors are sorted in terms of their eigenvalues and those corresponding to the largest  $K$  eigenvalues are chosen to construct the transformation matrix  $R = [b_1, b_2, \dots, b_k]$  of corresponding eigenvalues  $\lambda_1 \geq \lambda_2 \geq \dots \geq \lambda_K$ . The choice of  $K$  can be determined as

$$\frac{\sum_{i=1}^K \lambda_i}{\sum_{i=1}^M \lambda_i} \geq 1 - \eta \quad (12)$$

where  $\eta$  denotes the loss of energy.

In this paper, PCA is used (1) to denoise the samples; (2) to reduce the dimensionality of the samples and (3) to extract location-insensitive features for later model construction or online recognition.

## 4 ELM-based recognition model construction and location adaption

### 4.1 ELM-based classifier

ELM is a recent neural network algorithm, which is known to achieve good performance in complex problems as well as reduce the computation time compared with other machine learning algorithms (Huang et al. 2004a, b, 2006, 2011). The ELM algorithm does not train the input weights or the biases of neurons, but it acquires the output weights by using the norm least-squares solution and Moore–Penrose inverse of a general linear system (Feng et al. 2009; Huang and Chen 2007, 2008; Huang et al. 2010). By finding the node giving the maximum output value, we decide the final result.

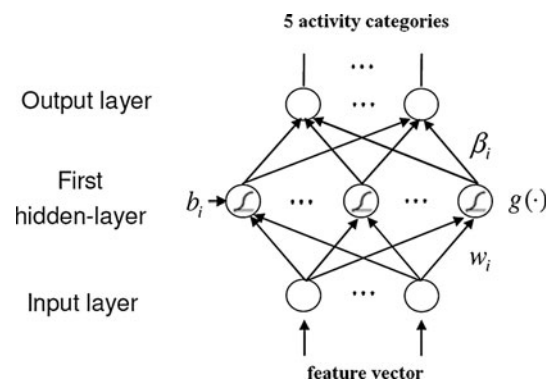
Figure 3 shows the network structure of ELM with a single hidden layer used for our experiments. We used 50 hidden neurons and the sigmoid activation function.

The learning phase for the ELM with a single hidden layer can be summarized as Fig. 4.

In the testing phase, for a testing sample  $x$ , the outputs can be calculated as follows:

$$TY_{1 \times m} = [g(w_1 \cdot x + b), \dots, g(w_{\tilde{N}} \cdot x + b)]_{1 \times \tilde{N}} \cdot \beta_{\tilde{N} \times m} \quad (13)$$

$TY = [\alpha_1, \alpha_2, \dots, \alpha_m]$ ,  $m$  is the number of output nodes, which equals the number of classes in classification problem. Then, the ELM selects the maximum value of  $TY$  and assigns its corresponding index,  $j$ , as the class label of the test sample. We can calculate the sample's confidence to the assigned class by the following steps:



**Fig. 3** The network structure of ELM

---

**the ELM Algorithm**

---

**Input:** Given a training set

$\mathbb{X} = \{(x_i, t_i) | x_i \in R^n, t_i \in R^m, i = 1, 2, \dots, N\}$ , activation function  $g(x)$ , and hidden node number  $\tilde{N}$ .

**Output:** the output weight  $\beta$ .

1. Randomly assign input weight  $w_i$  and bias  $b_i$ ,  $i = 1, 2, \dots, \tilde{N}$ .

2. Calculate the hidden layer output matrix  $H$ :

$$H = \begin{pmatrix} g(w_1 \cdot x_1 + b_1) & \cdots & g(w_{\tilde{N}} \cdot x_1 + b_{\tilde{N}}) \\ \vdots & \cdots & \vdots \\ g(w_1 \cdot x_N + b_1) & \cdots & g(w_{\tilde{N}} \cdot x_N + b_{\tilde{N}}) \end{pmatrix}_{N \times \tilde{N}}. \quad (13)$$

3. Calculate the output weight  $\beta = H^\dagger T$ , where  $H$  is the Moore-Penrose generalized inverse of matrix,  $T = [t_1, \dots, t_N]^T$ .
- 

**Fig. 4** The ELM algorithm

$$TY_i = TY_i - \min(TY_i) \quad (14)$$

$$\text{confidence} = \frac{\max(TY_i)}{\sum TY_i}, \quad i = 1, 2, \dots, m \quad (15)$$

#### 4.1.1 Recognition model adaptation to new locations

On the online phase, the user may place the mobile device to new locations. In order to adapt to new locations, the recognition model is retrained based on high confident recognition results by taking advantage of the fast learning speed of ELM. The model adaptation process contains following three steps:

1. During recognition, the test samples whose classification confidence larger than threshold  $\eta$  will be reserved and added into the training dataset.
2. When the number of new training samples exceeds the predefined threshold, ELM is used to retrain the recognition model based on whole training dataset.
3. Replacing the old recognition model with the new model.

## 5 Experimental evaluations

### 5.1 Data collection

This paper aims to propose a common method for location-adaptive activity recognition, based on triaxial accelerometer embedded in any mobile devices. The device used in this paper is a white box made by our hardware engineer. A XSens MTx sensor,<sup>1</sup> which contains a triaxial accelerometer, a triaxial gyroscope and a compass, is embedded in the box. This device is only used for data collection. All data

is transmitted to a PC and all the data preprocessing and analysis are done on the PC. The sampling rate of accelerometer is set to 100 Hz. Four participants with varying age and gender are recruited to perform five daily activities, including staying still, walking, running, going upstairs and going downstairs. During data collection, the sensor box can be placed at the subject's hands (left and right), chest pockets (left and right) and trousers pockets (left and right), as shown in Fig. 5. For each location, all participants will do each activity for 8–10 min. To ensure the quality of collecting samples, every two participants are organized into one group. When one is performing the activities, the other is recording the corresponding information, such as activity type, start time, terminal time and so on. Table 1 shows the number of samples obtained for each activity.

### 5.2 Experimental results

#### 5.2.1 Classifier performance comparison

The ELM algorithm is compared with two popular classifiers, SVM and nearest neighbor (NN), to evaluate its performance. Based on the dataset obtained in Sect. 5.1, 10-fold cross-validation test is done for each classifier. The number of ELM's hidden neuron nodes is set to 50 and the activation function is 'sigmoid'. The training time (s), testing time (s), training accuracy and testing accuracy are listed in Table 2.

As can be seen from Table 2, ELM obtains the highest accuracy among them, which is about 5 and 15% higher than that of SVM and NN, respectively. In addition, the training time and testing time of ELM are much less than that of SVM and NN. For 10-fold cross-validation test of 8,473 samples, the total training time of ELM is 14.98 s, while SVM consumes 197 s. This indicates that ELM has a much faster training speed than SVM. The total testing time of ELM is 0.91 s, which is obviously less than that of SVM and NN.

#### 5.2.2 Cross-location recognition without dimension reduction and model adaptation

In order to evaluate the performance degradation of recognition model when the device location is changed, an experiment of cross-location recognition is done. For each location, a classifier is learned and is tested on all the locations.

Table 3 shows the confusion matrix of cross-location recognition using ELM. The first column of Table 3 is training location and the first row is testing location. From Table 3 we can see that the ELM performs well at the same location but performs poor at the others. The maximum reduction is about 30% when the model learned from chest

<sup>1</sup> "XSens Technologies B.V.", <http://www.xsens.com>.





**Fig. 5** Location information: **a** in the right trousers pocket; **b** in the right hand; **c** in the right chest pocket; **d** in the left trousers pocket; **e** in the left hand; **f** in the left chest pocket

pocket is applied to hand. The minimum is about 6% when the model learned from hand is applied to chest pocket.

In Table 4, the confusion matrix of cross-location recognition using SVM is listed. Compared with Table 3, we can see that ELM and SVM obtain approximate accuracies at known locations, but the accuracies of SVM at new locations decrease a lot and much lower than that of ELM.

As can be seen from Table 5, NN always gets 100% accuracy at same locations because it is an instance-based lazy learning algorithm and the training dataset and testing dataset are same. Compared with Table 3, we can see that NN also performs worse than ELM at new locations.

### 5.2.3 Dimension reduction by PCA

In Sect. 3.2, we have extracted 145 features. Of these features, some are useful and some may be noise. In order to eliminate the noise features and extract robust ones, PCA is used in our experiments. Figure 6 shows the correlation between the preserved energy and the number of principal components in the transform space. From Fig. 6 we can see that, if the loss of energy is set as 0.05, 30 principal components are enough. Thus, for simplification, the number of principal components is set as 30 in the following experiments.

Seen from the confusion matrix of cross-location recognition using ELM illustrated in Tables 3 and 6, we can easily find that all the recognition accuracies of cross locations are almost increased about 2–8%. This proves that PCA indeed eliminates some noise features and the extracted features are more robust than the original ones.

**Table 1** Activity sample information

Activity name	Label	Number of sample
Staying still	1	1,751
Downstairs	2	1,579
Walking	3	1,846
Running	4	1,644
Upstairs	5	1,653
Total samples		8,473

**Table 2** Performance of three classifiers

	ELM	SVM	NN
Train time	<b>14.98</b>	197.00	\
Test time	<b>0.91</b>	16.60	29.10
Accuracy (%)	79.68	75.01	64.74

**Table 3** Confusion matrix of cross-location recognition by ELM

	Trousers pockets (%)	Chest pockets (%)	Hand (%)
Trousers pockets	<b>81.01</b>	68.66	62.96
Chest pockets	64.39	<b>88.13</b>	58.57
Hand	50.24	68.62	<b>74.97</b>

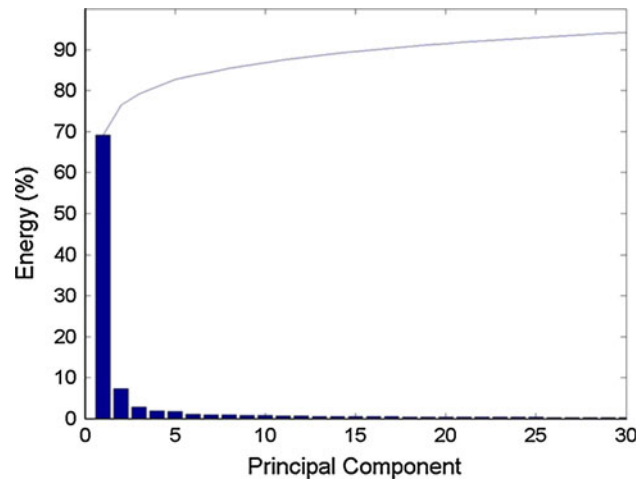
Table 7 shows that the accuracies of applying one location's model to other locations using SVM increase about 14–25%. But the accuracies at the same location decrease about 3–12%. However, in Tables 6 and 7, we can

**Table 4** Confusion matrix of cross-location recognition by SVM

	Trousers pockets (%)	Chest pockets (%)	Hand (%)
Trousers pockets	<b>87.39</b>	41.41	37.19
Chest pockets	32.02	<b>89.01</b>	48.75
Hand	36.70	49.26	<b>87.81</b>

**Table 5** Confusion matrix of cross-location recognition by NN

	Trousers pockets (%)	Chest pockets (%)	Hand (%)
Trousers pockets	<b>100</b>	57.14	41.61
Chest pockets	59.88	<b>100</b>	61.10
Hand	53.76	66.11	<b>100</b>

**Fig. 6** The effect of PCA

see that ELM outperforms SVM not only at known locations, but also at new locations. The similar situation can be found in Tables 5 and 8, where NN is used as classifier. Above experimental results demonstrate that ELM performs best and has strongest generalization capability among these three classification algorithm.

From Table 3 versus Table 6, and Table 4 versus Table 7 we can see that, after dimension reduction by PCA, the recognition accuracy on most unknown locations are increased, which proves that the extracted features by PCA can suit different device deployment locations better. However, this improvement is not obvious in Tables 5 and 8, which cause some confusion. The possible reason maybe that, compared with ELM and SVM, NN has the worst generalization capability and thus confuses the samples of different locations. The recognition accuracy of NN at the same location is meaningless because one sample is always the nearest neighbor of itself.

**Table 6** Confusion matrix of cross-location recognition after dimension reduction by ELM

	Trousers pockets (%)	Chest pockets (%)	Hand (%)
Trousers pockets	<b>82.90</b>	71.59	69.12
Chest pockets	65.51	<b>90.25</b>	63.67
Hand	58.60	66.93	<b>79.73</b>

**Table 7** Confusion matrix of cross-location recognition after dimension reduction by SVM

	Trousers pockets (%)	Chest pockets (%)	Hand (%)
Trousers pockets	<b>75.46</b>	64.81	58.39
Chest pockets	57.48	<b>85.97</b>	62.37
Hand	52.47	66.64	<b>79.10</b>

**Table 8** Confusion matrix of cross-location recognition after dimension reduction by NN

	Trousers pockets (%)	Chest pockets (%)	Hand (%)
Trousers pockets	<b>100</b>	48.83	42.54
Chest pockets	44.52	<b>100</b>	57.64
Hand	51.02	59.82	<b>100</b>

In the experiments from Sect. 5.2.1 to 5.2.3, we can see that compared with SVM and NN, ELM has the fastest training and testing speed, the strongest generalization capability. Therefore, we will use ELM as our classifier in the following experiments.

#### 5.2.4 Cross-location model adaptation

In this section, the experiments aim to test the ELM model's adaptability to new locations. Three locations, hand, chest pocket and trousers pocket, are presented as A, B and C. The datasets of these locations are represented as  $Data_A$ ,  $Data_B$  and  $Data_C$ , respectively. Each dataset is randomly divided into two parts, which are represented as  $Data_{A1}$  and  $Data_{A2}$ ,  $Data_{B1}$  and  $Data_{B2}$ , and  $Data_{C1}$  and  $Data_{C2}$ .

Without loss of generality, we first assume that A and B are known locations and C is a new one.  $Train_{AB}$ , which equals  $Data_{A1} \cup Data_{B1}$ , is used to train an ELM model named as initial model.  $Data_{C1}$  is used to adapt the initial model to a new one.  $Test_{AB}$ , which equals  $Data_{A2} \cup Data_{B2}$ , is used to test the two model's classification capability on the known locations.  $Data_{C2}$  is used to test the two model's classification capability on the new location. For the initial model and each test sample in  $Data_{C1}$ , if the classification

**Table 9** Recognition results on the known locations before and after adaptation

	Before adaptation	After adaptation
Train data	$Train_{AB}$	$Train_{AB} + HConf_{C1}$
Test data	$Test_{AB}$	$Test_{AB}$
Accuracy (%)	80.93	80.9

A and B are the known locations and C is the new location

confidence,  $\eta$ , is larger than 0.5, it will be added into a new dataset,  $HConf_{C1}$ . Then, using all samples in  $Train_{AB}$  and  $HConf_{C1}$ , a new recognition model can be retrained.

The performances of the initial model and the new model on the known locations are shown in Table 9. We can see that after model adaptation, the new model almost has the same classification capability as the initial model. The performances of the initial model and the new model on the new location are shown in Table 10. We can see that after adaptation accuracy is improved about 6%.

When B and C are assumed as known locations and A is as the new location, experiment results are shown in Tables 11, 12. After adaptation, the accuracy is improved about 7%.

**Table 10** Recognition results on the new location before and after adaptation

	Before adaptation	After adaptation
Train data	$Train_{AB}$	$Train_{AB} + HConf_{C1}$
Test data	$Data_{C2}$	$Data_{C2}$
Accuracy (%)	66.52	72.99

A and B are the known locations and C is the new location

**Table 11** Recognition results on the known locations before and after adaptation

	Before adaptation	After adaptation
Train data	$Train_{BC}$	$Train_{BC} + HConf_{A1}$
Test data	$Test_{BC}$	$Test_{BC}$
Accuracy (%)	73.76	72.02

B and C are the known locations and A is the new location

**Table 12** Recognition results on the new location before and after adaptation

	Before adaptation	After adaptation
Train data	$Train_{BC}$	$Train_{BC} + HConf_{A1}$
Test data	$Data_{A2}$	$Data_{A2}$
Accuracy (%)	79	86.21

B and C are the known locations and A is the new location

**Table 13** Recognition results on the known locations before and after adaptation

	Before adaptation	After adaptation
Train data	$Train_{AC}$	$Train_{AC} + HConf_{B1}$
Test data	$Test_{AC}$	$Test_{AC}$
Accuracy (%)	80.53	79.08

A and C are the known locations and B is the new location

**Table 14** Recognition results on the new location before and after adaptation

	Before adaptation	After adaptation
Train data	$Train_{AC}$	$Train_{AC} + HConf_{B1}$
Test data	$Data_{B2}$	$Data_{B2}$
Accuracy (%)	59.08	71.82

A and C are the known locations and B is the new location

When A and C are assumed as known locations and B is as the new location, experiment results are shown in Tables 13, 14. After adaptation, the performance is improved more than 12%.

## 6 Conclusions and future work

In this paper, we research on the physical activity recognition on an accelerometer-embedded mobile device. In contrast to the previous work of assuming that the phone is placed in a fixed position, this paper intends to recognize the physical activities in the daily life when the device locations and orientations are varying.

We propose a fast and robust activity recognition model to deal with the problem of varying device locations and orientations. ELM, a fast learning classification method, is used to retrain the recognition model online and adapt the model to new locations. Experimental results demonstrate that model adaptation improves the recognition accuracy obviously without any knowledge of new locations.

In the future, we will employ 30 persons to collect data of more daily activities and consider about more locations where the device is placed. To build a robust model, the participants should have different physical conditions such as gender, age, height, et al. We will collect the realistic data as the participants go about their normal activities. With the number of activities increased, the model will be more and more complex. Then we will test the performance of linear PCA and nonlinear PCA, such as kernel PCA. If nonlinear PCA outperforms linear PCA, we will select it as our dimensionality reduction method.

**Acknowledgments** This work is supported in part by the Natural Science Foundation of China (61070110, 90820303), and Beijing



Natural Science Foundation (4112056), and the National Basic Research Program of China (973 Program, 2011CB302803).

## References

- Chacko BP, Krishnan VRV, Raju G, Anto PB (2011) Handwritten character recognition using wavelet energy and extreme learning machine. *Int J Mach Learn Cybern*. doi:[10.1007/s13042-011-0049-5](https://doi.org/10.1007/s13042-011-0049-5)
- Feng G, Huang GB, Lin Q, Gay R (2009) Error minimized extreme learning machine with growth of hidden nodes and incremental learning. *IEEE Trans Neural Netw* 20(8):1352–1357
- Figio D, Diniz PC, Ferreira DR, Cardoso JMP (2010) Preprocessing techniques for context recognition from accelerometer data. *Pers Ubiquitous Comput* 14(7):645–662
- Huang GB, Chen L (2007) Convex incremental extreme learning machine. *Neurocomputing* 70:3056–3062
- Huang GB, Chen L (2008) Enhanced random search based incremental extreme learning machine. *Neurocomputing* 71:3460–3468
- Huang GB, Ding X, Zhou H (2010) Optimization method based extreme learning machine for classification. *Neurocomputing* 74:155–163
- Huang GB, Wang DH, Lan Y (2011) Extreme learning machines: a survey. *Int J Mach Learn Cybern* 2(2):107–122
- Huang GB, Zhu QY, Siew CK (2004a) Extreme learning machine. In: Technical report ICIS/03/2004
- Huang GB, Zhu QY, Siew CK (2004b) Extreme learning machine: a new learning scheme of feedforward neural networks. In: *IJCNN*, vol 2, pp 985–990
- Huang GB, Zhu QY, Siew CK (2006) Extreme learning machine: theory and applications. *Neurocomputing* 70:489–501
- Han J, Kamber M (2000) Data mining: concepts and techniques. Morgan Kaufmann, Menlo Park
- Jun W, Shitong W, Chung F (2011) Positive and negative fuzzy rule system, extreme learning machine and image classification. *Int J Mach Learn Cybern* 2:261–271
- Kunze K, Lukowicz P (2008) Dealing with sensor displacement in motion-based onbody activity recognition systems. In: *UbiComp*, USA, pp 20–29
- Kunze K, Lukowicz P, Junker H, Troster G (2005) Where am I: recognizing on-body positions of wearable sensors. In: Strang T, Linnhoff-Popien C (eds) Location- and context-awareness. Lecture notes in computer science, vol 3479. Springer, Berlin, pp 257–268
- Lai CF, Huang YM, Park JH, Chao HC (2010) Adaptive body posture analysis for elderly-falling detection with multisensors. *IEEE Intell Syst* 25(2):20–30
- Mizell D (2003) Using gravity to estimate accelerometer orientation. In: *ISWC 2003*, USA
- Reddy S, Burke J, Estrin D, Hansen M, Srivastava M (2008) Determining transportation mode on mobile phones. In: *IEEE international symposium on wearable computers (ISWC 2008)*, Pittsburgh, Pennsylvania, pp 25–28
- Roggen D, Magnenat S, Waibel M, Troster G (2011) Wearable computing: designing and sharing activity-recognition systems across platforms. *IEEE Robot Autom Mag* 18(2):83–95
- Zhang S, McCullagh P, Nugent C, Zheng H, Baumgarten M (2011) Optimal model selection for posture recognition in home-based healthcare. *Int J Mach Learn Cybern* 2:1–14
- Sun L, Zhang D, Li B, Guo B, Li S (2010) Activity recognition on an accelerometer embedded mobile phone with varying positions and orientations. In: *UIC 2010*, pp 548–562
- Wang X-Z, Dong C-R (2009) Improving generalization of fuzzy if-then rules by maximizing fuzzy entropy. *IEEE Trans Fuzzy Syst* 17:556–567
- Wang S, Chen C, Ma J (2010) Accelerometer based transportation mode recognition on mobile phones. In: *APWCS*, pp 44–46
- Wang XZ, Chen AX, Feng HM (2011a) Upper integral network with extreme learning mechanism. *Neurocomputing* 74:2520–2525
- Wang X-Z, Dong L-C, Yan J-H (2011b) Maximum ambiguity based sample selection in fuzzy decision tree induction. *IEEE Trans Knowl Data Eng*. doi:[10.1109/TKDE.2011.67](https://doi.org/10.1109/TKDE.2011.67)
- Xiao J-Z, Wang H-R, Yang X-C, Gao Z (2011) Multiple faults diagnosis in motion system based on svm. *Int J Mach Learn Cybern*. doi:[10.1007/s13042-011-0035-y](https://doi.org/10.1007/s13042-011-0035-y)
- Wang XL (2004) High accuracy distributed target detection and classification in sensor networks based on mobile agent framework. Phd thesis, University of Tennessee
- 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*, Beijing, China, pp 1–10
- Zhao Z, Chen Y, Liu J, Shen Z, Liu M (2011) Cross-people mobile-phone based activity recognition. In: *Proceedings of the twenty-second international joint conference on artificial intelligence (IJCAI)*, pp 2545–2550