

# Integrating the enriched feature with machine learning algorithms for human movement and fall detection

Chenghua Li · Man Lin · Laurence T. Yang ·  
Chen Ding

Published online: 3 December 2013  
© Springer Science+Business Media New York 2013

**Abstract** Movement detection is gaining more and more attention among various pattern recognition problems. Recognizing human movement activity types is extremely useful for fall detection for elderly people. Wireless sensor network technology enables human motion data from wearable wireless sensor devices be transmitted for remote processing. This paper studies methods to process the human motion data received from wearable wireless sensor devices for detecting different types of human movement activities such as sitting, standing, lying, fall, running, and walking. Machine learning methods K Nearest Neighbor algorithm (KNN) and the Back Propagation Neural Network (BPNN) algorithm are used to classify the activities from the data acquired from sensors based on sample data. As there are a large amount of real-time raw data received from sensors and there are noises associated with these data, feature construction and reduction are used to preprocess these raw sensor data obtained from accelerometers embedded in wireless sensing motes for learning and processing. The singular value decomposition (SVD) technique is used for constructing the enriched features. The enriched features are then integrated with machine learning algorithms for movement detection. The testing data are collected from five adults. Experimental results show that our methods can achieve promising performance on human movement recognition and fall detection.

**Keywords** Human movement detection · Wireless sensor network · Machine learning

---

C. Li  
CUIUC, ChangZhou University, ChangZhou, P.R. China

C. Li · M. Lin (✉) · L.T. Yang · C. Ding  
Department of Mathematics, Statistics and Computer Science, St. Francis Xavier University,  
Antigonish, Canada  
e-mail: [mli@stfx.ca](mailto:mli@stfx.ca)

## 1 Introduction

Embedded systems are designed to do some specific tasks. Examples of embedded systems are accelerometers, GPS, thermostats, digital cameras, medical equipments, etc. There has been extensive research on timing analysis [32, 33], energy efficiency [28, 29, 37, 39, 42, 45], reliability issues [38, 44], security issues [34–36], and software synthesis problems [43] on embedded systems. One current trend of an embedded application is to connect embedded systems to a network or internet to allow the embedded systems be controlled remotely, or to allow the data collected from embedded systems be analyzed remotely. Various types of embedded devices such as RFID, GPS, accelerometers are now connected to the Internet (called Internet of Things or Cyber-Physical Systems) to achieve intelligent identification, tracking, and monitoring to help improve the quality of life of people.

According to the health report in Canada [25], approximately 30 % of people over 65 years of age living in the community fall each year. Falls are the major cause of injury related to hospitalization among the elderly population. Injuries from falls include broken or fractured bones, and soft tissue damage, etc. Falls are also the major cause of injury deaths. Fall detection is thus an important application. An accelerometer attached to a human body can gather real-time body motion data. The development of wireless sensor network technologies makes it possible to transmit the motion data to a remote analysis center without affecting human motion (no wires needed). Movement detection and activity recognition have been applied for various applications such as interactive entertainment, and fall detection for elderly people [6, 7, 27]. This paper studies methods to recognize human movement activities using advanced intelligent methods or machine learning algorithms.

Movement detection is an emerging technique among various pattern recognition problems [3, 12, 30, 31]. There are a lot of previous works proposed for human movement detection [1–5, 40]. Some of them used one waist mounted triaxial accelerometer [1], and some of them used several biaxial accelerometers mounted on different parts of the body [2]. Chen [3] built a real-time motion monitoring system based on two waist mounted Mica2 motes. Dai et al. [27] implemented a fall detection prototype system based on mobile phone platform. Zhang et al. [46] designed an unobtrusive wireless sensor network for nighttime falls detection.

Machine learning algorithms or simple threshold comparison methods are used to classify the activities from the data acquired from sensors. The activity classification methods in [3, 27, 41] are threshold-based methods. Machine learning techniques are widely for activity classification. Karantonis et al. [1] constructed a real time decision tree based human movement classification system with the data acquired from one single, waist-mounted triaxial accelerometer unit. Chen et al. [13] employed a fuzzy basis function classifier for human activity detection. Yang [8] applied the neural network classifier for activity recognition. Huang et al. [5] first applied an ant colony clustering algorithm to estimate and classify the body posture and then used hidden Markov models for human motion recognition. Ward et al. [9] also used hidden Markov models to recognize continuous human activities such as sawing, hammering, drilling, etc. Ravi et al. [10] evaluated the performances of activity recognition on four different machine learning algorithms: support vector machine, decision tree,

K nearest neighbor and naive Bayes. Bao and Intille [2] compared several machine learning classifiers for activity recognition with the data acquired from five biaxial accelerometers worn on different parts of the body.

As there are large amount of real-time raw data received from accelerometers and there are noises associated with these data, the machine learning based classifier algorithms usually do not work directly on the raw motion data. Instead, feature extraction is performed first. Feature representation is an important procedure in pattern recognition. It includes feature extraction, feature selection, and feature reduction. Previous studies on activity detection are mostly based on three kinds of features:

- (1) the triaxial acceleration data [1],
- (2) the fusion of acceleration data [2, 8, 10–13] such as mean, correlation between axes, energy, and standard deviation,
- (3) features selected by selection and reduction methods such as feature subset selection based on common principal component analysis [8, 11], and linear discriminate analysis [11, 13].

How to extract and select good features to better represent a pattern is critical problem for pattern recognition and will directly affect the recognition rate. In this paper, we use singular value decomposition (SVD) [15, 16] to construct the enriched feature for human movement detecting.

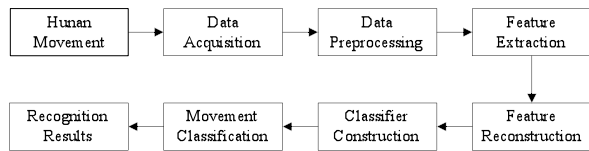
Singular value decomposition is a well-developed method for extracting the dominant features of large data sets and for reducing the dimensionality of the data by projecting the original high dimensional space onto low semantic dimensional space. It is a mathematical concept used in most of the Latent Semantic Indexing (LSI) methods. LSI was originally proposed as an information retrieval method [17] to construct the semantic relationship. It can also be used for pattern classification [18]. Compared to other feature reduction methods, such as principal component analysis and linear discriminate analysis, SVD takes advantage of the consideration of relationships between features to establish underlying semantic means. It generates an enriched feature space with a small number of features to represent the whole feature space, which can not only reduce the dimensionality, but also improve the performance.

The motivation of our work is to detect and monitor the individual activity and provide valuable information of people's health status and daily life. It is very useful for remote health care, especially for elderly people.

The overview of the system is shown in Fig. 1. In the data acquisition step, the accelerometers in the two ADXL202 mobile motes [3] read the signal stream resulted from the human movement, and the motes send the data to the processing computer. The raw triaxial acceleration data will go through the preprocessing stage and form sliding windows with 20 % overlap. Feature extraction method is performed on the signal sequence windows to generate the fusion feature such as mean, correlation of axes. Feature reconstruction is then used to enrich the feature by adapting the singular value decomposition technique. The enriched features will be used by the movement classification algorithms such as K-Nearest Neighbor (KNN) and neural network for human movement recognition.

The rest of the paper is organized as follows. Section 2 describes two different machine learning algorithms, which are used in our system for movement activity

**Fig. 1** The overview of the system



recognition. Section 3 describes the data acquisition and feature reconstruction process. Section 4 describes the experiments design, results comparison, and analysis. The conclusions are given in Sect. 5.

## 2 Machine learning algorithms

A major focus of machine learning research is to automatically learn to recognize complex patterns and make intelligent decisions based on experiments or samples. Based on whether the desired outcome for the training input data is given to the machine learning algorithm, there are mainly four types of machine learning algorithms, that is, supervised learning, unsupervised learning, semisupervised learning, and reinforcement learning. Based on different learning paradigms, there are many different kinds of machine learning algorithms such as decision tree learning, Bayesian networks, support vector machines, k-nearest neighbor, and artificial neural networks. The applications for machine learning include computer vision [19, 20], natural language processing [18, 22, 26], search engines [21], medical diagnosis [23], and pattern recognition [24]. In this paper, we use two commonly used machine learning algorithms: K Nearest Neighbor algorithm (KNN) and Back Propagation Neural Network (BPNN) algorithm, for our activity recognition problem.

### 2.1 K nearest neighbor algorithm

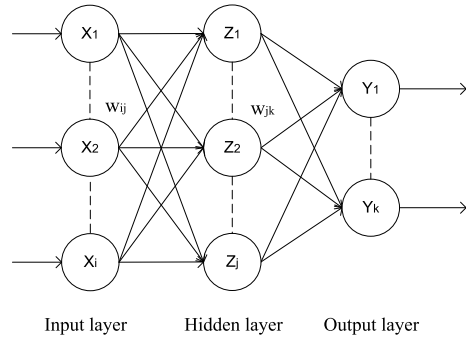
The K Nearest Neighbor algorithm (KNN) is a method for classifying objects based on the closest training examples in the feature space. It is one of the most fundamental and simple algorithms for classification problems. The KNN for classification can be briefly described as follows.

Given a sample  $S$  to be classified, KNN finds the  $k$  samples (or neighbors) in the training data that are closest to  $S$ . If most of these neighbors belong to class  $A$ , then  $S$  is assigned to class  $A$ . The KNN algorithm can be viewed as a voting system in which neighbors vote for classes.

The major steps of the KNN algorithm can be summarized as follows:

- (1) Construct a training matrix  $D$ , which includes  $x$  samples and  $f$  features, with each row corresponding to a training sample, and each column representing a feature.
- (2) For each training sample  $x$  that belongs to  $D$ , compute the corresponding feature weight of feature vector  $d_x$ .
- (3) For each test sample  $y$ , compute the corresponding feature weight of feature vector  $d_y$ .

**Fig. 2** Three layer neural networks



- (4) Calculate the cosine similarity (or distance) between the test sample and each training sample  $S_x = \cos \text{Sim}(d_y, d_x)$ .
- (5) Sort samples in  $D$  by decreasing order of the value of  $S_x$ .
- (6) Let  $N$  be the first  $k$  samples in  $D$ .
- (7) Return the majority class of samples in  $N$ .

## 2.2 Back propagation neural network algorithms

The Back Propagation Neural Network (BPNN) algorithm is the most popular of the neural network applications. The topology structure of the standard BPNN algorithm is shown in Fig. 2.

The BPNN for classification can be briefly described as: Given a sample  $S$  to be classified, the network will calculate the corresponding output for the certain input sample according to the neural network algorithm. The network will compare the actual output with the desired output for each class. If the actual output for input pattern  $S$  is the closest to the desired output for class  $A$ , then  $S$  should be assigned to class  $A$ .

Different from the KNN method, there is a training process in neural network. The neural network training process is to adjust the weights of the networks. The major steps of the BPNN training algorithm can be summarized as follows:

- (1) For each training sample, calculate the actual output value in hidden layer and output layer.
- (2) Compute the system error using mean absolute error (MAE) function.
- (3) Back-propagate the difference between the actual output value and the desired output value from output layer to hidden layer.
- (4) Back-propagate the difference between the actual output value and the desired output value from hidden layer to input layer.
- (5) Compute the change of the weight between the hidden layer and the output layer.
- (6) Compute the change of the weight between the input layer and the hidden layer.
- (7) Compute the change of bias in the hidden layer and in the input layer.
- (8) Stop training when no changes have been made on the network.

The BPNN classification steps are as follows:

- (1) For each test sample, calculate the actual output value by multiplying the input features with the final updated weights between two layers.

- (2) Compare the actual output value of the test samples with desired output value for each class.
- (3) Assign the test sample to the class in which the corresponding actual output value of the test sample is closest to the desired output value of the class.

Next, we will describe the data used by the two machine learning methods for movement activity classification.

### 3 Data acquisition and feature reconstruction

#### 3.1 Data acquisition

In our system, the acceleration data are acquired from the accelerometers embedded in the Mica2 motes, which are attached to the central back of the human waist. The accelerometer data includes gravitational acceleration (GA) component and dynamic acceleration (DA) component. In real world, for every object, GA is caused by the force of gravity. DA is due to the force applied to the object. It could be internal force or external force. The internal force is caused by the object itself, such as human body movement or exercise. So, the DA component is used to distinguish activity due to the body movement. The signals in each axis can be represented as [3]:

$$x = x_{DA} + x_{GA} \quad (1)$$

$$y = y_{DA} + y_{GA} \quad (2)$$

$$z = z_{DA} + z_{GA} \quad (3)$$

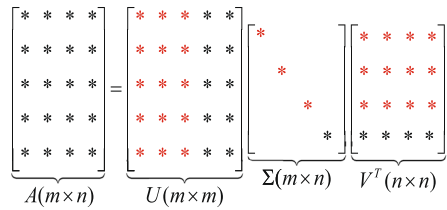
Because the movement signals form continuous sequence, we divide the sequence data into windows with the interval of 1 s [12]. One window includes 10 triaxial data points. For every 0.1 s, we obtain a group of triaxial  $x$ ,  $y$ ,  $z$  data. The triaxial accelerometer is captured as the original data and will be used to extract features.

#### 3.2 Feature extraction

Using only the original triaxial accelerometer data cannot easily discriminate various types of movement well. Some previous works explored fusion features from the three axes of the accelerometer [13, 14]. In this paper, we also extract the features in time domain and frequency domain from each window of the triaxial acceleration data using the same method as introduced in [13]. The time domain features include: (1) mean value of the acceleration data ( $x$ ,  $y$ ,  $z$ ) over a window, (2) correlation between axes ( $x$ ,  $y$ ), ( $y$ ,  $z$ ), ( $z$ ,  $x$ ), (3) interquartile range, (4) mean absolute deviation, (5) root mean square, (6) standard deviation, and (7) variance. The frequency domain feature (8) energy, is calculated as the sum of the magnitudes of squared discrete fast Fourier transform (FFT) components of the signal in a window.

So far, we have extracted 8 features for each axes, and 24 features for  $x$ ,  $y$ ,  $z$  axes. We also introduce another two features [1] to be part of the feature space:

**Fig. 3** The diagram of singular value decomposition



(1) Signal magnitude area (SMA) is defined as

$$SMA = 1/t \left( \int_0^t |x(t)| dt + \int_0^t |y(t)| dt + \int_0^t |z(t)| dt \right) \tag{4}$$

where  $x_t, y_t, z_t$  are components of the  $x, y, z$  axis for the samples. SMA can be used to determine whether human’s movement is dynamic or static during a time period.

(2) Signal magnitude vector (SVM) is defined as

$$SVM = \sqrt{x_i^2 + y_i^2 + z_i^2} \tag{5}$$

where  $x_i, y_i, z_i$  are the  $i$ th sample of the  $x, y, z$  axis signals. SVM can be used to determine whether human falls or not during a time period.

We get totally 26 features from the original acceleration data after feature extraction. The feature vector for sample  $i$  can be represented as

$$F_i = \{M_x, M_y, M_z, Cor_x, Cor_y, Cor_z \dots SMA, SVM\} \tag{6}$$

### 3.3 Feature reconstruction

In linear algebra, Singular value decomposition (SVD) is a factorization of a matrix. It has many applications in signal processing and information retrieval. Singular value decomposition (SVD) is a semantic indexing method attempting to capture the underlying relationship among the features, which is helpful for discriminating the classes. It is also a good dimension reduction technique as shown below.

The training data with  $m$  samples and  $n$  features for each sample can represent as a matrix  $A(m \times n)$ . Then the matrix decomposed into three different matrixes as Fig. 3.

The feature reconstruction using SVD technique can be described as follows:

- (1) Construct a training matrix  $A(m \times n)$  which includes  $x$  samples and  $y$  features, with each row corresponding to a training sample, and each column representing a feature.
- (2) Apply the SVD technique on matrix  $A$  as shown in Fig. 3.
- (3) Reduce the dimensionality by taking the first  $K$  columns and rows as shown in Fig. 3 marked by red point.
- (4) Feature reconstruction according to the formula  $F_{\text{new}} = F_{\text{old}} \times U_k$ , where  $F_{\text{new}}$  is the feature space after feature reconstruction,  $F_{\text{old}}$  is the original feature space, and  $U_k$  is the first  $k$  columns of  $U$  matrix.

**Table 1** 1: Movement classification result when distinguishing the active and inactive manner

Movement type	No. of correct/total No.		Recognition rate	
	KNN	BPNN	KNN	BPNN
Inactive sitting	49/60	54/60	0.816	0.900
Active sitting	53/60	55/60	0.883	0.916
Inactive standing	53/60	57/60	0.883	0.950
Active standing	58/60	59/60	0.966	0.983
Inactive lying	52/60	55/60	0.866	0.916
Active lying	50/60	54/60	0.833	0.900
Falling	47/60	52/60	0.783	0.866
Walking	52/60	57/60	0.866	0.950
Running	56/60	58/60	0.933	0.966
<i>Average</i>	470/540	501/540	0.870	0.927

## 4 Experiments design and results

To evaluate the performance of the algorithms on the movement and fall detection, a set of experiments have been designed to test the recognition rate. Mica2 Motes [26] are used in our system. Two motes are used for building the triaxial mobile node and are attached to the tester's back of waist. And the other motes are placed in the room. Five adults took part in the test. We test 3 rounds with 20 samples for each round. All the tests are evaluated in the indoor environment based on the activities of standing, sitting, lying, walking, running, and fall.

Two sets of experiments have been designed to test our classifier. Set 1 distinguishes the active and inactive manner. Inactive manner stands for the static activity and active manner stands for the dynamic activity. For example, doing some tasks while sitting on the chair is classified as active sitting, and sitting on the chair in still is classified as inactive sitting. Experiment set 2 does not distinguish the active and inactive manner. The sitting activity in this set of experiments includes both active and inactive sitting. The experimental results are shown in the following tables.

Table 1 gives the recognition results based on nine movement types which distinguish the active and inactive manner. The results show that the neural network learning algorithm works better than the KNN algorithm.

Table 2 gives the recognition results based on six types of movements without distinguishing the active and inactive manner. The results show that the recognition rate on the six movement types in experiment set 2 is much higher than that of the nine movement types in experiment set 1. The results in experiment set 2 also show that the neural network learning algorithm works better than the KNN algorithm.

The first set of experiments needs to distinguish the active and inactive manner, which makes the patterns more complex, thus causes more misclassification. This explains that the recognition rate in set 1 is not as accurate as that in set 2. Therefore, the discrimination ability of the KNN and BPNN algorithm to classify movement types in set 1 is less than that in set 2.

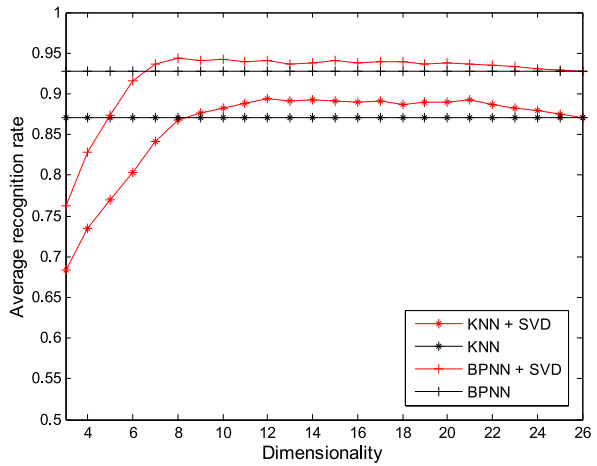
A proper number of dimension can obtain a good result since SVD captures the most important information as well as establish the semantic relationship between



**Table 2** 2: Movement classification result without distinguishing the active and inactive manner

Movement type	No. of correct/total No.		Recognition rate	
	KNN	BPNN	KNN	BPNN
Sitting	56/60	59/60	0.933	0.983
Standing	59/60	60/60	0.966	1
Lying	55/60	58/60	0.916	0.966
Falling	54/60	57/60	0.900	0.95
Walking	57/60	60/60	0.950	1
Running	58/60	60/60	0.966	1
<i>Average</i>	339/360	354/360	0.941	0.983

**Fig. 4** Apply SVD on experiment set 1

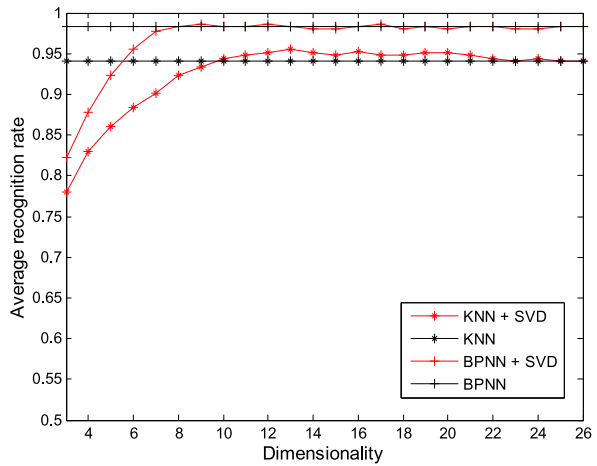


features. Figure 4 shows the application of SVD on the first set of experiments, where the dimensions vary from 3 to 25. The original feature space (when SVD is not applied) has the dimensionality of 26. We test how the performance varies when the dimensionality is decreased. When the reduction of the dimensionality is small, we find that the performance (the recognition rate) increases when the dimension decreases. This is because some noisy information is removed when reducing the dimensions. However, when dimension becomes too small, the recognition rate becomes lower. This is because a small feature space resulted from SVD may cause important information loss, and thus lead to bad performance.

From the Fig. 4, we found that the best performance of KNN is obtained when the dimensions are 12, reduced more than half. The best performance of BPNN is obtained when the dimensions are 8, reduced more than 2/3.

Figure 5, shows the application of SVD on the second set of experiments, where the dimensions also vary from 3 to 25. In Fig. 5, The best performance of 0.955 for KNN is obtained when the dimensions are 13, reduced around half. The best performance of 0.986 for BPNN is obtained when the dimensions are 9, reduced almost 2/3. The average recognition rate for experiment 1 and 2 with different features and different algorithms are shown in Table 3. The introduction of SVD improves the av-

**Fig. 5** Apply SVD on experiment set 2



**Table 3** Performance comparison

	Average recognition rate			
	KNN	KNN + SVD	BPNN	BPNN + SVD
Experiment 1	0.870	0.894	0.927	0.944
Experiment 2	0.941	0.955	0.983	0.986

verage recognition rate in the first set of experiment from 0.870 to 0.894 for KNN and 0.927 to 0.944 for BPNN. The introduction of SVD improves the average recognition rate of the second experiment from 0.941 to 0.955 for KNN and 0.983 to 0.986 for BPNN.

### 5 Conclusions

Recognizing human movement activity types is especially useful for elderly people and has gained a lot of attention. In this paper, we present a framework for human activity recognition using machine learning algorithms. Singular value decomposition technique has been adapted to generate a set of enriched features based on the acceleration data collected from wireless sensor network. Experiments conducted on different setting achieve different performances. The first set of experiments distinguishes the active and inactive manner; the activity patterns are thus complex and have a higher rate of misclassification. The second set of experiments does not distinguish the active and inactive manners, and the misclassification rate is lower. The introduction of SVD in the experiments has reduced the dimensions of features, which not only leads to lower computation cost, but also improves the recognition rate.

**Acknowledgement** This work was supported by NSERC (Natural Sciences and Engineering Research Council, Canada).

## References

1. Karantonis DM, Narayanan MR, Mathie MJ, Lovell NH, Celler BG (2006) Implementation of a real-time human movement classifier using a tri-axial accelerometer for ambulatory monitoring. *IEEE Trans Inf Technol Biomed* 10(1):156
2. Bao L, Intille SS (2004) Activity recognition from user-annotated acceleration data. In: Proc second international conference on pervasive computing, pp 1–17
3. Ding C (2009) A real time motion monitoring system. Master thesis, St. Francis Xavier University, Antigonish, Canada
4. Scanaill NC, Carew S, Barralon P, Noury N, Lyon D, Lyons GM (2006) A review of approaches to mobility telemonitoring of the elderly in their living environment. *Ann Biomed Eng* 34(4):547–563
5. Huang WY, Zhang J, Liu ZJ (2007) Activity recognition based on hidden Markov models. In: Knowledge science, engineering and management. LNCS, vol 4798, pp 532–537
6. Song KT, Wang YQ (2005) Remote activity monitoring of the elderly using a two-axis accelerometer. In: Proc CACS automatic control conference (EMBECE 2005)
7. Choi ES, Bang WC, Kim DY, Kim SR Beatbox music phone: Gesture based interactive mobile phone using a tri-axis accelerometer Proc IEEE international conference industry technology, pp 97–102
8. Wang JY, Yang JS, Chen YP (2008) Using acceleration measurements for activity recognition: an effective learning algorithm for constructing neural classifiers. *Pattern Recognit Lett* 29:2213–2220
9. Ward JA, Lukowicz P, Troster G, Starner TE (2006) Activity recognition of assembly task using body-worn microphones and accelerometers. *IEEE Trans Pattern Anal Mach Intell* 28(10):1553–1567
10. Ravi N, Dandekar N, Mysore P, Littman ML (2005) Activity recognition from accelerometer data. In: Proc seventeenth innovative applications of artificial intelligence conference, pp 1541–1546
11. Yang JY, Chen YP, Lee GY, Liou SN, Wang JS (2007) Activity recognition using one triaxial accelerometer: a neuro-fuzzy classifier with feature reduction. In: ICEC 2007. LNCS, vol 4740, pp 395–400
12. Mathie MJ (2003) Monitoring and interpreting human movement patterns using a tri-axial accelerometer. Ph.D thesis, Univ. New South Wales, Sydney, Australia
13. Chen YP, Yang JY, Liou SN, Lee GY, Wang JS (2008) Online classifier construction algorithm for human activity detection using a tri-axial accelerometer. *Appl Math Comput* 205:849–860
14. Wang S, Yang J, Chen N, Chen X, Zhang Q (2005) Human activity recognition with user-free accelerometers in the sensor networks. In: Proc IEEE int conf neural networks and brain, vol 2, pp 1212–1217
15. Noorinaeini A, Lehto MR (2006) Hybrid singular value decomposition; a model of human text classification. *Int J Hum Factors Model Simul* 1(1):95–118
16. Yany Y (1995) Noise reduction in a statistical approach to text categorization. In: Proc 18th ACM international conference on research and development in information retrieval, New York, pp 256–263
17. Deerwester SC, Dumais ST, Landauer TK, Furnas GW, Harshman RA (1990) Indexing by latent semantic analysis. *J Am Soc Inf Sci* 41(6):391–407
18. Li CH, Park SC (2009) An efficient document classification model using an improved back propagation neural network and singular value decomposition. *Expert Syst Appl* 36:3208–3215
19. Park SB, Lee JW, Kim SK (2004) Content-based image classification using a neural network. *Pattern Recognit Lett* 25(3):287–300
20. Chapelle O, Haffner P, Vapnik VN (1999) Support vector machines for histogram-based image classification. *IEEE Trans Neural Netw* 10(5):1055–1064
21. Neuman L, Kozłowski J, Zgrzywa A (2004) Information retrieval using Bayesian networks. In: Proc ICCS 2004. LNCS, vol 3038, pp 521–528
22. Lawrence S, Giles CL, Fong S (2000) Natural language grammatical inference with recurrent neural networks. *IEEE Trans Knowl Data Eng* 12(1):126–140
23. Veropoulos K (2001) Machine learning approaches to medical decision making. PhD thesis, University of Bristol, March 2001
24. Graves A, Schmidhuber J (2008) Offline handwriting recognition with multidimensional recurrent neural networks. In: Proc NIPS 2008
25. Health report, Canada (2009) [http://www.health.gov.on.ca/english/providers/program/ohnac/pdf/progress/2009/falls\\_en.pdf](http://www.health.gov.on.ca/english/providers/program/ohnac/pdf/progress/2009/falls_en.pdf)
26. Mica2 Datasheet, [http://bullseye.xbow.com:81/Products/Product\\_pdf\\_files/Wireless\\_pdf/MICA2\\_Datasheet.pdf](http://bullseye.xbow.com:81/Products/Product_pdf_files/Wireless_pdf/MICA2_Datasheet.pdf)

27. Dai J, Bai X, Yang Z, Shen Z, Xuan D (2010) PerFallID: a pervasive fall detection system using mobile phones. In: Personal and ubiquitous computing, April
28. Zheng NG, Wu ZH, Lin M, Yang LT (2010) Enhancing battery efficiency for pervasive health monitoring systems based on electronic-textiles. *IEEE Trans Inf Technol Biomed* 14(2):350–359
29. Yang L, Lin M, Yang LT (2009) Integrating preemption threshold to fixed priority DVS scheduling algorithms. In: Proceedings of the 15th IEEE international conference on embedded and real-time computing systems and applications (RTCSA'09), Beijing, China, 24–26 August, 2009, pp 165–171
30. Li CH, Yang LT, Lin M (2013) Parallel training of an improved neural network for text categorization. *Int J Parallel Program*. doi:[10.1007/s10766-013-0245-x](https://doi.org/10.1007/s10766-013-0245-x)
31. Lai CF, Lin M, Wen YG, Ma YW, Chen JL (2012) Applied lightweight parallel multi-appliance recognition on smart meter. In: The 15th IEEE international conference on computational science and engineering (CSE 2012), Paphos, Cyprus, 5–7 December, 2012, pp 361–366
32. Lin M (2000) Timing analysis of PL programs. *J Control Eng Practice* 8:697–703
33. Lin M, Malec J (1998) Timing analysis of reactive rule-based programs. *J Control Eng Practice* 6:403–408
34. Xie T, Qin X, Sung AH, Lin M, Yang LT (2006) Real-time scheduling with quality of security constraints. *Int High Perform Comput Networking* 4(3):188–197
35. Lin M, Xu L, Yang LT, Qin X, Zheng NG, Wu ZH, Qiu MK (2009) Static security optimization for real time systems. *IEEE Trans Ind Inform* 5(1):22–37
36. Yang LT, Xu L, Lin M (2005) Integer factorization by a parallel GNFS algorithm for public key cryptosystems. In: Embedded software and systems. LNCS, vol 3820, pp 683–695
37. Lin M, Ding C (2007) Parallel genetic algorithms for DVS scheduling of distributed embedded systems. In: High performance computing and communications. LNCS, vol 4782, pp 180–191
38. Pan YW, Lin M, Yang LT (2011) Reducing total energy for reliability-aware DVS algorithms. In: Ubiquitous intelligence and computing—8th international conference (UIC 2011), Banff, Canada, 2–4 September, 2011, pp 576–589
39. Yang L, Lin M, Yang LT (2012) Multi-core fixed priority DVS scheduling. In: The 12th international conference on algorithms and architectures for parallel processing (ICA3PP 2012), Fukuka, Japan, 4–7 September, 2012, pp 517–530
40. Zhao G, Huang B, Liu G, Guo Y, Mei Z, Wang L (2011) A low-cost body sensor networks for elderly activities recognition and fall prediction. In: Proceedings of the international bionic engineering conference 2011, Boston, USA, 18–20 September, 2011
41. Bagalà F, Becker C, Cappello A, Chiari L, Aminian K et al (2012) Evaluation of accelerometer-based fall detection algorithms on real-world falls. *PLoS ONE* 7(5):e37062. doi:[10.1371/journal.pone.0037062](https://doi.org/10.1371/journal.pone.0037062)
42. Benini L, De Micheli G (1999) Energy-efficient design of battery-powered embedded systems. In: Low power electronics and design: Proceedings of 1999 international symposium on date of conference, Aug 1999, pp 212–217
43. Lin M (1999) Synthesis of control software in a layered architecture from hybrid automata. In: Hybrid systems: computation and control. LNCS, vol 1569. Springer, Berlin, pp 152–164
44. Lin M, Pan YW, Yang LT, Guo MY, Zheng NG (2013) Scheduling co-design for reliability and energy in cyber-physical systems. *IEEE Trans Emerg Top Comput*. doi:[10.1109/TETC.2013.2274042](https://doi.org/10.1109/TETC.2013.2274042)
45. Qiu M, Sha E, Liu M, Lin M, Hua S, Yang LT (2008) Energy minimization with loop fusion and multi-functional-unit scheduling for multidimensional DSP. *J Parallel Distrib Comput* 68(4):443–455
46. Zhang Z, Kapoor U, Narayanan M, Lovell NH, Redmond SJ (2011) Design of an unobtrusive wireless sensor network for nighttime falls detection. *IEEE Eng Med Biol Soc*