Original Article

Healthc Inform Res. 2017 July;23(3):147-158.
https://doi.org/10.4258/hir.2017.23.3.147
pISSN 2093-3681 • eISSN 2093-369X

HIR
Healthcare Informatics Research

Fall Detection System for the Elderly Based on the Classification of Shimmer Sensor Prototype Data

Moiz Ahmed, BS[1], Nadeem Mehmood, PhD[1], Adnan Nadeem, PhD[2], Amir Mehmood, MS[3], Kashif Rizwan, MCS[1,3]

[1]Department of Computer Science, University of Karachi, Karachi, Pakistan; [2]Faculty of Computer and Information System, Islamic University in Madinah, Madinah, Saudi Arabia; [3]Department of Computer Science, Federal Urdu University of Arts Science and Technology, Karachi, Pakistan

Objectives: Falling in the elderly is considered a major cause of death. In recent years, ambient and wireless sensor platforms have been extensively used in developed countries for the detection of falls in the elderly. However, we believe extra efforts are required to address this issue in developing countries, such as Pakistan, where most deaths due to falls are not even reported. Considering this, in this paper, we propose a fall detection system prototype that s based on the classification on real time shimmer sensor data. **Methods:** We first developed a data set, 'SMotion' of certain postures that could lead to falls in the elderly by using a body area network of Shimmer sensors and categorized the items in this data set into age and weight groups. We developed a feature selection and classification system using three classifiers, namely, support vector machine (SVM), K-nearest neighbor (KNN), and neural network (NN). Finally, a prototype was fabricated to generate alerts to caregivers, health experts, or emergency services in case of fall. **Results:** To evaluate the proposed system, SVM, KNN, and NN were used. The results of this study identified KNN as the most accurate classifier with maximum accuracy of 96% for age groups and 93% for weight groups. **Conclusions:** In this paper, a classification-based fall detection system is proposed. For this purpose, the SMotion data set was developed and categorized into two groups (age and weight groups). The proposed fall detection system for the elderly is implemented through a body area sensor network using third-generation sensors. The evaluation results demonstrate the reasonable performance of the proposed fall detection prototype system in the tested scenarios.

Keywords: Aged Humans, Computer Communication Network, Accidental Fall Detection, Information Systems, Shimmer, Wireless Technology, Machine Learning

Submitted: March 19, 2017
Revised: May 30, 2017
Accepted: June 28, 2017

Corresponding Author
Adnan Nadeem, PhD
Faculty of Computer Science and Information System, Islamic University in Madinah, Madinah, Saudi Arabia. Tel: +966-565425963, E-mail: adnan.nadeem@iu.edu.sa

I. Introduction

A fall, according to [1], is defined as "an event which results in a person coming to repose unintentionally on the ground or any other lower surface". This definition has been adopted by many fall aversion and fall-risk assessment studies, and it covers most types of falls targeted by fall detection research. The risks of fall-related problems increase with advancing age. As elderly people become physically weaker, the risk of falling anytime increases. A study by the World Health Organization [2] shows that 30%–50% of elderly people fall each year, and of these falls 10%–20% may lead to serious injury or even death [3].

We can find several examples of research that has focused on understanding fall-related problems. For example, the problems of fall-related injuries among the elderly have been investigated to provide a systematic approach to fall prevention [4]. The investigation of prediction accuracy is important to predict falls in older people living in residential care facilities [5]. Population-based interventions for reducing injuries related to falls among older people is necessary [6]. Fall prevention strategies for assisted living residents as well as the importance of evidence-based fall assessment and interventions to reduce the risk of falling were discussed in [7]. Detailed study of the application of wireless sensor networks to real-world habitat monitoring is essential to improved the identification of falls using wireless networks [8].

This work focused on falls in elderly people who are not chronic disease patients. In this research, we used a body sensor network of Shimmer sensors to analyse falls in the elderly. In the proposed system, real-time data obtained from Shimmer sensors is used for analysis to detect certain activities that could lead to falls in the elderly. Feature selection is done in such a way that accelerometer and gyro meter data can be gathered from the Shimmer sensors. Our dataset was gathered through the involvement of 140 subjects, and it was categorized based on their ages and weights; the subjects performed certain activities that could lead to falls. Furthermore, we implemented and analyzed the shimmer sensor data using three classification methods, namely, support vector machine (SVM), K-nearest neighbor (KNN), and neural network for various scenarios.

In this section, we present a selected review of fall detection systems and methodologies. Sposaro and Tyson [9] proposed an Android-based fall application, iFall, which uses Android phones having accelerometer sensors to gather data for analysis based on a thresholding algorithm. In their paper, Yavuz et al. [10] presented a fall detector that uses an integrated accelerometer on smartphones, in which they use the discrete wavelet transformation algorithm for fall detection. In their conference paper, Tacconi et al. [11] proposed a ubiquitous sensing user-friendly device, called iTUG. Neggazi et al. [12] suggeted a compression sensing approach in which they use Intel's Shimmer as a wireless wearable sensing platform which uses 3D acceleration data as well as ECG and gyroscope data. Ojetola et al. [13] presented a description of datasets and experimental protocols designed for the simulation of falls and near-falls, and activitiesof daily living (ADL) simulation was carried out in a laboratory environment using wearable inertial sensors (acceleration and angular velocity). Dau et al. [14] introduced a classifier based on a genetic programming learning-based approach for fall detection, which finds the discrimination rule to separate falls from non-falls. Rakhman et al. [15] presented a prototype for a ubiquitous fall detection and alert system, uFast, which uses a smartphone. Soto-Mendoza et al. [16] proposed a scheduling system for improving the lives of the elderly. Feldwieser et al. [17] conducted a clinical study to detect falls and analyzed their effects and consequences with the help of a fall protocol using an accelerometer as well as optic and acoustic sensors. Kansiz et al. [18] evaluated the time-domain features to determine the most discriminative features from supervised learning methods for fall detection. Brezmes et al. [19] developed a real-time monitoring system of human activities using a single cell phone equipped with an accelerometer, a magnetometer and an ambient light sensor. Dai et al. [20] propose a platform for a pervasive fall detection system using mobile phones, called PerFallId. Kwolek and Kepski [21] adopted the depth map approach to improve fall detection.

Figure 1 shows the percentages of methodologies used for the work mentioned in this section. It is quite clear that the threshold-based algorithm has been used in most of this research.

Since most of these studies have used an accelerometer-based range of values in their evaluations, it can be the case for such a high percentage of threshold-based algorithms. Table 1 compares the studies that included threshold-based algorithms. As seen in Table 1, most of these studies used mobile phones to acquire data due to their easy availability. However, there are certain issues related to the reliability of readings obtained from cell phones due to their placement

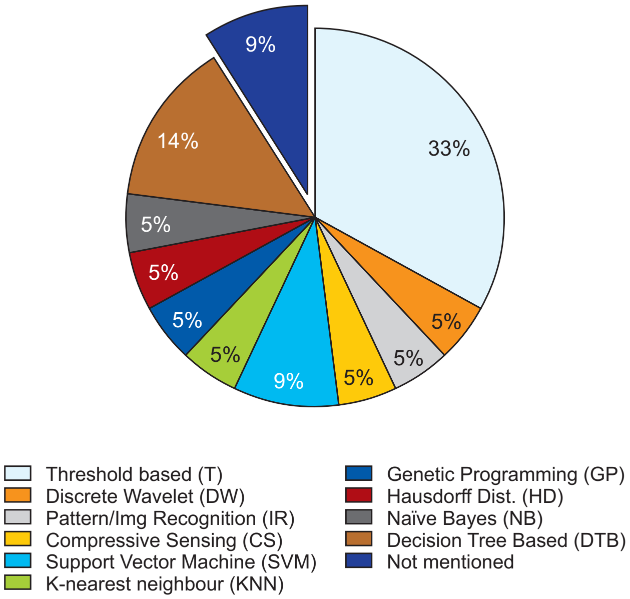


Figure 1. Percentage of methodologies used.

Table 1. Comparison of related work

Reference	Year	Methodology	Devices used	Experimental subjects	Placement of sensors	Parameter
Sposaro and Tyson [9]	2009	Threshold based	Android phone	N/A	Anywhere	Accelerometer
Tacconi et al. [11]	2011	Threshold based	Smartphone	3 young healthy	Waist	Tri-axial accelerometer
Dai et al. [20]	2010	Threshold based, Shape context and Hausdorff distance	Smartphone	Dummy + 15 young	Chest, waist, thigh, legs, pant/shirt pockets	Accelerometer, magnetometer
Yavuz et al. [10]	2010	Discrete wavelet transformation, threshold based	Smartphone	5 healthy	Pocket	Accelerometer
Brezmes et al. [19]	2010	Pattern recognition, SVM	Smartphone	N/A	Anywhere	Accelerometer, magnetometer, light sensor
Kwolek and Kepski [21]	2015	Threshold based, image recognition, extraction based, KNN, SVM	Kinect sensors, Smartphone	5 persons (over 26 years)	Pelvis	Accelerometer, depth sensors
Ojetola et al. [13]	2015	Decision tree based	Shimmer Sensor	42 volunteers	Chest and thigh	Accelerometer, gyroscope
Dau et al. [14]	2014	Genetic programming, threshold based	Smartphone	1 (teen male)	Tight/loose pant pockets	Accelerometer, magnetometer, gyroscope
Rakhman et al. [15]	2014	Ubiquitous based	Smartphone	N/A	Left chest pocket	Accelerometer, gyroscope
Kansiz et al. [18]	2013	Decision tree based, Naïve Bayes	Smartphone	8 subjects	Pocket	Accelerometer
Soto-Mendoza et al. [16]	2015	Decision tree based	Smartphone, fixed sensors for acquiring images	19 (non-participation interview based)	Anywhere	Accelerometer, location sensors, proximity, pressure sensors
Neggazi et al. [12]	2014	Compressive sensing	Intel Shimmer	5 (different ages, 22–58)	Not mentioned	Accelerometer, ECG, gyroscope

SVM: support vector machine, KNN: K-nearest neighbour, ECG: electrocardiogram, N/A: not available.

and the uncontrolled environment. Also, in most of these previous studies, sensors have been placed on the subject's waist for reliability [22]. Therefore, in this work, we use Shimmer sensors, which are specifically designed for these types of applications. The rest of this paper is organized as follows. Section II presents a summary of related work and the methodology used in developing the proposed fall detection system. We present the fall detection prototype including its architecture and core functionalities in this section. In Section III, we report the analysis and evaluation of the prototype. Finally, we summarize our findings and suggest directions for future work in Section IV.

II. Methods

In this section, we propose the mechanism of our system on the basis of the following assumptions. (1) We consider elderly people with no chronic health problems, such as diabetes or cardiac disease. (2) We consider 3 to 4 ADL that could lead to falls in the elderly. (3) We assume that the proposed system works under a controlled environment, i.e., external factors or noise could not change the state of the system. (4) We categorized our data set on the basis of age and weight groups to ensure the accuracy of our analysis.

1. Prototype

To build the prototype, we used the Shimmer3 IMU evalu-

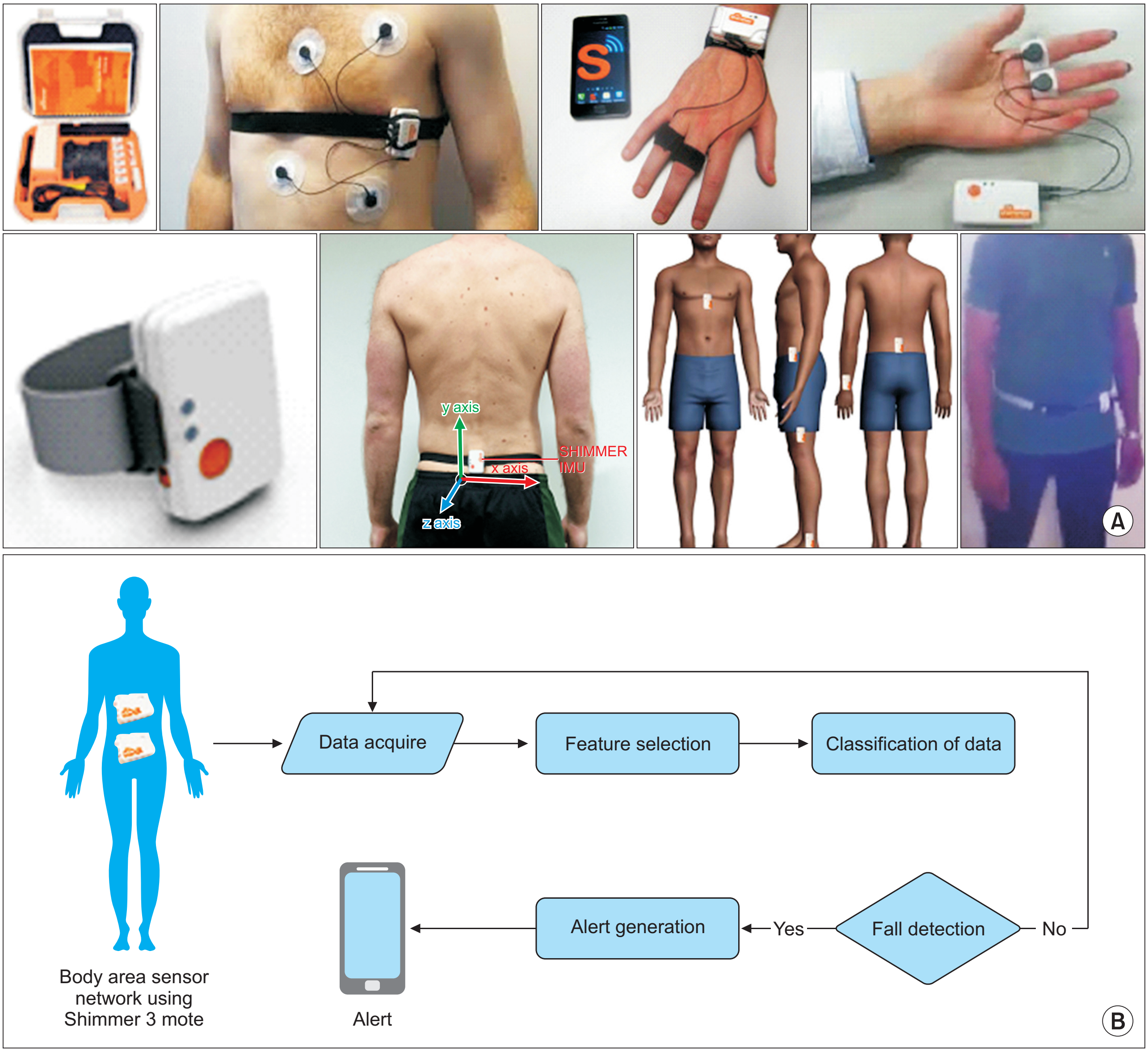


Figure 2. Using Shimmer for fall detection. (A) Deployment of Shimmer kit. (B) Proposed system architecture.

ation kit, as shown in Figure 2A, with inertial measurement sensors. Keeping in view the recent advancement and assessments in wireless body area sensor network (WBASN) technology, we use Shimmer sensors placed on the body as the WBASN for the proposed fall detection system. Figure 2B shows the architecture of the proposed system prototype and its core functionalities. The WBASN of Shimmer sensors is shown in the diagram on the left. In this sense, the real-time monitoring of elderly subjects performing certain activities was done for both data collection and classification. Shimmer sensors were selected because they are well-designed, wearable wireless sensors that provide superior data quality, which is important for the data accumulation process. They offer the best data quality with 10 DoF (degree of freedom) inertial sensing via acceleration, a gyrometer, magnetometer, and altimeter, each with selectable range [23]. Hence, many researchers, such as Park et al. [24] have worked to develop a web-based information system for the elderly; their efforts have focused on contributing to health promotion for seniors and providing a community for elderly health. Moreover, Haghi et al. [25] briefly reviewed the usability of wearable sensors in a hospital IoT as a reliable tool for healthcare observations.

2. Architecture and Core Functionalities

The proposed fall detection architecture is presented in Figure 2B, which also shows the core functionalities of the proposed system, which are as the following: data acquisition, feature extraction, classification, and alert generation.

1) Data acquisition phase

In this phase, we acquired data from the Shimmer sensors as shown in Figure 2. The dataset was gathered through the involvement of 140 people. These participants were asked to perform several activities, and data was collected using our prototype system. For the proof of concept, in this study, we selected activities that could lead to a fall such as sitting from the standing position, The dataset consisted of 132 samples of standing posture, 138 samples of standing to sitting movement 134 samples of walking movement, and 5 samples of falls, which were collected using male volunteers who were not above 25 years of age. However, these latter samples were excluded later as our interim analysis suggested that these were outliers and could not contribute to fall detection with respect to the other samples. Moreover, only one of the five fall samples was pertinent enough to be used for future analysis of falls. We collected data for each of the samples for 5 seconds. To be more precise about the present research, we considered different locations for collecting data. It included the Edhi Foundation old age home in Karachi, the old age home Dar-ul-sukoon, Federal Urdu University of Arts Science and Technology (Karachi), and the Sheikh Zayed Spe-

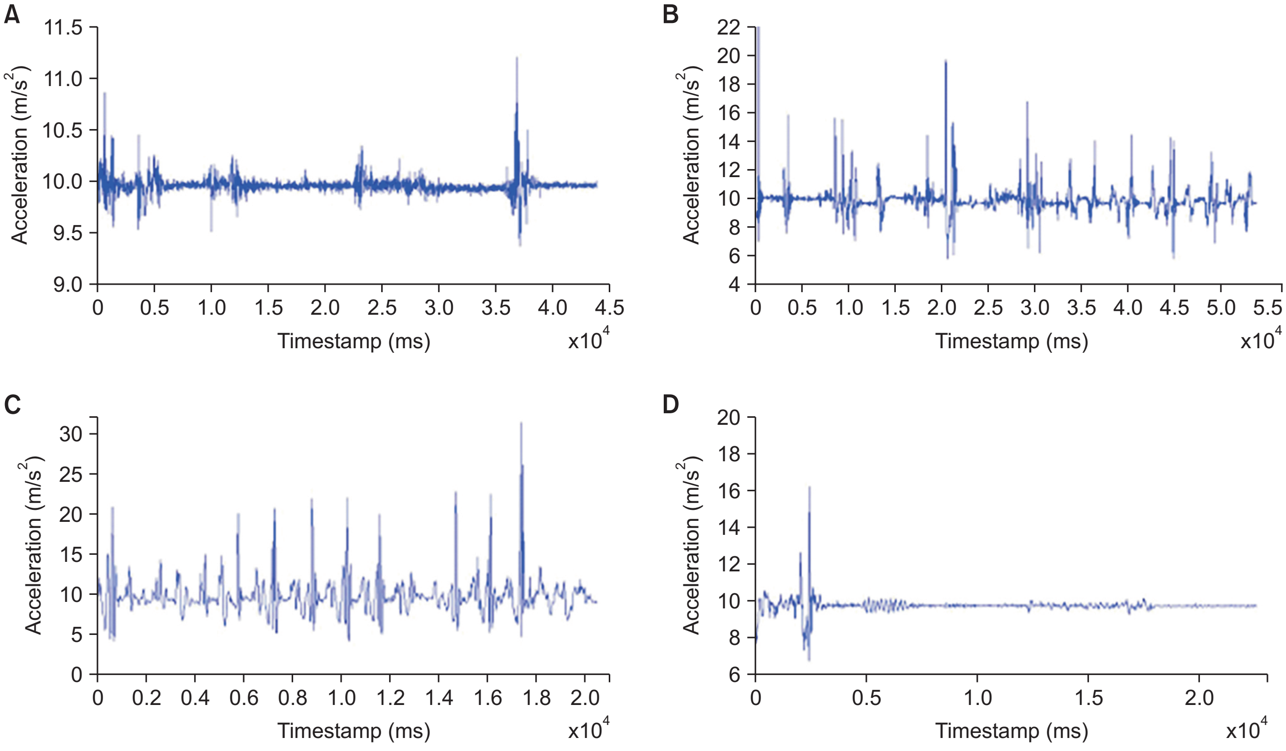


Figure 3. Acceleration over time while standing (A), standing to sitting (B), walking (C), and falling (D).

cial Education Centre (University of Karachi) for elderly people in Pakistan. This entire process of data acquisition has resulted in the data set called "SMotion" [26].

For the categorization of the dataset, we categorized the various items using age and weight groups. Here, we made 3 age groups with intervals of 31–40, 41–50, and above 50 years. Whereas, interval of weight groups are 30–39, 40–49, 50–59, 60–69, 70–79, 80–89, and 90–99 kg. Since, we are more concerned with the elderly, so we did not include young adults. We categorized the data according to different weight groups to recognize the overweight and underweight of elderly because these people have a greater risk associated with falling than other weight groups.

2) Feature selection phase

We collated data using Shimmer inertial measurement sensors, such as triaccelerometer, trigyrometer, temperature, pressure, and magnetometer sensors. In the data acquisition phase, we constructed the SMotion dataset, considering certain activities that can lead to falls in the elderly. The data is stored as CSV files that consist of raw and calibrated data obtained from accelerometer, gyrometer, pressure, temperature, and magnetometer sensors. For the classification and detection of falls using this dataset, we selected only low-noise raw data of the x, y, z axes of the accelerometer and gyrometer sensors after initial pre-processing as they are the most relevant features. The total acceleration can be calculated as

$$Acc_i = \sqrt{Accx_i^2 + Accy_i^2 +}.$$

Similarly, the total gyroscope value can be calculated as

$$Gyr_i = \sqrt{Gyrx_i^2 + Gyry_i^2 + Gyrz_i^2}.$$

The graphs shown in Figure 3 justify the selection of features in all four activities. It also shows the variation of sense values over a specific time during standing, standing to sitting, walking, and fallin, respectively.

3) Classification phase

In the classification phase, we employed key classification algorithmsona selected feature of the dataset to classify and then predict falls in the elderly. Since we are concerned with fall detection in the elderly, we used the machine learning approach to form a model that can predict and learn from the process. In the supervised learning approach, each observation is assigned a label or response.It is then the task of the classification model to predict a response from known responses for each observation from the given dataset. We used SVM and KNN classification in the supervised learning approach. An SVM classifies data by finding the hyperplane on the basis of the best-fit margin and separating data points of one class from those of another. The mathematical definition for SVM is given as

$$y(x) = sign\left[\sum_{i=1}^{k} \alpha_i y_i \psi(x, x_i) + b\right].$$

This classifier is constructed based on the following assumptions:

$$\{w^t\varphi(x_i) + b \geq 1, \text{ if } y_i = P\},$$

$$\{w^t\varphi(x_i) + b \leq -1, \text{ if } y_i = N\},$$

which is equivalent to $y_i\{w^t\varphi(x_i) + b \geq 1, \text{ where } i = 1, \ldots, k$ [27].

If there is a set M of x points and a distance function, KNN search lets us find the k closest points in M to a query point or set of points N. In the dataset, we used the k-fold cross-validation method to select the portion of data using the number of folds or divisions. The model was trained for each fold using all the data outside the fold. The testing was done for each model using the data inside the fold. We used the k-fold cross-validation method to check the accuracy of each given algorithm, and we compared them as shown in Figure 4.

In the classification phase, we used neural network as a supervised learning-based algorithm. Neural networks consist

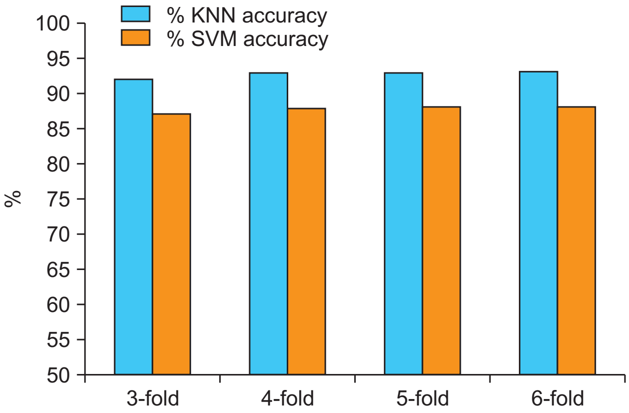


Figure 4. Comparison of support vector machine (SVM) and K-nearest neighbour (KNN) accuracies for k-fold cross-validations.

of simple elements functioning in parallel. These elements are based on biological nervous systems. Also, in nature, the links between elements largely determine the function of a network. A neural network can be trained to accomplish a function by changing the values of the links (weights) between elements.

Normally, neural networks are accommodated, or trained, so that a specific input leads to an exact target output. Here, the network is accommodated, based on evaluation of the output and the target, until the network output matches the target. However, many such input/target sets are required to train a network. Thedataset was divided as follows: 70% training samples, 15% validation samples, and 15% testing sample with 10 neurons per hidden layer.Sincethe dataset was not very large, selecting 70% of data for training produced the desired result. However, we also changed the sampling strategy to check for accuracy. Figure 5 shows various sampling strategies that we have applied to the dataset. It shows that different numbers of samples for different training, testing, and validation setsproduced different mean square errors (MSEs). The MSE linefor training, testing, and validation also suggests that 70% training data producesthe best results.

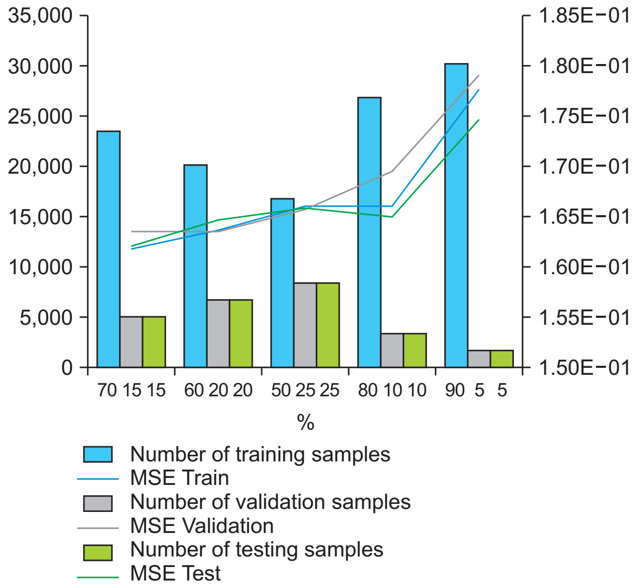


Figure 5. Number of training, testing and validation samples for different samples with mean squared error (MSE) lines.

4) Alert generation phase

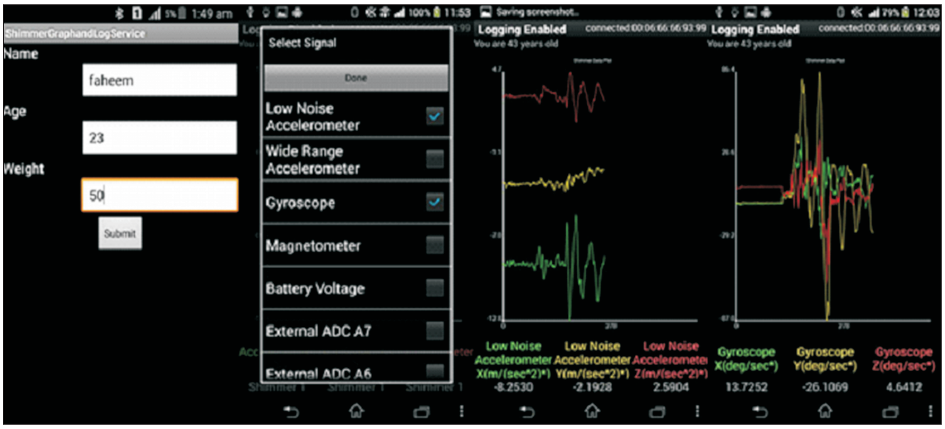
Based on the classification results, this phase defined the interaction of the prototype with end users, i.e., caregivers, doctors, or emergency services. This fall detection prototype was design to be used with a Windows or Android platform. For simplicity, here we show results for the proposed system implemented on an Android app. Figure 6 shows screen shots of the initial version of the Android app prototype testing. The four screens clearly show the phases of the system.

Since Shimmer can easily be calibrated through a Windows or Android phone, this system allow users to sense through the Shimmer device by easy connectivity. Furthermore, this system has ease of access for end users. The screens are designed to provide easy interaction with the system for end users.

Finally, the data can be saved in several file formats for easy analysis using any spread sheet software. Alertsare generated by triggering response values after the analysis phase.

III. Results

In this section, we present analysis and evaluation regarding implementation of the proposed prototype on the dataset. Considering the above-mentioned factors, we now present the system parameter settings for our prototype. After that, we present analysis and results based on the outcomes of datasets by using data visualization, prediction accuracy, and performance evaluation techniques.


ShimmerGraphandLogService

Name: faheem

Age: 23

Weight: 50

Select Signal

- Low Noise Accelerometer ☑
- Wide Range Accelerometer □
- Gyroscope ☑
- Magnetometer □
- Battery Voltage □
- External ADC A7 □
- External ADC A6 □

Logging Enabled connected:00:06:66:66:93:99

You are 43 years old

Low Noise Accelerometer X(m/(sec^2)*)	Low Noise Accelerometer Y(m/(sec^2)*)	Low Noise Accelerometer Z(m/(sec^2)*)
-8.2530	-2.1928	2.5904

Logging Enabled connected:00:06:66:66:93:99

You are 43 years old

Gyroscope X(deg/sec*)	Gyroscope Y(deg/sec*)	Gyroscope Z(deg/sec*)
13.7252	-26.1069	4.6412

Figure 6. Android interfaces for data collection using Shimmer sensor.

1. System Parameters

In this subsection, we will present the system parameter settings for our prototype. We will further subdivide this section into two categories: general simulation parameters and configuration parameters.

1) General simulation parameters

For general simulations, we present here different parameters to define our system setting. The simulator we used for analyzing our datasets was MATLAB version R2015a on 64-bit Windows operating system. We used the 'Classification Learner' and 'Neural Network Pattern Recognition' toolbox to calculate the overall accuracy of the dataset for the corresponding classifiers.

2) Configuration parameters

For the configuration parameters, we present different scenarios for each classifier. Configuration of scenario 1 is, 'number of neighbours' are (3,5,10,100), 'cross-validation folds' are (6,5,4,3), 'distance metric' set to Euclidian, and 'distance weight' set to equal.

Configuration of scenario 2 is, 'kernel function' was Gaussian, 'box constraint level' was 1, 'kernel scale mode' was auto and 'multiclass method' set to one-vs-one. Finally, configuration of scenario 3 is, 'number of hidden layer' is 1, 'number of neurons per hidden layer' is 10, 'number of neurons per input layer' is 6, 'number of neurons per output layer' is 3–6, 'data division' set to random, 'training mode' is Levenberg-Marquardt and 'performance' measured on MSE.

In the KNN classifier, we set different numbers of neighbours and different cross-validation folds to check for accuracy on each fold. The distance metric was set to Euclidian as weights were assigned equal distance. For different neighbours and cross-validation folds, Figure 7 shows that, as the number of neighbours increase, the accuracy of the KNN classifier decreases. This is because when there are too many neighbours of different classes, it leads to an ambiguous result. Similarly, an increase in cross-validation folds increases the accuracy of the classifier because the number of training samples in each fold increased.

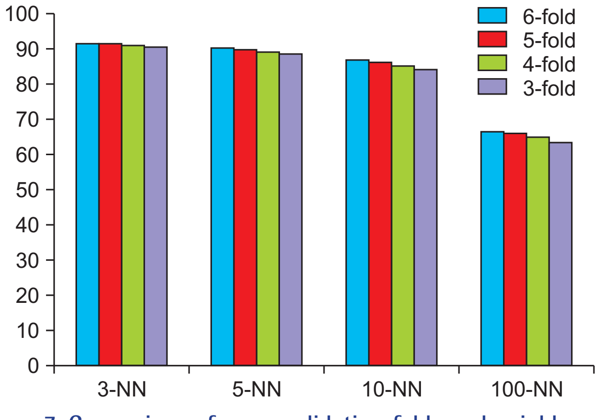


Figure 7. Comparison of cross-validation folds and neighbors in K-nearest neighbor classifier.

In the SVM classifier, weset the kernel function to be Gaussian or a radial basis function that identifies classes with finely-detailed distinctions with the kernel set to sqrt(P)/4, where P is the number of predictors. The box constraint level allows values of the Lagrange multipliers in a box of a bounded region. An increase in the box constraint level results in a decrease in the number of support vectors, but the training time eventually increases. The kernel scale mode is set to auto because it uses the heuristic procedure to select the scale values. To reproduce the result in the kernel scale mode, the random number seed is set since the heuristic procedure uses subsampling. The multiclass method we used in SVM was one-to-one, because it uses one learner for each pair of classes and distinguishes one class from the other class.

For neural network, we used a feedforward network that contains a series of layers. The first layer is the input layer; in our case, it consists of 6 neurons, which are the 3-axial accelerometer and gyrometer obtained information from the dataset. The next layer is a hidden layer, which consists of 10 neurons. Each neuron is used to compute the fitting behaviour for the input layer by using sigmoid functions. In the next layer, the hidden neurons transfer information to the output layer, which then transfers the output using a linear function. In the output layer, there are 3–6 neurons which represent different classes associated with the dataset. For data division, we used a random function. Levenberg-Marquardt training updates the weights and biases in each layer using Levenberg-Marquardt optimization, which is the fastest optimization algorithm for supervised learning, but it requires more memory. This training algorithm requires a MSE performance function.

Table 2. Comparison of classifiers

Algorithm	Prediction speed	Fitting speed	Memory usage	Interpretation
KNN	Fast	Fast	High	Medium
SVM	Slow	Medium	High	Medium
NN	Medium	Medium	High	Complex

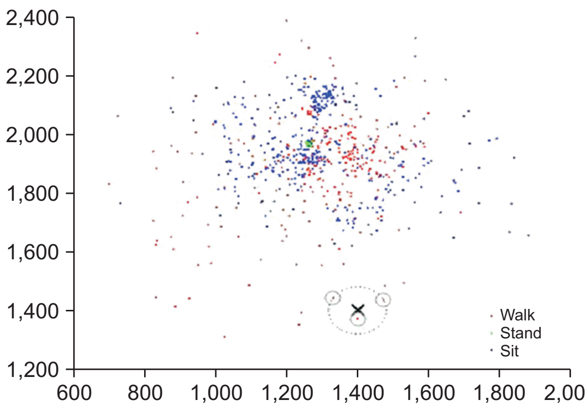


Figure 8. K-nearest neighbor classification for different activities.

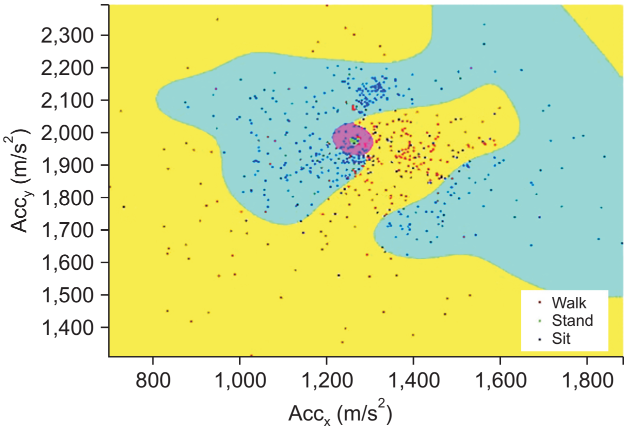


Figure 9. Support vector machine classification for acceleration regions.

Table 3. Snapshot of dataset values for activity

	Parameters used						Activity performed				
1	AccX	AccY	AccZ	GyrX	GyrY	GyrZ	FallL	FallR	Sit	Stand	Walk
2	3,971	2,054	2,235	0	0	2,235	1	0	0	0	0
540	2,751	1,838	2,008	-662	-1,328	2,008	0	0	1	0	0
840	2,778	1,841	2,341	-250	-252	2,341	0	0	1	0	0
885	2,856	1,866	2,045	0	0	2,045	0	0	0	1	0
1,326	2,800	1,864	2,075	959	-5	2,075	0	0	0	0	1
1,348	2,702	2,128	2,567	-482	863	2,567	0	0	1	0	0

Table 4. Snapshot of movement records in dataset

AccX	AccY	Label
1,272	1,969	Sit
1,260	1,965	Stand
988	1,559	Walk

2. Trade-Off Comparison of Classifiers

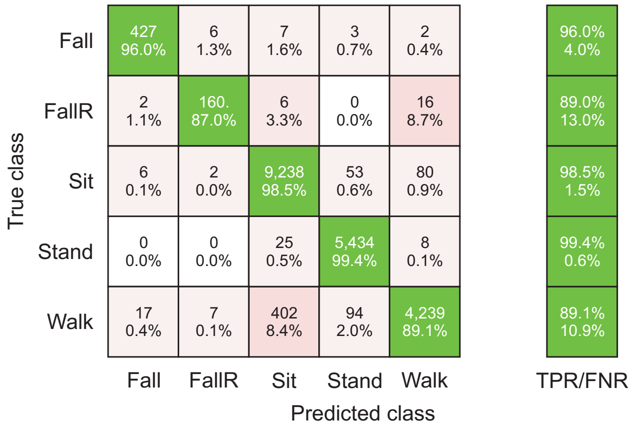
A variety of trade-offs for the different classifiers in this context are compared and shown in Table 2.

3. Data Visualization

In this research, weanalyzed various classifiers and then tabulated the results obtained for SVM, KNN, and neural network. By using data visualization tools, such as MATLAB and R, we visualized the classification regions of SVM and KNN. We used a 5-fold cross-validation method for training dataset.

Figures 8 and 9 show the results, accumulated from 20,236 records in dataset and a snapshot as listed in Tables 3 and 4. Whereas, Figure 8 also shows KNN classification for different activities in which accelerometer data was used to classify 3 nearest neighbours for 3-NN classification for a single person. The region lies in the walking area since all the three neighbours are identified as having a walk response label.

For SVM classification, we used MATLAB multiclass functions, svmtrain and fitcsvm, to correctly classify amongmultiple set of classes. To visualize those classes, we set a fine grid plot within a scatter plot and then predicted the region



True class	Fall	FallR	Sit	Stand	Walk	TPR/FNR
Fall	427 96.0%	6 1.3%	7 1.6%	3 0.7%	2 0.4%	96.0% 4.0%
FallR	2 1.1%	160. 87.0%	6 3.3%	0 0.0%	16 8.7%	89.0% 13.0%
Sit	6 0.1%	2 0.0%	9,238 98.5%	53 0.6%	80 0.9%	98.5% 1.5%
Stand	0 0.0%	0 0.0%	25 0.5%	5,434 99.4%	8 0.1%	99.4% 0.6%
Walk	17 0.4%	7 0.1%	402 8.4%	94 2.0%	4,239 89.1%	89.1% 10.9%

Figure 10. Confusion matrix for overall data. TPR: true positive rate, FNR: false negative rate.

Table 5. Accuracy (%) of different classifiers based on weight groups

Weight (kg)	SVM	KNN	Neural network
30–39	88.7	86.7	79.5
40–49	68.5	72.8	67.5
50–59	82.9	84.8	67.6
60–69	72.6	80.9	67.3
70–79	89.0	92.5	68.0
80–89	70.6	80.9	68.3
90–99	90.6	93.2	77.8

for each class with respect to its corresponding SVM model. Figure 8 shows the classification regions of the three activities we considered. The SVM classification can be clearly seen through the predicted regions seen in Figures 9 and 10. This is a sample image which was obtained by visualization based on thex and y axes data of the accelerometer. Any two axes can be visualized using 2D scatter plot visualization.

For KNN classification, we used the MATLAB fitcknn function from the ClassificationKNN method. We used a scatter plot to visualize different responses. Then a marker was set to identify a new point, Xnew, which predicts the K-nearest neighbours close to Xnew. In our dataset, the number of neighbours was set to 1. We used Euclidean distance metrics calculated as

$$d_{ij}^2 = (x_i - y_j)(x_i - y_j)',$$

where d_{ij} is the distance from i to j, x is the matrix mx by n treated as mx(1 by n) row vectors x_1, x_2, xmx. Similarly, y is the matrix my by n treated as my(1 by n) row vectors y_1, y_2, ymy [28].

4. Prediction Accuracy

We calculated the prediction accuracy of the classifiers based on age and weight groups by testing the model based onthecross-validation techniques discussed above. Table 5 show the result in terms of the percentage of accuracy of different classifiers as calculated for our dataset. It is clear from Table 6 that, at a certain point when the dataset is not very big, KNN and SVM give the best results, whereas neural network gives an average result. On the other hand, SVM gives data with nearer accuracy to KNN but it takes a bit more time to evaluate. Thus, based on these observations, we concluded that KNN gives the best possible result in the least possible time in comparison to other classifiers.

Table 6. Accuracy (%) of different classifiers based on age groups

Age (yr)	SVM	KNN	Neural network
31–40	90.8	92.6	68.6
41–50	86.7	87.9	67.9
Above 51	93.3	95.7	74.3

5. Performance Evaluation

We further analysed the classification performance using performance evaluation-based metrics of sensitivity and specificity. Sensitivity is the capability of identifying all real falls, which can be measured as

$$Sensitivity = \frac{True\ positives}{True\ positives + False\ negatives},$$

whereas specificity is the capability of identifying only real falls, which can be measured as

$$Specificity = \frac{True\ negatives}{True\ negatives + False\ positives}.$$

Figure 10 shows the confusion matrix of overall data and applying the most accurate classifier, i.e., KNN to the dataset, the sensitivity of elderly data was 94% and the specificity was 96.23%. Such high sensitivity and specificity also suggest that KNN has most true positives and true negatives from the dataset.

IV. Discussion

Considering the high mortality rate due to falls in the elderly we focused on developing a fall detection system prototype. We utilized third-generation Shimmer sensor in a wireless body area sensor network connected with the application. We first built the SMotion dataset of some activities performed bytheelderly that could lead to falls. In this work, we employed the most commonly use classification techniques such as SVM, KNN, and neural network as a supervised learning approach. The data was visualized using SVM and KNN classification. The evaluation results of this study identified KNN as the most accurate classifier with the maximum accuracy of 96% for age groups and 93% for weight groups as shown in Tables 5 and 6, respectively. We also presented the confusion matrix and sensitivity and specificity of implementation for in-depth analysis. We also tested the end user experience using the Android app. Finally, we will focus on fall risk estimation using the Shimmer EMG sensor kit in the future.

Conflict of Interest

No potential conflict of interest relevant to this article was reported.

Acknowledgments

We acknowledge the collaboration of old age home administration for facilitating our data collection and testing process. We also acknowledge the dean research grant, FUUAST, which has allow us to import the Shimmer sensor kit for the project. Finally, we are thankful to the University of Karachi for collaboration in this project and acknowledge that this work is extracted from the thesis of Moiz Ahmed (first author) in partial fulfilment of the BS (CS) requirement in Department of Computer Science, University of Karachi.

References

1. Pannurat N, Thiemjarus S, Nantajeewarawat E. Automatic fall monitoring: a review. Sensors 2014;14(7):12900-36.
2. Yoshida S. A global report on falls prevention epidemiology of falls. Geneva, Switzerland: World Health Organization; 2007.
3. Rubenstein LZ. Falls in older people: epidemiology, risk factors and strategies for prevention. Age Ageing 2006;35(Suppl 2):ii37-ii41.
4. Sattin RW. Falls among older persons: a public health perspective. Annu Rev Public Health 1992;13(1):489-508.
5. Rosendahl E, Lundin-Olsson L, Kallin K, Jensen J, Gustafson Y, Nyberg L. Prediction of falls among older people in residential care facilities by the Downton index. Aging Clin Exp Res 2003;15(2):142-7.
6. McClure R, Turner C, Peel N, Spinks A, Eakin E, Hughes K. Population-based interventions for the prevention of fall-related injuries in older people. Cochrane Database Syst Rev 2005;(1):CD004441.
7. Mitty E, Flores S. Fall prevention in assisted living: assessment and strategies. Geriatr Nurs 2007;28(6):349-57.
8. Mainwaring A, Culler D, Polastre J, Szewczyk R, Anderson J. Wireless sensor networks for habitat monitoring. Proceedings of the 1st ACM International Workshop on Wireless Sensor Networks and Applications; 2002 Sep 28; Atlanta, GA. p. 88-97.
9. Sposaro F, Tyson G. iFall: an Android application for fall monitoring and response. ConfProc IEEE Eng Med Biol Soc 2009;2009:6119-22.
10. Yavuz G, Kocak M, Ergun G, Alemdar HO, Yalcin H, Incel OD, et al. A smartphone based fall detector with online location support. Proceedings of International Workshop on Sensing for App Phones; 2010 Nov 2; Zurich, Switzerland. p. 31-5.
11. Tacconi C, Mellone S, Chiari L. Smartphone-based applications for investigating falls and mobility. Proceedings of 2011 5th International Conference on Pervasive Computing Technologies for Healthcare (PervasiveHealth); 2011 May 23-26; Dublin, Ireland. p. 258-61.
12. Neggazi M, Hamami L, Amira A. Efficient compressive sensing on the shimmer platform for fall detection. Proceedings of 2014 IEEE International Symposium on Circuits and Systems (ISCAS); 2014 Jun 1-5; Melbourne, Australia. p. 2401-4.
13. Ojetola O, Gaura E, Brusey J. Data set for fall events and daily activities from inertial sensors. Proceedings of the 6th ACM Multimedia Systems Conference; 2015 Mar 18-20; Portland OR. p. 243-8.
14. Dau HA, Salim FD, Song A, Hedin L, Hamilton M. Phone based fall detection by genetic programming. Proceedings of the 13th International Conference on Mobile and Ubiquitous Multimedia; 2014 Nov 25-28; Melbourne, Australia. p. 256-7.
15. Rakhman AZ, Nugroho LE. u-FASt: ubiquitous fall detection and alert system for elderly people in smart home environment. Proceedings of 2014 Makassar International Conference on Electrical Engineering and Informatics (MICEEI); 2014 Nov 26-30; Makassar, Indonesia. p. 136-140.
16. Soto-Mendoza V, Garcia-Macias JA, Chavez E, Martinez-Garcia AI, Favela J, Serrano-Alvarado P, et al. Design of a predictive scheduling system to improve assisted living services for elders. ACM Trans Intell Syst Technol 2015;6(4):53.
17. Feldwieser F, Gietzelt M, Goevercin M, Marschollek M, Meis M, Winkelbach S, et al. Multimodal sensor-based fall detection within the domestic environment of elderly people. Z Gerontol Geriatr 2014;47(8):661-5.
18. Kansiz AO, Guvensan MA, Turkmen HI. Selection of time-domain features for fall detection based on supervised learning. Proceedings of the World Congress on Engineering and Computer Science (WCECS); 2013 Oct 23-25; San Francisco, CA.
19. Brezmes T, Rersa M, Gorricho JL, Cotrina J. Surveillance with alert management system using conventional

cell phones. Proceedings of 2010 5th International Multi-Conference on Computing in the Global Information Technology (ICCGI); 2010 Sep 20-25; Valencia, Spain. p. 121-5.

20. Dai J, Bai X, Yang Z, Shen Z, Xuan D. Mobile phone-based pervasive fall detection. Pers Ubiquitous Comput 2010;14(7):633-43.
21. Kwolek B, Kepski M. Improving fall detection by the use of depth sensor and accelerometer. Neurocomputing 2015;168:637-45.
22. Lee JK, Robinovitch SN, Park EJ. Inertial sensing-based pre-impact detection of falls involving near-fall scenarios. IEEE Trans Neural Syst Rehabil Eng 2015;23(2):258-66.
23. Shimmer. Simmer2 IMU unit [Internet]. Dublin, Ireland: Simmer; c2017 [cited at 2017 Jul 1]. Available from: http://www.shimmersensing.com/shop/shimmer3.
24. Park H, Kim HJ, Song MS, Song TM, Chung YC. Development of a web-based health information service system for health promotion in the elderly. J Korean Soc Med Inform 2002;8(3):37-45.
25. Haghi M, Thurow K, Stoll R. Wearable devices in medical Internet of Things: scientific research and commercially available devices. Healthc Inform Res 2017;23(1):4-15.
26. Nadeem A. Data repository [Internet]. Karach, Pakistan: Adnan Nadeem; c2016 [cited at 2017 Jul 1]. Available from: http://adnan-nadeem.com/data-repository/.
27. Suykens JA, Vandewalle J. Least squares support vector machine classifiers. Neural Process Lett 1999;9(3):293-300.
28. Neural network overview [Internet]. Natick (MA): MathWorks; c2017 [cited at 2017 Jul 1]. Available from: http://www.mathworks.com/help/nnet/gs/neural-networks-overview.html.