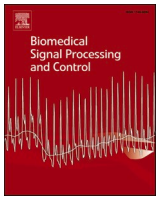
Biomedical Signal Processing and Control 71 (2022) 103242

Contents lists available at ScienceDirect

Biomedical Signal Processing and Control

journal homepage: www.elsevier.com/locate/bspc



Deep learning based fall detection using smartwatches for healthcare applications

Gökhan Şengül [a], Murat Karakaya [a], Sanjay Misra [b], Olusola O. Abayomi-Alli [c], Robertas Damaševičius [c,*]

[a] Department of Computer Engineering, Atilim University, Ankara, Turkey
[b] Department of Computer Science and Communication, Ostfold University College, Halden, Norway
[c] Department of Software Engineering, Kaunas University of Technology, Kaunas, Lithuania

ARTICLE INFO

Keywords:
Fall detection
Activity recognition
Smartwatch
Digital health

ABSTRACT

We implement a smart watch-based system to predict fall detection. We differentiate fall detection from four common daily activities: sitting, squatting, running, and walking. Moreover, we separate falling into falling from a chair and falling from a standing position. We develop a mobile application that collects the acceleration and gyroscope sensor data and transfers them to the cloud. In the cloud, we implement a deep learning algorithm to classify the activity according to the given classes. To increase the number of data samples available for training, we use the Bica cubic Hermite interpolation, which allows us to improve the accuracy of the neural network. The 38 statistical data features were calculated using the rolling update approach and used as input to the classifier. For activity classification, we have adopted the bi-directional long short-term memory (BiLSTM) neural network. The results demonstrate that our system can detect falling with an accuracy of 99.59% (using leave-one-activity-out cross-validation) and 97.35% (using leave-one-subject-out cross-validation) considering all activities. When considering only binary classification (falling vs. all other activities), perfect accuracy is achieved.

1. Introduction

Falling is an irregular activity that occurs among older people over 65 years of age and usually results in severe health challenges [45,17]. Fall can be described as a sudden change in the position of the human body from the normal state such as walking, sitting, standing, etc. to a reclining state without control [40]. Some factors that increase the chances of falls among older people such as poor balance, chronic disease, vision problems, cognitive and dementia challenges [1], or Parkinson's disease [44]. The need to employ algorithms from recent technologies in the detection or prevention of this activity to reduce negative consequences is known as activity recognition. Activity recognition has grown its popularity in recent years and is made up of three main components, which include a low power sensing module, feature selection, and classification activities. There are several categories of approaches presented in existing studies to identify human activities especially in fall detection such as vision-based, environmental, and wearable methods [1]. Activity recognition using various sensor modalities has emerged as an innovative technology for real-time observations in elderly healthcare, ambient assisted living (AAL), and surveillance applications in intelligent home environments [41,20].

Several methods have been suggested to recognize human activities. Some of the solutions based on computer vision techniques include image capture, segmentation, tracking, identification, and classification [11]. Other solutions suggest using wearable sensors that are attached to the subject's body to monitor signals of interest such as motion, location, temperature, and heart rate, among others [19]. Some of the widely used wearable sensors used for fall detection include smartphones [18], smartwatches, waist tester [5,21], etc. These devices play a major role in data acquisition and allow the adoption of threshold-based algorithms for improving detection and triggering alarm notification [5,27,48].

Smartphones are popular and affordable devices that are used as a wearable sensor platform, which has several sensors such as accelerometer, gyroscope, magnetometer, temperature, barometric pressure, etc. [24]. However, a smartphone can have a significant usage limitation: it should be carried by the user during the activity. Recently, smartwatches were adopted for different activity recognition purposes due to decreasing prices and increasing capabilities [61]. Smartphones

* Corresponding author.
E-mail addresses: ummihadiza@gmail.com, hadiza_16000717@utp.edu (R. Damaševičius).

https://doi.org/10.1016/j.bspc.2021.103242
Received 17 August 2020; Received in revised form 19 September 2021; Accepted 6 October 2021

and smartwatches have a similar level of accuracy for activity recognition [50]. As smartwatches are mounted on the arm wrist, for certain activities, the sensors on the smartwatch can generate much more related data with the activities, which can lead to higher prediction accuracy [46]. Unfortunately, there are some disadvantages to smartwatches when used as wearable sensors. For example, the resources (computation power, memory capacity, and battery power) of smartwatches are very limited. Thus, collecting sensory data, processing them, and transferring them can consume these limited resources quickly.

The novelty and contribution of this are the following: (1) To increase the number of data samples in the dataset we used the Bica cubic Hermite interpolation, which allowed us to improve the training of the neural network. (2) For activity classification, we have adopted the bi-directional long short-term memory (BiLSTM) neural network. (3) We have received a 99.6% recognition of human activities, while for fall detection a perfect accuracy was achieved. (4) The proposed data augmentation scheme allows a significant improvement of accuracy when only a small number of data samples are available for neural network training.

The organization of the paper is as follows. In Section 2, an overview of related works is presented. In Section 3, materials and methods are described, including implementation, data acquisition, augmentation, classification, result evaluation, and statistical analysis methods. The results are provided and discussed in Section 4, and in Section 5 the conclusion and further work are presented.

2. Literature review

This section discusses current research trends in human activity detection with a major focus on fall detection systems. Several surveys have been conducted to analyze research progress based on different techniques proposed to recognize human activities [1,3,29,35,36,42,51,59]. These surveys focused on the monitoring of fall detection and activity identification using a wireless sensor network (WSN). However, Ren and Peng [45] conducted a recent systematic review based on recent studies on fall detection. The study focused on four main aspects that include the recent literature, fall detection and prevention algorithms considering varieties of sensors and algorithms targeting the minimal power consumption approach and sensor placements.

2.1. Subject tracking

Notable works that used subject-tracking technologies include Yachirema et al. [62], who introduced an Internet-of-Things (IoT)-based system using a 3D-axis accelerometer inserted into a wearable device. The authors used low-power WSNs, intelligent devices, and cloud computing services for data collection. The proposed system was able to increase the detection accuracy, gain, and precision of falls and activate an alert system. A similar study was proposed by Lee et al. [34], whereby tracking location in real time using Google Map was proposed to enhance early intervention in an emergency.

2.2. Machine learning algorithms for human activity recognition

Machine learning algorithms have been deployed in several activity recognition systems, some of which include a study from Tian et al. [53], who proposed an ensemble-based filter feature method using wavelet energy spectrum features for detecting human activity. The study used two classification algorithms based on kNN and SVM to effectively improve the accuracy of activity recognition. The further study by Chelli and Patzold [16] used acceleration sensor and angular velocity data with quadratic support vector machine (QSVM), and ensemble bagged tree (EBT) to achieve perfect fall recognition. Hussain et al. [28] achieved 99.80% accuracy for fall detection using KNNs classifier, while in recognizing different falling activities 96.82% accuracy was achieved using the RF classifier. Balli et al. [10] used to recognize the motions of a person using a smartwatch step counter, gyroscope, accelerometer, and heart rate sensors. Features were obtained from principal component analysis (PCA) and classification using random forest (RF) was applied.

2.3. Deep learning algorithms for human activity recognition

Recently, deep learning-based methods have also been enrolled for human activity recognition. For example, Bianchi et al. [12] achieved a 92.5% accuracy in recognizing six activities from the UCI dataset using CNN classifier. Uddin et al. [55] used the Gaussian kernel-based PCA to extract informative features from various body signals and adopted a deep activity CNN to recognize 12 activities with an accuracy of 93.90%. Zhu et al. [65] achieved up to 96.11% accuracy of a dataset of signals collected from 100 subjects using ensemble CNNs. Gjoreski et al. [23] considered eight locomotion activities (Bike, Bus, Car, Run, Still, Subway, Train, Walk) using smartphone sensor data. They trained deep learning models for deep multimodal spectrotemporal fusion combined with a hidden Markov model to model temporal relationships of activities and reached an F1 score of 94.9%. Jeong and Kim [30] proposed a two-step solution, where segment-level change detection is used to identify activity change, while a fully convolutional network (FCN) is employed to classify actual activity when the change is detected. Casilari et al. [15] suggested a CNN optimized to differentiate falls from other daily activities using raw data obtained from two inertial sensors (gyroscope and accelerometer). Martinez et al. [38] propose using fully FCNN models pre-trained on a pedestrian activity recognition dataset. Then transfer learning was applied to the network for fall recognition, obtaining, an area under the receiver operating characteristic (AUC) of 93.3%. Baldominos et al. [8] used two mobile devices placed on the wrist and in the pocket of subjects to acquire accelerometer and gyroscope data and calculated statistical features, which were used for classification using classical and statistical methods, achieving the best accuracy found of 97.44%. Gharghan et al. [21] presented a sensor-based fall detection algorithm using an artificial neural network (ANN) for detecting falls in elderly individuals. The proposed model was said to reduce power consumption for data-driven algorithms and computational effort. Lu et al. [37] suggested using a three-dimensional convolutional neural network (3-D CNN), which is trained video kinematic data, while the 3-D convolution helps to acquire motion features from the temporal sequence of frames. Voicu et al. [58] used a multilayer perceptron (MLP) to recognize six activities (walking, running, sitting, standing, upstairs, and downstairs) with an average accuracy of 76.8%. Noori et al. [39] performed wearable sensor data fusion on data-level, feature-level, and decision-level and then applied Deep CNNs to perform activity classification., achieving an accuracy of 98.4% on Context-Awareness via Wrist-Worn Motion Sensors (HANDY), and 98.7% on Wireless Sensor Data Mining (WISDM) datasets. Yao et al. [63] used a histogram of the surface normal orientations as low-level features and sparse coding and pooling on the aggregation of polynormals as high-level features. Then, feature fusion is performed and matrix decomposition is applied to perform human-activity recognition. The method was validated on the 3D Action Pairs, and MSR Hand Gesture 3D datasets. Afza et al. [2] used an optical flow algorithm to extract human motion features from videos. The extracted shape and texture features were fused and a Weighted Entropy-Variances approach was used for feature selection. Then, M−SVM was used for the final classification of features into relevant human actions. Finally, Kiran et al. [32] adopted a fusion of deep features extracted from video frames capturing human activities by ResNet-50 deep learning model. Canonical correlation is adopted for deep features fusion. After feature selection using the Shanon Entropy-based threshold function, classification is performed using multiple machine learning classifiers while the approach is validated on five benchmark datasets.

2.4. Summary and evaluation of related work

We further summarized related studies based on methods, contributions, limitations, etc as depicted in Table 1.

Based on overall findings, some of the shortcomings of existing methods in the detection of falls among elderly people include challenges with limited public datasets for elderly falls, challenges of increasing false rate, the inability of some models to differentiate between varieties of human activities posture, etc. Most of the existing human activity recognition methods suffer from failure to account for temporal dynamics of features, which is essential for correct recognition of activities, and the problem of class imbalance when the activities that occur less often are underrepresented in the collected data leading to data imbalance problem and failure to train the classifiers to correctly recognize rare activities.

Currently, the Recurrent Neural Network (RNN) models such as Long-Term Short-Term (LSTM) network are considered state-of-the-art, which can learn complex patterns in time series data [49]. We have selected BiLSTM due to their ability to learn well from sequential input data and to learn patterns in data time series. It has more power than LSTM as it consists of two independent LSTMs, which analyze and learn from two directions of time series simultaneously: one accesses past information in the forward direction and another accesses future information in the reverse direction. Thus, it can offer some benefit in terms of better results to those domains where it is appropriate, such as the human activity recognition domain. Therefore, this study adopted the Bi-LSTM model to effectively improve the fall detection rate and minimize the false alarm rate.

To address the problem of missing data in real-world applications, when sensor signals are corrupted or not read correctly, which makes the neural network training more difficult, as well as the dataset imbalance problem, when some of the activities are underrepresented, we have proposed a novel data augmentation scheme, which allows augmenting the human activity dataset for increasing the representation of minority classes by oversampling.

3. Methodology

This section describes in detail the proposed system and the implementation methods. For this study, the application of the supervised learning-based approach for activity recognition was evaluated and analyzed. We further categorized this section into two modules, the data collection module and the system implementation module.

3.1. Data sampling and processing

We collected sensory data from the six different activities of 15 volunteers and a total of 824 data samples were collected for this study. Volunteers are university students in the age range of 19 to 25 (22.3 ± 1.4) years and have similar characteristics (height: 178.4 ± 6.5 cm, weight 74.5 ± 4.6 kg). The gender ratio of the volunteers with respect to the number of male and female volunteers is ratio 8: 7, respectively. Activities include falling from standing, falling from a chair, sitting on a chair, squatting, running, and walking. The description of the activity presented is labeled A1-A6 with their names and the number of data collection presented in Table 2. Schematics of the simulated fall while standing are illustrated in Fig. 1.

Table 2
The Classified User Activities and duration of collected data.

Activity Tag	Activity Name	Definition of Activity	Number of data collected
A1	Falling while standing	The person stands up, and, then falls down to the floor	153
A2	Falling from a chair	The person sits on a chair and, then, falls down to the floor.	150
A3	Sitting down	The person stands up and, then, sits down on a chair.	153
A4	Squatting	The person stands up and, then, squats down.	156
A5	Walking	The person walks.	109
A6	Running	The person runs.	103

Table 1
Summary of Related Studies.

Reference	Methodology	Solution implemented	Contributions	Limitations	Dataset
Asif et al. [6]	Single Shot Human Fall Detection with Occluded Joints Resilience using Neural network model	Camera	The method is robust. Ability to handle challenging scenes with occlusion.	Relatively low results for F1 score, precision, and recall for both public databases.	Multi-Cam fall dataset [2] and Le2i dataset
Hagui et al. [25]	Coupled Hidden Markov Models	Camera	The model is effective and detects most human falls and presents superiority	Test dataset was on young person and simulating fall differs in elderly people.	Le2i database, Weizman database and Youtube videos
Thanh et al. [57]	The study is ased on inter-integrated circuit and connection between ADXL345 and MCU	Wearable sensor(3-DOF accelerometer)	The method is a simple, and improves accuracy with lower computation cost	Slight degradation in performance when experimented on real elderly individuals	Primary dataset and MobiFall dataset
Harrou et al. [26]	Multi-variate exponentially weighted moving average (MEWMA). SVM, NN, NB and KNN algorithms were used evaluation.	Camera	high accuracy with a reduced computational time	Inability to differentiate between falls.	URFD and FDDdatasets
Aziz et al. [7]	Support Vector Machine (SVM) classifier using a Radial Basis Function (RBF) kernel	Wearable sensor (tri-axial accelerometers)	Higher fall detection and reduced false positive rate	Experiment was conducted on limited fall data	Primary dataset
Noori et al. [39]	Deep Convolutional Neural Networks (CNNs)	Wearable sensor	Sensor data fusion on data-level, feature-level, and decision-level		HANDY, WISDM datasets
Yao et al. [63]	Feature fusion, matrix decomposion	Depth camera	Learning system based on matrix decomposition is used to accelerate training and enhance classification	With increasing number of features, computational complexity does not scale well	3D Action Pairs, MSR Hand Gesture 3D datasets
Afza et al. [2]	Optical flow algorithm, Weighted Entropy-Variances approach, M−SVM	Camera	More accurate image segmentation	High probability of redundant feature extraction in case of multiple actions	Weizmann, KTH, UCF Sports, UCF YouTube datasets
Kiran et al. [32]	ResNet-50, Canonical correlation, Shanon Entropy-based thresholding	Camera	Computationally fast	Requires expert knowledge which network layers should be used for feature extraction	IXMAS, UCF Sports, YouTube, UT-Interaction, and KTH datasets

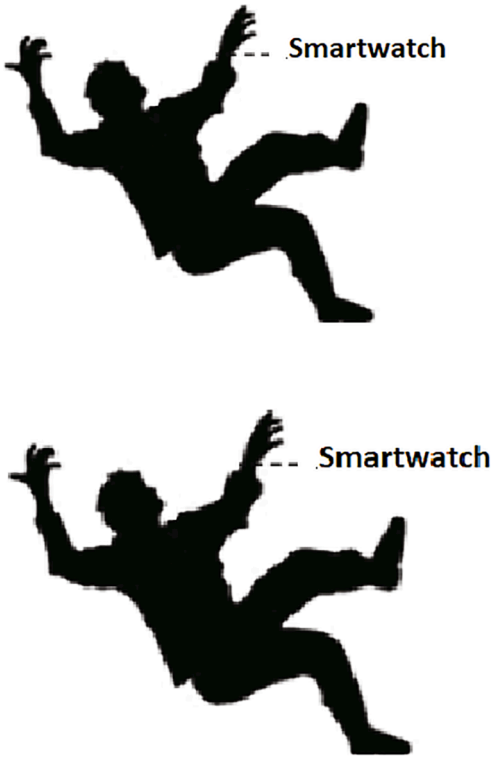


Fig. 1. Schematics of simulated fall from standing position.

For the first four activities, volunteers conducted each activity at least 10 times. Therefore, data from at least 150 activities for each activity are collected from volunteers. For walking and running, volunteers are asked to perform the activity and 1 s samples of those activities are selected. For the collected data, a time frame of 1 s is used, which is enough for the aforementioned activities. In this period, the sensor measurements are sampled with a frequency of 50 Hz. After the data are acquired; mean, min, max, and variance of the power of the sensor data, the maximum value and minimum value of the data at each axis of the sensor, the difference of the data at each axis of the sensor, the variance of the data at each axis of the sensor, and entropy of the data at each axis of the sensor are extracted as the features of that sensor. For each sensor, 19 signal features are extracted, and the process was repeated for the accelerometer and gyroscope sensor data. The features were later used to train a neural network. The above-mentioned approach of learning from handcrafted features rather than raw data has been used before, most recently in [14].

3.2. Data augmentation

As the data set is imbalanced, we augment the data set to increase the representation of minority classes by oversampling. For data augmentation, we use multivariate interpolation in the feature space implemented using Bica cubic Hermite interpolation [13]. The Hermite interpolation formula results when the function and first derivative values are made to agree with the interpolating polynomial at several locations. In this case, $n+1 = 2m$ is an even integer, so there are m function values and m derivative values in the fit. By convention, odd-numbered evaluations are at function values and even-numbered evaluations are at derivatives. The normalized transfer matrix of is:

$$\widehat{\Phi}_H = \begin{bmatrix} 1 & x_0 & x_0^2 & \cdots & x_0^n \\ 0 & 1 & 2x_0 & \cdots & nx_0^{n-1} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & x_{m-1} & x_{m-1}^2 & \cdots & x_{m-1}^n \\ 0 & 1 & 2x_{m-1} & \cdots & nx_{m-1}^{n-1} \end{bmatrix} \tag{1}$$

The transformation matrix above depends only on the sample locations and is constant, independent of the interval length. Eq. (2) relates the vector of normalized coefficients $\boldsymbol{a}_H$ to the vector of Hermite sample values,

$$\widehat{\mathbf{y}}_H = \left(y_0, hy_0^{'}, \cdots, y_m, hy_m^{'}\right)^T = \widehat{\Phi}_H \mathbf{a}_H \tag{2}$$

Solution for $\boldsymbol{a}_H$and multiplication by $\boldsymbol{x}$ produces the Hermite interpolation formula as the quadratic form:

$$\widehat{f}_H(x) = \mathbf{x}^T \widehat{\Phi}_H^{-1} \mathbf{y}_H = \mathbf{x}^T \widehat{\Lambda}_H \mathbf{y}_H \tag{3}$$

The Bica Hermite interpolation is obtained by minimizing quadratic oscillation. For a partition Δ of an interval $[a,b]$, $\Delta : a = x_0 < x_1 < \cdots < x_{n-1} < x_n = b$, the cubic spline is

$$s(x) = A_i(x)\cdot m_{i-1} + B_i(x)\cdot m_i + C_i(x)\cdot y_{i-1} + D_i(x)\cdot y_i, x \in [x_{i-1}, x_i], i = \overline{1,n} \tag{4}$$

with $A_i(x) = \frac{(x_i - x)^2(x - x_{i-1})}{h_i^2}$, $B_i(x) = -\frac{(x - x_{i-1})^2(x_i - x)}{h_i^2}$, $C_i(x) = \frac{(x_i - x)^2[2(x - x_{i-1}) + h_i]}{h_i^3}$, $D_i(x) = \frac{(x - x_{i-1})^2[2(x_i - x) + h_i]}{h_i^3}$, where $h_i = x_i - x_{i-1}, i = \overline{1,n}$, and $y_0, y_1, \cdots, y_n \in \mathbb{R}$.

The quadratic oscillation $\rho_2$of a function $f \in C([a,b], \Delta, y)$ is defined by

$$\rho_2(f) = \sqrt{\sum_{i=1}^{n} \int_{x_{i-1}}^{x_i} \left[f(x) - \frac{x_i - x}{h_i}\cdot y_{i-1} - \frac{x - x_{i-1}}{h_i}\cdot y_i \right]^2 dx}. \tag{5}$$

The iterative algorithm solves the above equation system as follows: Let $b_i = 1, i = \overline{0,n}, a_i = -\frac{3h_i^3}{4\left(h_i^3 + h_{i+1}^3\right)}, i = \overline{1,n-1}, a_n = -\frac{3}{4}, c_0 = -\frac{3}{4}$, $c_i = -\frac{3h_{i+1}^3}{4\left(h_i^3 + h_{i+1}^3\right)}, i = \overline{1,n-1}, v_{-1} = u_{-1} = 0$, and for $i = \overline{0,n}$, we put $p_i = a_i \cdot v_{i-1} + b_i$, $v_i = -\frac{c_i}{p_i}$, $u_i = \frac{d_i - a_i \cdot u_{i-1}}{p_i}$. The solution is obtained by the backward recurrence $\overline{m_n} = u_n, \overline{m_k} = v_k \cdot \overline{m_{k+1}} + u_k, k = \overline{n-1, 0}$.

3.3. Classification

Here, we applied a supervised learning-based approach for the activity recognition. In the proposed method, the features are first extracted from the collected data and then we use the BiLSTM neural network to implement activity classification. As a baseline against which to compare, we use the well-known k-nearest neighbor (KNN) classifier. The KNN classifier assigns a predicted value to a new observation based on the plurality of means (sometimes weighted) of its nearest neighbors in the training set.

The classification is performed for 3 options: using the accelerometer data only, using the gyroscope data only, and using the accelerometer and gyroscope data together. The results obtained for each option and each activity combination are given in Section 5.

The principle of BiLSTM is to divide the neurons of regular LSTM into two directions: one for the forward time direction and the other for the backward time direction (see Fig. 2 and Fig. 3). The output of these two states is not linked to the input of the opposite state. Thus, BiLSTM "looks" not only at the "past" but also at the "future", which allows it to better understand the context of analyzed data [47].

Given the set of training data $T := \{(x,y)\}$, made from the inputs $\times$ and the associated results y, the learning algorithm creates a mapping to deduce the output y, which is intended for the input $x \in T$. Neural networks produce representations using an enclosed layered model, while

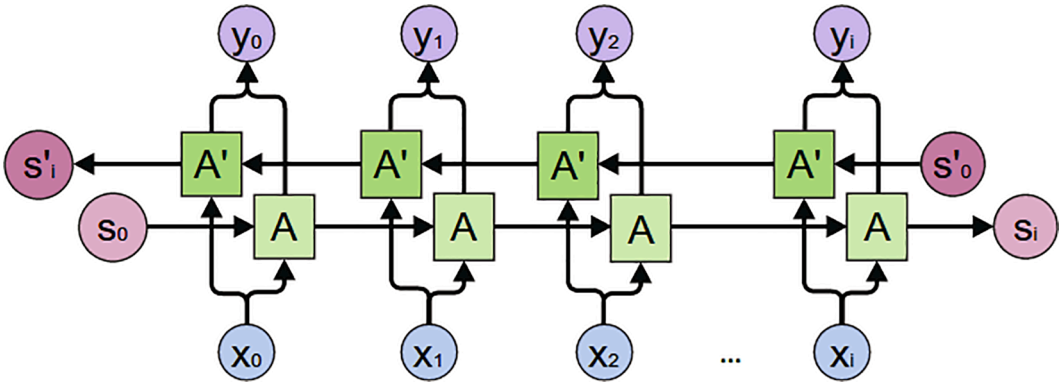


Fig. 2. Architecture of Bi-directional long short-term memory (Bi-LSTM) network model.

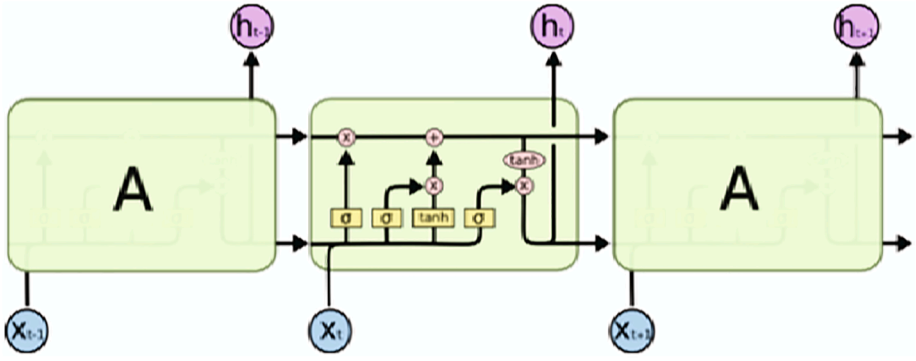


Fig. 3. One cell of the Bi-directional long short-term memory (Bi-LSTM) network.

each layer consists of a linear transform with element-by-element nonlinearity. The first layer starts with a linear transform to obtain an intermediate product output $u_1 := A_1 x_0 = A_1 x$, and then - the current nonlinearity generates the output of the first layer $x_1 := \sigma_1(u_1) = \sigma_1(A_1 x_0)$. This step is repeated recursively so that the transform $x_l := \sigma_l(u_1) := \sigma_l(A_l l x_{l-1})$ is calculated on the n-th layer.

In the network with L-layers, the input $x = x_0$ to the first layer is introduced, and the output $y = x_L$ is received from the last layer. The members of the training set are used to find the matrix, which optimizes the cost of learning the form $(x, y) \in Tf(y, x_L)$, where $f(y, x_L)$is the corresponding metric that estimates the difference between the output x_L at the input x and the target output y. As a rule, the calculation of the optimal coefficients of the NN is performed by a stochastic gradient descent, which is calculated by the propagation algorithm.

To summarizing, the LSTM model is expressed by a set of equations:

$$X = \begin{bmatrix} x_t \\ h_{t-1} \end{bmatrix}$$

$$f_t = \delta\left(W_f \hat{\mathrm{A}} \cdot X + b_f\right)$$

$$i_t = \delta(W_i \hat{\mathrm{A}} \cdot X + b_i)$$

$$o_t = \delta(W_o \hat{\mathrm{A}} \cdot X + b_o)$$

$$c_t = f_t \odot c_{t-1} + i_t \odot tanh(W_c \hat{\mathrm{A}} \cdot X + b_c)$$

$$h_t = o_t \odot tanh(c_t) \tag{6}$$

Where x_t is the network input at t, h_t is the latent state at t; W_f, W_i, W_o, W_c are weight matrices, b_f, b_i, b_o, b_c are biases; $\delta(\hat{\mathrm{A}}\cdot)$ is activation function; and $\odot$ is the dot multiplication.

The latent state H_t of Bi-LSTM at time t composes forward $\overrightarrow{h_t}$ and backward $\overleftarrow{h_t}$:

$$\overrightarrow{h_t} = \overrightarrow{LSTM}(h_{t-1}, x_t, c_{t-1}), t \in [1, T]$$

$$\overleftarrow{h_t} = \overleftarrow{L}\ STM(h_{t+1}, x_t, c_{t+1}), t \in [T, 1]$$

$$H_t = \left[\overrightarrow{h_t}, \overleftarrow{h_t}\right] \tag{7}$$

For classification, the human activity data is formed using the rolling update approach [60]. The main principle of the rolling update is to maintain the length of the data constant, replacing old data with new data in a rolling way. The updated vectors can better represent the current properties of the activity state.

Let $r(i)$ is the activity feature value at time$i(i = 1,\ 2, \ldots, n)$, so that the activity feature data is described as $R = [r(1), r(2), \cdots, r(n)]$. If the current time is t and the number of activity feature sample is s, then the activity feature vector is $R_t = [r(t - s\ +\ 1), r(t - s + 2), \cdots, r(t)]$. When time t increases to $t + 1$, a new activity feature value $r(t\ +\ 1)$ will roll in and the obsolete activity feature value $r(t - s\ +\ 1)$ will be deleted and the vector R_t will become $R_{t+1} = [r(t - s + 2), r(t - s + 3), \cdots, r(t + 1)]$.

3.4. Cross-validation protocols

To assess the performance of a classifier, we adopted two cross-validation protocols on unknown samples. First, we used the leave-one-activity-out approach. In this approach, activity is selected for testing, the remaining activities are used as the training data, and the result is obtained. After then the testing activity is changed and the process is repeated N times following a cross-validation protocol based on random resampling [9], where N is the total number of activities, which is 824 in this study.

Next, to check how well the proposed methodology performs for

previously unseen subjects, we have performed leave-one-subject-out (LOSO) cross-validation [22]. The LOSO cross-validation effectively splits the data set into multiple segments based on the number of participants. One subject at a time is chosen for testing, while the others are used to train the machine learning model. This procedure continues until all subjects have been tested. Finally, overall accuracy is estimated based on all the subjects' folds to demonstrate the approach's performance.

3.5. Performance evaluation

Performance evaluation is based on the number of testing instances rightly and wrongly foreseen by the model. These numbers are depicted in the confusion matrix, which shows how instances belonging to a specific class are assigned by the model to a set of possible classes. Some metrics are used to assess the efficiency of a classification model [52]. The accuracy provides the percentage of correctly classified objects:

$$Acc = \frac{1}{K}\sum_{i}^{K}[f(X_i = y_i] \tag{8}$$

where sampling objects are taken from a pending sample of size K.

F1 is the harmonic mean of precision and recall:

$$F1 = \frac{TP}{2{*}TP + FN + FP} \tag{9}$$

Here TP is true positives, FN is false negatives, and FP is false positives. However, F1 is not a suitable measure for imbalanced datasets [43], therefore we do not use it.

3.6. Statistical analysis of the results

For comparing the performance of classifiers, we use the McNemar's test. The test uses a Chi-square (χ^2) test applied to a 2×2 contingency table, the cells of which contain, true positives, false positives, false negatives, and true negatives as follows:

$$\chi^2 = \frac{\left(|n_{ij} - n_{ji}| - 1\right)^2}{n_{ij} + n_{ji}} \tag{10}$$

where n_{ij} is the number of instances classified wrongly by classifier i, but classified correctly by classifier j, and n_{ji} is the number of instances classified wrongly by classifier j, but not by classifier i. If the test value is greater than the χ value of 3.84 at 95% confidence, then the difference in accuracy between the compared classifiers is statistically significant.

To test the difference among multiple classifiers, we use the nonparametric Friedman test and post hoc Nemenyi test as described in [66]. The Friedman test ranks the classifiers based on their performance. The average ranks allow for a comparison between the classifiers. The performance of two classifiers is statistically different if their mean ranks differ by at least the critical distance:

$$\mathrm{CD} = q_\alpha\sqrt{\frac{k(k+1)}{6N}} \tag{11}$$

Where q_α is the critical value for the Nemenyi test at the significance level α; k is the number of classifiers, and N is the number of samples.

To analyze the significance of features, we use feature ranking, which is made using the t-test based features ranking [64] on the training dataset as follows. Suppose

$$z_j = \frac{|m_j^{+1} - m_j^{-1}|}{\sqrt{v_j^{+1}/n^+ + v_j^{-1}/n^-}}. \tag{12}$$

here $v_j^{+1} = \mathrm{var}(x_{ij}|y_i = +1), v_j^{-1} = \mathrm{var}(x_{ij}|y_i = -1)$, where $\mathrm{var}(\cdot)$ is the variance of a group of values, $m_j^{+1} = \mathrm{mean}(x_{ij}|y_i = +1), m_j^{-1} = \mathrm{mean}(x_{ij}|y_i = -1)$, and $\mathrm{mean}(\cdot)$ is the mean of values.

To rank the features, we also use the Permutation Feature Importance (PFI) which computes the scores for each of the datasets. PFI evaluates the contribution of each feature to the total performance of the method [4]. It is computed as an average decrease in accuracy for all features in each model over all classes, as well as at class level.

3.7. Hardware

Sony SmartWatch 3 SWR50 is used as the smartwatch and a mobile application is developed to collect and transfer sensor data and obtain modifications from the cloud server. The selected smartwatch has two related sensors: an accelerometer and a gyroscope. The properties of the smartwatch are given in Table 3. Note that the selected smartwatch has several sensors, but only accelerometer and gyroscope sensor data are used for activity recognition.

For classification, we used the BLSTM neural network implemented in MATLAB. We have adopted Adam optimizer [31] with 'MaxEpochs', 100 maximum number of epochs for training, a gradient threshold of 1, an initial learning rate of 0.005, piecewise learning rate schedule, a learning rate drop period of 50, and a learning rate drop factor of 0.2. For cross-validation, 70% of data were used for training, and 30% for testing.

3.8. Implementation

We proposed a system based on cloud computing. In the proposed system, the sensors in a smartwatch are used to collect the necessary data about the person in question. Furthermore, the smartwatch is used as an agent to collect data and transfer them to the cloud server via a wireless connection as seen in Fig. 4.

The cloud service is implemented as a web service. The uploaded data is stored temporally on the webserver and then passed to the classification application. The operation of The system is as follows: the accelerometer and gyroscope data of the person in question are acquired by a smartwatch application. The acquired data are transferred to the Web Server in JSON format, and the sensor data are fed to the machine learning-based classifier. The classifier predicts the activity and returns it to the web server. If the predicted activity falls, a warning message is sent back to the mobile application. Since smartwatches have very limited battery power and running machine learning algorithms on them requires high processing power, we opt for installing these methods on a cloud server. We use Apache Web server and develop a PHP-based web application to get sensor data from the smartwatch, transfer the data to the classifier, receive the predicted activity from the classifier, and send modifications to the smartwatch if required.

4. Results

4.1. Data collection results

The data samples collected by the acceleration and gyroscope sensors for several different activities (running and walking are shown) are illustrated in Fig. 5. The 38 features calculated for the acceleration and gyroscope sensor values within 1 sec length frames are illustrated for the

Table 3
Technical specifications of Sony SmartWatch 3 SWR50 device.

Property	Capacity
CPU	Quad ARM A7, 1.2 GHz
Main Memory	512 MB RAM
Battery power / time	420 mA /up to 2 days for normal use
Water protected	IP68
Android release	Android 4.3 and onwards
Weight	45 g
Sensors	Ambient light sensors, Accelerometer, Compass, Gyro, GPS
Connections	Bluetooth® 4.0, NFC, Micro USB, Wi-Fi

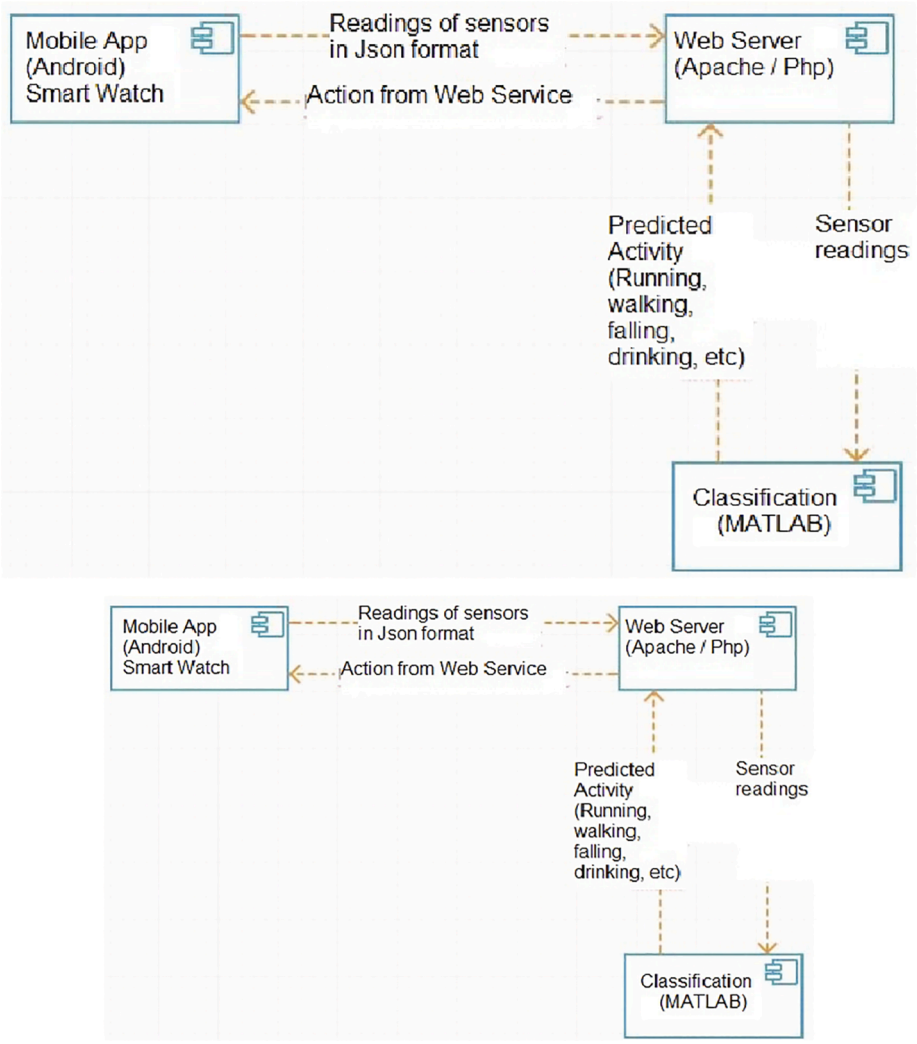


Fig. 4. The deployment model of the fall detection system implementation.

walking activity in Fig. 6.

4.2. Classification results

Since we use two different sensors to collect data, we evaluate their impact on classification. In Tables 4 and 5, we provide results according to the data obtained from each sensor and combined data. In each case, the results for leave-one-activity-out and leave-one-subject-out cross-validation are presented. For comparison, we present the results using baseline machine learning algorithms of k-Nearest Neighbor (KNN), Random Forest (RF), and Support Vector Machine (SVM). In the first experiment, we considered all six activities. As seen in Table 4, the overall precision of the classification using KNN is 78.76%, while BiLSTM achieved an accuracy of 99.59%. Using only acceleration data allows for comparable results with an accuracy of 75.24% (KNN) and 99.19% (BiLSTM), while gyroscope data allow for achieving an accuracy of 62.50% (KNN) and 98.78% (BiLSTM).

As seen in Table 5, BiLSTM achieved an accuracy of 97.35%. Using only acceleration data allows for comparable results with an accuracy of 96.33%, while gyroscope data allow for achieving an accuracy of 95.77%, which is better than the baselines.

In the second experiment, our aim is to observe how accurately falling movements can be classified. An experiment is conducted in which only falling while standing and falling from chair activities are used. To evaluate the performance of differentiating the falling activity from the others, we combined the other activities into a new class, namely the others. The Other activities class includes activities of sitting, squatting, walking, and running. Table 6 shows the performance of the BiLSTM classifier (and the kNN, RF, and SVM classifiers, for comparison) when sitting, walking, and running activities are combined and the classification for falling against the other activities. Falling can be distinguished from the other activities with an accuracy of 95.99% when accelerometer data is used only, 88.35% when gyroscope data are used only, and 97.09% when accelerometer and gyroscope data are combined when using the KNN classifier. In the case of BiLSTM, very good performance was achieved using only the acceleration data or both accelerometer and gyroscope data, but the classifier performed worse than KNN when using only the gyroscope sensor data. This suggests that the data obtained from the gyroscope sensor data is not useful when detecting falling activities.

We present the confusion matrix of the activity classification results using the BiLSTM network model in Fig. 7. Note that there was only one misclassification (using one-activity-leave-out cross-validation), which allowed one to achieve an accuracy of 99.6% on the testing dataset, which is better than the accuracy achieved using the baseline KNN (97.09%), RF (93.54%) and SVM (90.12%) classifiers.

The results of McNemar's test show that the BiLSTM classifier has significantly better accuracy ($p < 0.001$) that the KNN classifier for the 1-vs-All classification of all activities.

Finally, we have performed visualization of the activations of the

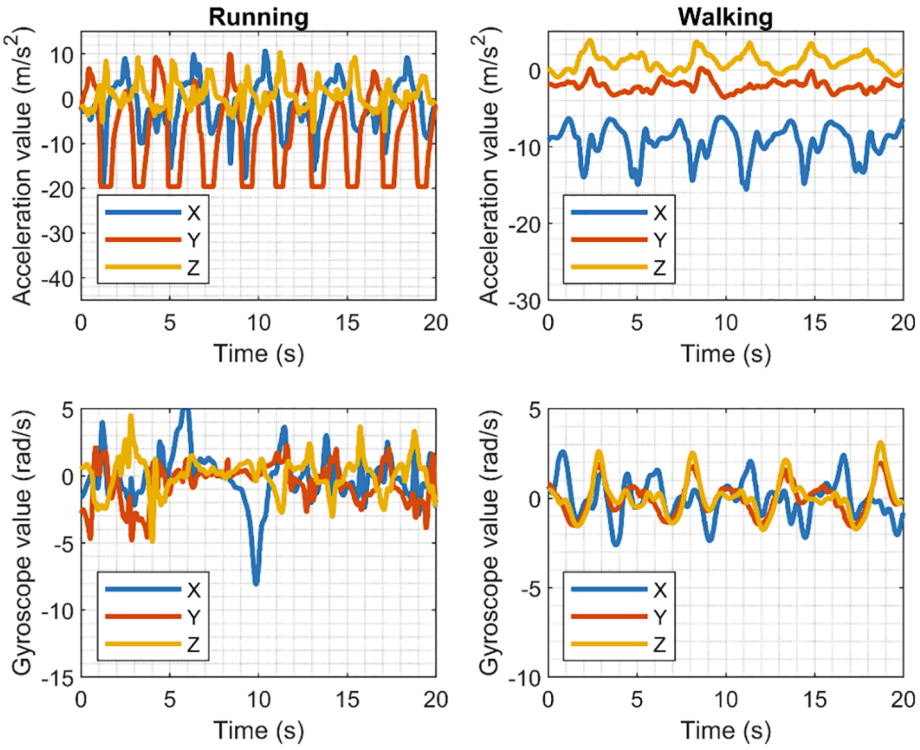


Fig. 5. An example of raw accelerometer and gyroscope sensor data values (in three axes: X, Y, and Z) for running and walking activities.

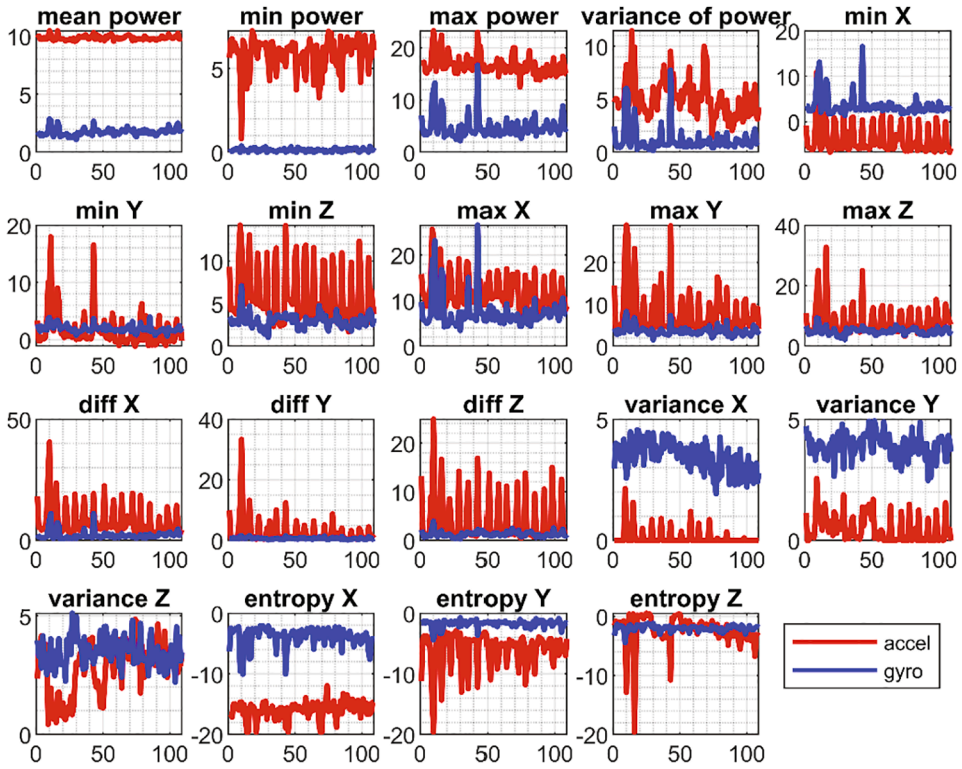


Fig. 6. An example of feature values of the accelerometer (accel) and gyroscope (gyro) sensor data calculated using the rolling update for walking activity.

final BiLSTM network layer using t-distributed stochastic neighbor embedding (t-SNE) [56]. The results presented in Fig. 8 show that the activities are well separated in the lower-dimensional embedding manifold.

4.3. *Feature ranking results*

Feature ranking was performed using the *t*-test method [64] as described in Section 3 and presented in Fig. 9 (the top-6 features sorted according to the Z-score value are presented for each activity). The best

Table 4
Accuracy of human activity classification using machine learning and BiLSTM classifiers (leave-one-activity-out cross-validation).

Activites and numbers of each activities		Classifier	Accuracy (%) Accelerometer Data	Gyroscope Data	Accelerometer & Gyroscope Data
A1	153	KNN	72.55	53.59	69.93
		RF	71.85	60.73	72.15
		SVM	70.35	64.92	67.32
		BiLSTM	100	100	100
A2	150	KNN	71.33	64.00	80.00
		RF	72.94	61.83	74.02
		SVM	70.02	62.45	75.38
		BiLSTM	100	100	97.8
A3	153	KNN	63.40	49.02	68.63
		RF	64.92	54.96	66.73
		SVM	63.92	58.92	64.32
		BiLSTM	100	100	100
A4	156	KNN	64.74	46.79	69.87
		RF	63.04	52.05	67.83
		SVM	61.94	53.05	64.93
		BiLSTM	100	100	100
A5	109	KNN	93.58	85.32	96.33
		RF	92.86	81.38	95.03
		SVM	89.32	78.54	93.92
		BiLSTM	100	92.7	100
A6	103	KNN	99.03	93.29	100
		RF	96.84	93.65	97.35
		KNN	93.02	91.75	94.03
		BiLSTM	93.1	100	100
All	824	KNN	75.24	62.50	78.76
		RF	74.99	65.06	76.84
		SVM	72.85	66.26	74.63
		BiLSTM	**99.19**	**98.78**	**99.59**

Table 5
Accuracy of human activity classification using machine learning and BiLSTM classifiers (leave-one-subject-out cross-validation).

Activites and numbers of each activities		Classifier	Accuracy (%) Accelerometer Data	Gyroscope Data	Accelerometer & Gyroscope Data
A1	153	KNN	66.49	47.37	63.75
		RF	68.21	55.79	67.64
		SVM	66.21	62.19	65.30
		BiLSTM	98.18	97.82	98.37
A2	150	KNN	66.63	58.14	75.69
		RF	70.89	60.87	71.96
		SVM	68.17	60.91	75.01
		BiLSTM	96.74	95.99	97.04
A3	153	KNN	58.34	44.48	64.60
		RF	62.22	52.27	64.98
		SVM	60.51	55.99	61.55
		BiLSTM	95.08	96.52	99.15
A4	156	KNN	62.53	45.44	67.68
		RF	62.03	50.85	67.75
		SVM	60.71	51.69	63.85
		BiLSTM	98.25	98.77	99.02
A5	109	KNN	87.43	80.89	93.10
		RF	89.32	79.29	92.43
		SVM	84.78	75.05	90.83
		BiLSTM	97.23	90.22	97.31
A6	103	KNN	96.11	88.80	97.87
		RF	90.10	89.15	92.53
		KNN	88.12	87.84	89.81
		BiLSTM	87.00	93.15	94.43
All	824	KNN	71.89	59.52	74.70
		RF	74.21	61.06	75.64
		SVM	71.67	62.62	73.74
		BiLSTM	**96.33**	**95.77**	**97.35**

Table 6
Accuracy of the baseline machine learning and BiLSTM classifiers for recognition of falling against other activities.

Activity Name	Classifier	Accelerometer Data (%)	Gyroscope Data (%)	Accelerometer + Gyroscope Data (%)
Falling	KNN	96.70	84.16	97.69
	RF	93.47	81.72	93.27
	SVM	89.56	78.44	91.25
	BiLSTM	100	62.2	100
Other activitiess (sitting, squatting, walking, running)	KNN	95.59	90.79	96.74
	RF	92.83	86.71	93.62
	SVM	88.73	83.24	90.83
	BiLSTM	100	100	100
Total	KNN	95.99	88.35	97.09
	RF	92.03	85.35	93.54
	SVM	87.62	82.94	90.12
	BiLSTM	**100**	**86.23**	**100**

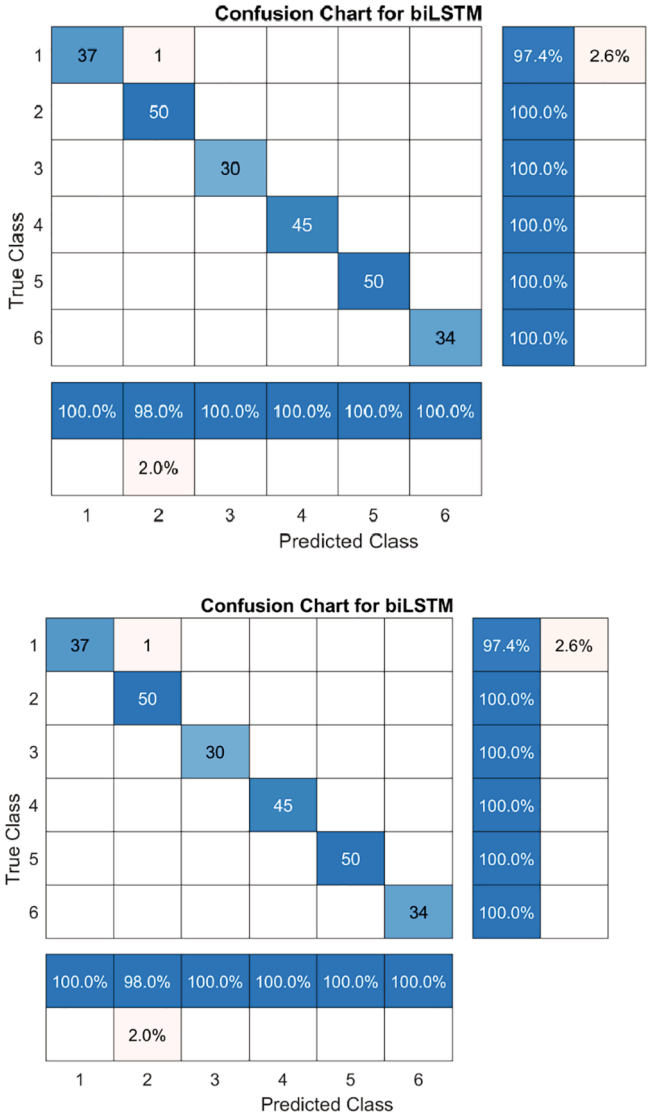


Fig. 7. The confusion matrix of the activity classification results using biLSTM network.

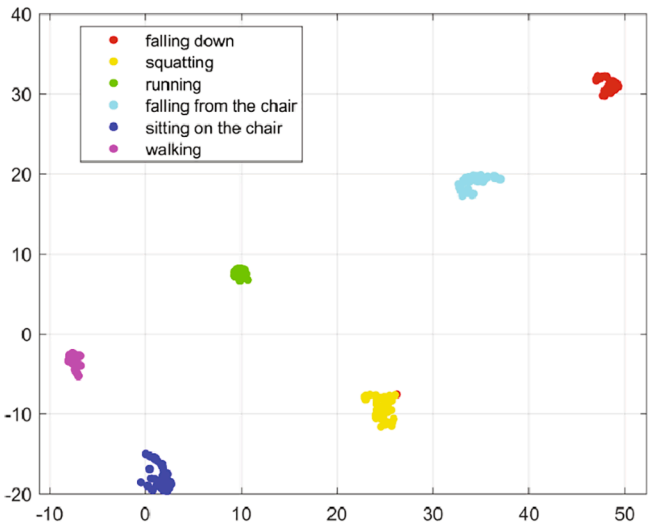


Fig. 8. Visualization of the activations of the final BiLSTM network layer.

features were found to be from the acceleration sensor data and include the variance of power (squatting, running, sitting) and minimum value of acceleration signal (falling down, falling from the chair, walking).

We also calculated the mean ranks of the features and evaluated their statistical significance using the Nemenyi post hoc test. The results are presented in the critical difference diagram in Fig. 10. The difference between mean ranks is not significant if it is smaller than the critical difference (CD). The best ranks were achieved by the min, mean, and max power of the acceleration sensor data.

Fig. 11 shows the best features ranked according to the accuracy drop using the Permutation Feature Importance (PFI) method. The best feature identified is the entropy of the gyroscope sensor data on the Y-axis.

4.4. Analysis of the data augmentation results

To analyze the influence of the data augmentation method on the accuracy of the classification results using the BiLSTM, we performed an experiment with different training:testing dataset ratios. We trained one BiLSTM model on the training data without data augmentation, while the other BiLSTM model was trained on the training data to which the proposed data augmentation procedure was applied. To calculate the mean accuracy and standard deviation, the network training and testing process was repeated 40 times.

The statistical significance between the performance of the classifiers was evaluated using the Student's *t*-test. The results are presented in Table 7. The results show that in case there is only a small dataset available, the proposed data augmentation procedure can significantly increase the performance of the classifier on the testing dataset, thus allowing to deal with the "small data" problem [33] while avoiding performance decrease due to overfitting. In other cases, the increase is not statistically significant, although the accuracy improvement is also achieved.

5. Limitations

Against the promising results obtained, the proposed method has some limitations as well. The first limitation is the energy requirements of the smartwatch used for activity recognition. We observe that constantly reading sensory data and transferring them via wireless (Bluetooth or Wi-Fi) considerably diminish the watch battery. To minimize energy consumption, we reduce the sensor reading sampling rate to 50 Hz. We also aggregate the data and transfer them periodically. We observed that the selected watch can function for a limited time with the implemented mobile application.

Another limitation of the proposed method is that the use of a cloud server which can present some risks to the proposed system. For example, communication between the smartwatch and the cloud server can be broken or interrupted. To handle such problems, a mobile

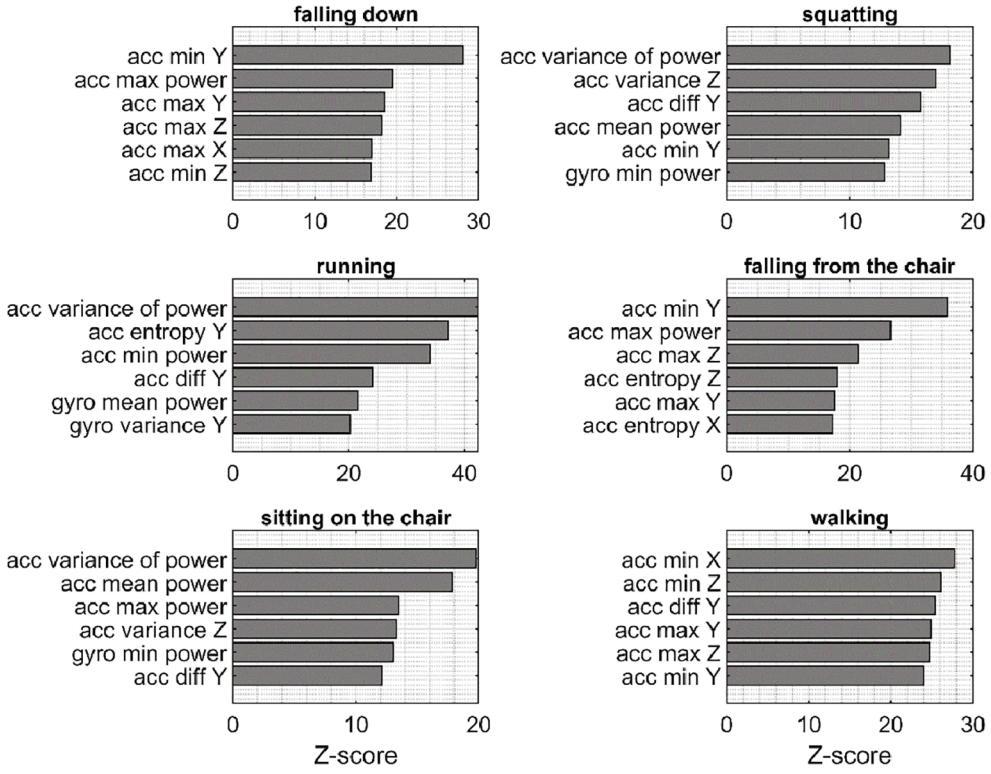


Fig. 9. Results of sensor data feature ranking using *t*-test. Here acc is acceleration sensor data, gyro is gyroscope sensor feature data.

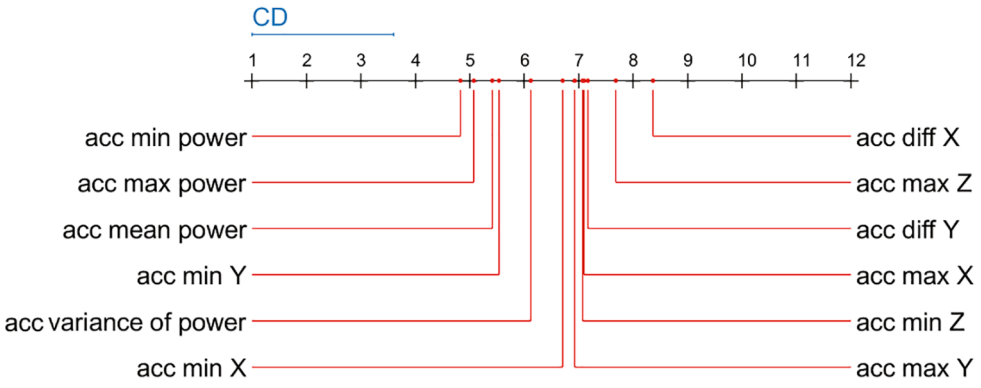


Fig. 10. Critical difference (CD) diagram of the sensor data features. Only 12 best features are shown. Acc is acceleration sensor data.

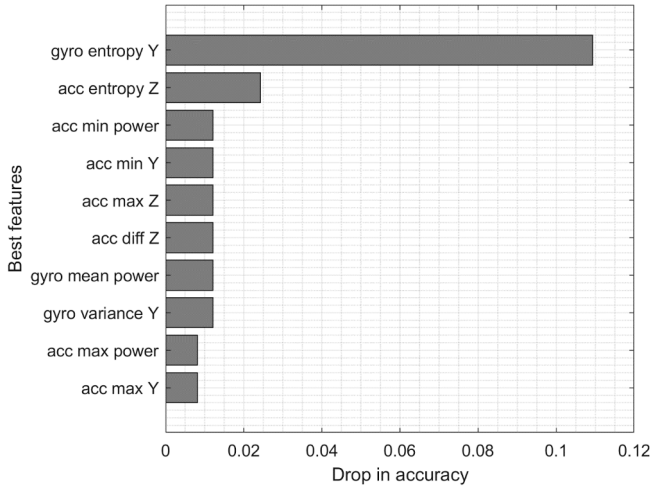


Fig. 11. Top-10 features based on Permutation Feature Importance (PFI). Here acc is acceleration sensor data, gyro is gyroscope sensor feature data.

Table 7
Sensitivity of the BiLSTM classifier performance to training:testing ratio.

Training / testing ratio (%)	Withoutdata augmentation		Withdata augmentation		Statistical significance
	Mean accuracy (%)	Standard deviation	Mean accuracy (%)	Standard deviation	
5:95	64.24	7.13	73.18	9.17	***
10:90	75.93	6.90	81.61	9.91	**
15:85	84.24	8.22	88.74	11.08	*
20:80	86.94	8.52	91.34	8.17	*
25:75	90.90	8.57	91.70	9.22	NS
30:70	94.36	5.84	97.10	8.15	NS
35:65	95.07	6.27	94.04	9.79	NS
40:60	96.34	4.47	98.13	5.11	NS
45:55	97.93	3.86	99.06	2.22	NS
50:50	97.77	4.60	98.42	3.16	NS
55:45	98.98	2.57	97.84	4.97	NS
60:40	97.42	6.52	98.91	2.94	NS

NS, not significant
*** $p < 0.001$;
** $p < 0.01$;
* $p < 0.05$;

application can be developed so that it can locally perform feature extraction and classification computations. However, the side effects of such approaches are increased processing, storage, and energy requirements. We may apply this solution by using a smartphone as an intermediate computing platform.

Finally, the composition of the subject groups used for the collection of the human activity data set may have been biased. We collected data from a student group so that they have similar characteristics, but older people may walk and fall quite differently than young aged university students. Thus, to create a system useful in real-life situations, we would have to collect data from subjects whose characteristics (especially, age) are different from each other.

6. Conclusion

In this paper, we present the results of the proposed smartwatch-based activity recognition system for fall detection. We use the accelerometer and gyroscope sensor data of a smartwatch to classify 6 different activities, namely falling while standing, falling from a chair, sitting, squatting, walking, and running. Our main results are as follows.

(1) Using a fusion accelerometer and gyroscope data features together for activity recognition gives better performance than single sensor data.

(2) Activity recognition using the BiLSTM model allowed to achieve 99.59% accuracy recognition of all activities (using leave-one-activity-out cross-validation), which is better than the accuracy achieved by the baseline KNN classifier. This demonstrated the superiority of the deep learning method over the classical machine learning method.

(3) Our results show that the proposed methodology also works well on a previously unseen subject as well: activity recognition using the BiLSTM deep learning model allowed to achieve 97.35% accuracy recognition of all activities (using leave-one-subject-out cross-validation).

(4) Power of acceleration sensor data and entropy of gyroscope sensor data were found to be the most significant features.

(5) The data augmentation scheme allowed to enhance the accuracy of classification on the testing dataset when only a small number of data samples are available for neural network training.

In future work, our goal is to reduce the number of features extracted from the data, which we believe will reduce the energy consumption of the smartwatch. To perform the feature selection, we plan to apply the nature-inspired heuristic optimization feature search.

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgments

This project is supported by Atilim University as an Undergraduate Research Project (ATÜ-LAP-C-1516-04). The student members of the project group are Alkım Kulakci, Derya İgde, Orhan Sarıbal, Canberk

Sepici, Adil Kutan Çingisiz, and Mehmet Serhat Özdal. We would like to extend our gratitude to all the project group student members for realizing this project.

Ethical statement

The research was performed in accordance with the principles of the Declaration of Helsinki Ethical approval for this research was obtained from the Department of Computer Engineering, Atilim University. The participants provided informed consent to participate in this study by signing a consent form following reading the information sheet provided for them.

Credit Author Statement

The authors have contributed equally to this work.

References

[1] S. Abbate M. Avvenuti P. Corsini J. Light A. Vecchio Y.K. Tan Wireless Sensor Networks: Application-Centric Design 2010 InTech 10.5772/13802.

[2] F. Afza, M.A. Khan, M. Sharif, S. Kadry, G. Manogaran, T. Saba, I. Ashraf, R. Damaševičius, A framework of human action recognition using length control features fusion and weighted entropy-variances based feature selection, Image and Vision Computing 106 (2021) 104090, https://doi.org/10.1016/j.imavis.2020.104090.

[3] A.A. Aguileta, R.F. Brena, O. Mayora, E. Molino-Minero-re, L.A. Trejo, Multi-sensor fusion for activity recognition—a survey, Sensors 19 (17) (2019) 3808, https://doi.org/10.3390/s19173808.

[4] R.O. Alabi, M. Elmusrati, I. Sawazaki-Calone, L.P. Kowalski, C. Haglund, R. D. Coletta, A.A. Mäkitie, T. Salo, A. Almangush, I. Leivo, Comparison of supervised machine learning classification techniques in prediction of locoregional recurrences in early oral tongue cancer, International Journal of Medical Informatics 136 (2020) 104068, https://doi.org/10.1016/j.ijmedinf.2019.104068.

[5] Aphairaj, D., Kitsonti, M., Thanapornsawan, T. (2019). Fall detection system with 3-axis accelerometer. In Journal of Physics: Conference Series (Vol. 1380, No. 1, p. 012060).

[6] Asif, U., Von Cavallar, S., Tang, J., Harrer, S. (2020). SSHFD: Single Shot Human Fall Detection with Occluded Joints Resilience. 24th European Conference on Artificial Intelligence, ECAI 2020, 29 August-8 September 2020, Santiago de Compostela, Spain. Frontiers in Artificial Intelligence and Applications 325, IOS Press 2020, pp. 2656-2663.

[7] O. Aziz, J. Klenk, L. Schwickert, L. Chiari, C. Becker, E.J. Park, G. Mori, S. N. Robinovitch, Y.-K. Jan, Validation of accuracy of SVM-based fall detection system using real-world fall and non-fall datasets, PLoS one 12 (7) (2017) e0180318, https://doi.org/10.1371/journal.pone.018031810.1371/journal.pone.0180318.g00110.1371/journal.pone.0180318.g00210.1371/journal.pone.0180318.t00110.1371/journal.pone.0180318.t00210.1371/journal.pone.0180318.t003.

[8] A. Baldominos, A. Cervantes, Y. Saez, P. Isasi, A comparison of machine learning and deep learning techniques for activity recognition using mobile devices, Sensors 19 (3) (2019) 521, https://doi.org/10.3390/s19030521.

[9] D. Ballabio, R. Todeschini, V. Consonni, in: Recent Advances in High-Level Fusion Methods to Classify Multiple Analytical Chemical Data, Elsevier, 2019, pp. 129–155, https://doi.org/10.1016/b978-0-444-63984-4.00005-3.

[10] S. Balli, E.A. Sağbaş, M. Peker, Human activity recognition from smart watch sensor data using a hybrid of principal component analysis and random forest algorithm, Measurement and Control 52 (1–2) (2019) 37–45, https://doi.org/10.1177/0020294018813692.

[11] D.R. Beddiar, B. Nini, M. Sabokrou, A. Hadid, Vision-based human activity recognition: a survey, Multimed Tools Appl 79 (41-42) (2020) 30509–30555, https://doi.org/10.1007/s11042-020-09004-3.

[12] V. Bianchi, M. Bassoli, G. Lombardo, P. Fornacciari, M. Mordonini, I. De Munari, IoT Wearable Sensor and Deep Learning: An Integrated Approach for Personalized Human Activity Recognition in a Smart Home Environment, IEEE Internet of Things Journal 6 (5) (2019) 8553–8562, https://doi.org/10.1109/JIoT.648890710.1109/JIOT.2019.2920283.

[13] A.M. Bica, Fitting data using optimal Hermite type cubic interpolating splines, Applied Mathematics Letters 25 (12) (2012) 2047–2051, https://doi.org/10.1016/j.aml.2012.04.016.

[14] H. Bragança, J.G. Colonna, W.S. Lima, E. Souto, A Smartphone Lightweight Method for Human Activity Recognition Based on Information Theory, Sensors 20 (7) (2020) 1856, https://doi.org/10.3390/s20071856.

[15] E. Casilari, M. Álvarez-Marco, F. García-Lagos, A Study of the use of gyroscope measurements in wearable fall detection systems, Symmetry 12 (4) (2020) 649, https://doi.org/10.3390/sym12040649.

[16] A. Chelli, M. Patzold, A Machine Learning Approach for Fall Detection and Daily Living Activity Recognition, IEEE Access 7 (8672567) (2019) 38670–38687, https://doi.org/10.1109/ACCESS.2019.2906693.

[17] L. Chen, R. Li, H. Zhang, L. Tian, N. Chen, Intelligent fall detection method based on accelerometer data from a wrist-worn smart watch, Measurement: Journal of the International Measurement Confederation 140 (2019) 215–226, https://doi.org/10.1016/j.measurement.2019.03.079.

[18] T.T. Dang, H. Truong, T.K. Dang, Automatic fall detection using smartphone acceleration sensor, International Journal of Advanced Computer Science and Applications 7 (12) (2016).

[19] F. Demrozi, G. Pravadelli, A. Bihorac, P. Rashidi, Human Activity Recognition using Inertial, Physiological and Environmental Sensors: A Comprehensive Survey, IEEE access : practical innovations, open solutions 8 (2020) 210816–210836, https://doi.org/10.1109/Access.628763910.1109/ACCESS.2020.3037715.

[20] B. Fu, N. Damer, F. Kirchbuchner, A. Kuijper, Sensing technology for human activity recognition: A comprehensive survey, IEEE Access 8 (9083980) (2020) 83791–83820, https://doi.org/10.1109/ACCESS.2020.2991891.

[21] S.K. Gharghan, S.L. Mohammed, A. Al-Naji, M.J. Abu-AlShaeer, H.M. Jawad, A. M. Jawad, J. Chahl, Accurate fall detection and localization for elderly people based on neural network and energy-efficient wireless sensor network, Energies 11 (11) (2018) 2866.

[22] D. Gholamiangonabadi, N. Kiselov, K. Grolinger, Deep Neural Networks for Human Activity Recognition With Wearable Sensors: Leave-One-Subject-Out Cross-Validation for Model Selection, IEEE Access 8 (2020) 133982–133994, https://doi.org/10.1109/Access.628763910.1109/ACCESS.2020.3010715.

[23] M. Gjoreski, V. Janko, G. Slapničar, M. Mlakar, N. Reščič, J. Bizjak, V. Drobnič, M. Marinko, N. Mlakar, M. Luštrek, M. Gams, Classical and deep learning methods for recognizing human activities and modes of transportation with smartphone sensors, Information Fusion 62 (2020) 47–62, https://doi.org/10.1016/j.inffus.2020.04.004.

[24] J. Guo, X. Zhou, Y. Sun, G. Ping, G. Zhao, Z. Li, Smartphone-Based Patients' activity recognition by using a self-learning scheme for medical monitoring, Journal of medical systems 40 (6) (2016) 140.

[25] M. Hagui, M. Mahjoub, F. ElAyeb, A new Framework for Elderly Fall Detection Using Coupled Hidden Markov Models, The International Arab Journal Of Information Technology 16 (4) (2019) 775–783.

[26] F. Harrou, N. Zerrouki, Y. Sun, A. Houacine, Vision-based fall detection system for improving safety of elderly people, IEEE Instrumentation Measurement Magazine 20 (6) (2017) 49–55.

[27] A.M. Helmi, M.A.A. Al-qaness, A. Dahou, R. Damaševičius, T. Krilavičius, M. A. Elaziz, A novel hybrid gradient-based optimizer and grey wolf optimizer feature selection method for human activity recognition using smartphone sensors, Entropy 23 (8) (2021) 1065, https://doi.org/10.3390/e23081065.

[28] F. Hussain, F. Hussain, M. Ehatisham-ul-Haq, M.A. Azam, Activity-Aware Fall Detection and Recognition Based on Wearable Sensors, IEEE Sensors Journal 19 (12) (2019) 4528–4536, https://doi.org/10.1109/JSEN.736110.1109/JSEN.2019.2898891.

[29] Z. Hussain M. Sheng W.E. Zhang Different Approaches for Human Activity Recognition: A Survey. arXiv preprint arXiv:1906.05074 2019.

[30] C.Y. Jeong, M. Kim, An energy-efficient method for human activity recognition with segment-level change detection and deep learning, Sensors 19 (17) (2019) 3688, https://doi.org/10.3390/s19173688.

[31] Kingma, D.P.; Ba, J. Adam: A Method for Stochastic Optimization. In 3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015.

[32] S. Kiran, M. Attique Khan, M. Younus Javed, M. Alhaisoni, U. Tariq, Y. Nam, R. Damaševičius, M. Sharif, Multi-Layered Deep Learning Features Fusion for Human Action Recognition, Computers, Materials & Continua 69 (3) (2021) 4061–4075, https://doi.org/10.32604/cmc.2021.017800.

[33] R. Kitchin, T.P. Lauriault, Small data in the era of big data, GeoJournal 80 (4) (2014) 463–475, https://doi.org/10.1007/s10708-014-9601-7.

[34] Y. Lee H. Yeh K.-H. Kim O. Choi A real-time fall detection system based on the acceleration sensor of smartphone International Journal of Engineering Business Management 10 2018 184797901775066 10.1177/1847979017750669.

[35] A. Lentzas, D. Vrakas, Non-intrusive human activity recognition and abnormal behavior detection on elderly people: a review, Artificial Intelligence Review 53 (3) (2020) 1975–2021, https://doi.org/10.1007/s10462-019-09724-5.

[36] W.S. Lima, E. Souto, K. El-Khatib, R. Jalali, J. Gama, Human activity recognition using inertial sensors in a smartphone: An overview, Sensors 19 (14) (2019) 3213, https://doi.org/10.3390/s19143213.

[37] N.a. Lu, Y. Wu, L.i. Feng, J. Song, Deep learning for fall detection: Three-dimensional CNN Combined with LSTM on video kinematic data, IEEE Journal of Biomedical and Health Informatics 23 (1) (2019) 314–323, https://doi.org/10.1109/JBHI.2018.2808281.

[38] M. Martinez, P.L. De Leon, Falls Risk Classification of Older Adults Using Deep Neural Networks and Transfer Learning, IEEE Journal of Biomedical and Health Informatics 24 (1) (2020) 144–150, https://doi.org/10.1109/JBHI.622102010.1109/JBHI.2019.2906499.

[39] F.M. Noori, M. Riegler, M.Z. Uddin, J. Torresen, Human Activity Recognition from Multiple Sensors Data Using Multi-fusion Representations and CNNs, ACM Transactions on Multimedia Computing, Communications, and Applications 16 (2) (2020) 1–19, https://doi.org/10.1145/3377882.

[40] N. Noury, A. Fleury, P. Rumeau, A.K. Bourke, G.O. Laighin, V. Rialle, J.E. Lundy, in: Fall detection-principles and methods, IEEE, 2007, pp. 1663–1666.

[41] H.F. Nweke, Y.W. Teh, G. Mujtaba, M.A. Al-garadi, Data fusion and multiple classifier systems for human activity detection and health monitoring: Review and open research directions, Information Fusion 46 (2019) 147–170, https://doi.org/10.1016/j.inffus.2018.06.002.

[42] A. Patel, J. Shah, Sensor-based activity recognition in the context of ambient assisted living systems: A review, Journal of Ambient Intelligence and Smart Environments 11 (4) (2019) 301–322, https://doi.org/10.3233/AIS-190529.
[43] D. Powers, Evaluation: From Precision, Recall and F-Measure to ROC, Informedness, Markedness Correlation, Journal of Machine Learning Technologies 2 (1) (2011) 37–63.
[44] S.J. Priya, A.J. Rani, M.S.P. Subathra, M.A. Mohammed, R. Damaševičius, N. Ubendran, Local pattern transformation based feature extraction for recognition of parkinson's disease based on gait signals, Diagnostics 11 (8) (2021) 1395, https://doi.org/10.3390/diagnostics11081395.
[45] L. Ren, Y. Peng, Research of fall detection and fall prevention technologies: A systematic review, IEEE Access 7 (2019) 77702–77722.
[46] P. Sarcevic, Z. Kincses, S. Pletl, Online human movement classification using wrist-worn wireless sensors, Journal of Ambient Intelligence and Humanized Computing 10 (1) (2019) 89–106, https://doi.org/10.1007/s12652-017-0606-1.
[47] M. Schuster, K.K. Paliwal, Bidirectional recurrent neural networks, IEEE Transactions on Signal Processing 45 (11) (1997) 2673–2681, https://doi.org/10.1109/78.650093.
[48] G. Şengül, E. Ozcelik, S. Misra, R. Damaševičius, R. Maskeliūnas, Fusion of smartphone sensor data for classification of daily user activities, Multimedia Tools and Applications (2021), https://doi.org/10.1007/s11042-021-11105-6.
[49] A. Sherstinsky, Fundamentals of Recurrent Neural Network (RNN) and Long Short-Term Memory (LSTM) network, Physica D: Nonlinear Phenomena 404 (2020) 132306, https://doi.org/10.1016/j.physd.2019.132306.
[50] A. Stisen, H. Blunck, S. Bhattacharya, T.S. Prentow, M.B. Kjærgaard, A. Dey, T. Sonne, M.M. Jensen, in: Smart devices are different: Assessing and mitigating mobile sensing heterogeneities for activity recognition, ACM, 2015, pp. 127–140.
[51] D. Thakur, S. Biswas, Smartphone based human activity monitoring and recognition using ML and DL: a comprehensive survey, Journal of Ambient Intelligence and Humanized Computing 11 (11) (2020) 5433–5444, https://doi.org/10.1007/s12652-020-01899-y.
[52] A. Tharwat, Classification assessment methods, Applied Computing and Informatics 17 (1) (2021) 168–192, https://doi.org/10.1016/j.aci.2018.08.003.
[53] Y. Tian, J. Zhang, J. Wang, Y. Geng, X. Wang, Robust human activity recognition using single accelerometer via wavelet energy spectrum features and ensemble feature selection, Systems Science Control Engineering 8 (1) (2020) 83–96.
[55] M.Z. Uddin, M.M. Hassan, Activity Recognition for Cognitive Assistance Using Body Sensors Data and Deep Convolutional Neural Network, IEEE Sensors Journal 19 (19) (2019) 8413–8419, https://doi.org/10.1109/JSEN.736110.1109/JSEN.2018.2871203.
[56] L.J.P. van der Maaten, G.E. Hinton, Visualizing Data Using t-SNE, Journal of Machine Learning Research. 9 (2008) 2579–2605.
[57] P. Van Thanh, D.-T. Tran, D.-C. Nguyen, N. Duc Anh, D. Nhu Dinh, S. El-Rabaie, K. Sandrasegaran, Development of a real-time, simple and high-accuracy fall detection system for elderly using 3-DOF accelerometers, Arabian Journal for Science and Engineering 44 (4) (2019) 3329–3342.
[58] R.-A. Voicu, C. Dobre, L. Bajenaru, R.-I. Ciobanu, Human physical activity recognition using smartphone sensors, Sensors 19 (3) (2019) 458, https://doi.org/10.3390/s19030458.
[59] J. Wang, Y. Chen, S. Hao, X. Peng, L. Hu, Deep learning for sensor-based activity recognition: A survey, Pattern Recognition Letters 119 (2019) 3–11, https://doi.org/10.1016/j.patrec.2018.02.010.
[60] S. Wang, X. Wang, S. Wang, D. Wang, Bi-directional long short-term memory method based on attention mechanism and rolling update for short-term load forecasting, International Journal of Electrical Power Energy Systems 109 (2019) 470–479, https://doi.org/10.1016/j.ijepes.2019.02.022.
[61] G.M. Weiss, J.L. Timko, C.M. Gallagher, K. Yoneda, A.J. Schreiber, in: Smartwatch-based activity recognition: A machine learning approach, IEEE, 2016, pp. 426–429.
[62] D. Yacchirema, J.S. de Puga, C. Palau, M. Esteve, Fall detection system for elderly people using IoT and big data, Procedia computer science 130 (2018) 603–610.
[63] H. Yao M. Yang T. Chen Y. Wei Y.u. Zhang Depth-based human activity recognition via multi-level fused features and fast broad learning system International Journal of Distributed Sensor Networks 16 2 2020 155014772090783 10.1177/1550147720907830.
[64] N. Zhou, L. Wang, A modified T-test feature selection method and its application on the HapMap genotype data, Genomics, proteomics bioinformatics 5 (3-4) (2007) 242–249.
[65] R. Zhu, Z. Xiao, Y. Li, M. Yang, Y. Tan, L. Zhou, S. Lin, H. Wen, Efficient Human Activity Recognition Solving the Confusing Activities Via Deep Ensemble Learning, IEEE Access 7 (8734079) (2019) 75490–75499, https://doi.org/10.1109/ACCESS.2019.2922104.
[66] J. Demšar, Statistical comparisons of classifiers over multiple data sets. J. Mach. Learn. Res. 7 (2006) 1–30.