Computer Systems Science & Engineering
DOI: 10.32604/csse.2023.032931
Article

Deep Forest-Based Fall Detection in Internet of Medical Things Environment

Mohamed Esmail Karar[1,2,*], Omar Reyad[1,3] and Hazem Ibrahim Shehata[1,4]

[1]College of Computing and Information Technology, Shaqra University, P.O. Box 33, Shaqra, 11961, Saudi Arabia
[2]Department of Industrial Electronics and Control Engineering, Faculty of Electronic Engineering, Menoufia University, Menouf, 32952, Egypt
[3]Computer Science Department, Faculty of Computers and Artificial Intelligence, Sohag University, Sohag, 82524, Egypt
[4]Department of Computer and Systems Engineering, Faculty of Engineering, Zagazig University, Zagazig, 44519, Egypt
*Corresponding Author: Mohamed Esmail Karar. Email: mkarar@su.edu.sa
Received: 02 June 2022; Accepted: 05 July 2022

Abstract: This article introduces a new medical internet of things (IoT) framework for intelligent fall detection system of senior people based on our proposed deep forest model. The cascade multi-layer structure of deep forest classifier allows to generate new features at each level with minimal hyperparameters compared to deep neural networks. Moreover, the optimal number of the deep forest layers is automatically estimated based on the early stopping criteria of validation accuracy value at each generated layer. The suggested forest classifier was successfully tested and evaluated using a public SmartFall dataset, which is acquired from three-axis accelerometer in a smartwatch. It includes 92781 training samples and 91025 testing samples with two labeled classes, namely non-fall and fall. Classification results of our deep forest classifier demonstrated a superior performance with the best accuracy score of 98.0% compared to three machine learning models, i.e., K-nearest neighbors, decision trees and traditional random forest, and two deep learning models, which are dense neural networks and convolutional neural networks. By considering security and privacy aspects in the future work, our proposed medical IoT framework for fall detection of old people is valid for real-time healthcare application deployment.

Keywords: Elderly population; fall detection; wireless sensor networks; internet of medical things; deep forest

1 Introduction

World Health Organization (WHO) defines the accident of a fall as an event that occurs when the persons unintentionally come to rest on the ground, floor, or any lower level [1]. Falls pose a threat to the health of older adults and can limit their ability to stay independent. Moreover, some of the injuries caused by falls might be deadly to an elderly person. According to WHO, compared to other age groups, adults over the age of 60 account for the majority of fatal falls (with an estimated total of 684k fatal falls occurring each year). Without a doubt, some of these lives could have been saved if the fall of the elderly person was timely detected by the caregiving/medical personal. This was one of the main motivations for the emergence of many of the fall detection technologies that we witness today.

An important component of a fall detection system is the sensor(s) that collect(s) the physiological and/or ambient data from the parson being monitored. Fall detection sensors are classified by Mozaffari et al. [2] into three categories: motion sensors, physiological sensors, and ambient (or environmental) sensors. The most used motion sensors are the accelerometers (which measure rate of change in velocity with respect to time), the gyroscopes (which detect angular velocity in 3-D), and the magnetometers (which sense orientation). The physiological sensors track vital body indicators such as blood pressure, blood oxygen level and body temperature. The ambient sensors scan the environment surrounding the elderly patient. Vision and sound sensors (like surveillance cameras and microphones) are the most common examples of the ambient sensors category.

Fall detection systems employ various methods to interpret the data collected from the sensors and look for certain patterns that characterize fall events. Wang et al. [3] classify fall detection methods into two main categories: threshold-based and data-driven. Threshold-based methods are usually used when the data produced by the sensor has a simple form like the case of accelerometers and gyroscopes [4,5]. In this case, a fall is detected whenever the sensor readings exceed an empirical threshold value. Threshold-based methods are lightweight and require low computational resources. However, the outcome of these methods is prone to variability when applied to different groups of patients.

Data-driven methods for fall detection harness the power of machine learning (ML) and deep learning (DL) to analyze the collected data and recognize fall events. Unlike threshold-based methods, data-driven methods can deal with complex forms of data (such as sound and video signals) and handle multiple hybrid sensors (i.e., sensor fusion). The most common algorithms used in data-driven methods for fall detection are: Hidden Markov Model (HMM) [6], Support Vector Machine (SVM) [7], Decision Trees [8] and Convolutional Neural Networks (CNN) [9,10]. Data-driven methods are potentially more accurate than threshold-based methods yet require more computational resources.

The computational resources needed for fall detection can be provided (by cloud computing) using internet-of-things (IOT) technologies. Many IOT-based fall detection systems for the elderly have been proposed in recent years [11–14]. These IOT-based systems are potentially more scalable than regular architectures and hence can be quite helpful if the healthcare staff is not enough to provide the required assistance compared to the number of patients. Moreover, the architecture of IOT-based systems can be flexible such that some computations can be pushed from the cloud to the edge to reduce data transmission overhead time and provide faster response.

In this article, we propose a new intelligent IoT-based fall detection system to support indoor healthcare of old patients. The contributions of this study are summarized as follows.

- Building a simple and advanced fall detection framework for senior people based on wearable smartwatches and smart internet of medical things (IoMT).
- Developing a new deep forest classifier to identify fall accidents of elderly patients instead of using complex deep neural network models with high computational resources.
- Validating the outperformance of our developed deep forest classifier *vs.* other machine learning and deep learning classifiers in previous studies by conducting a comparative study.

The remainder of this paper is organized as follows. Section 2 presents an overview of relevant research works of fall detection systems including types of sensors, machine and deep learning methods and current fall detection systems. Section 3 gives a full description of our proposed IoT-based framework for fall detection of elderly patients using our developed deep forest classifier. Overall results and evaluations of our extensive experiments and discussions are demonstrated in Sections 4 and 5, respectively. Concluding remarks and main prospects of this study are finally given in Section 6.

2 Related Work

Over the last few years, there has been a significant advance of new technology designed by researchers to detect and prevent falls in Elderly Patients. A scalable design for the Whoops system is presented in [11]. The suggested architecture can monitor thousands of elderly people's cases, detect falls, and alert caregivers. In addition, numerous ML algorithms for evaluating the detection system's applicability have been validated. Boosted decision trees had the highest classification performance of all the models examined. In [13], Yacchirema et al. proposed a new IoT-based system for identifying senior people's falls in indoor situations. Wireless sensor networks, smart devices, and cloud computing platform are all integrated into this solution. Each time whenever a fall is discovered, the system uses data from previous falls to develop a new ML model. The suggested method has a high-level ratio in the recognition of falling, as evidenced by its identification accuracy of 92%.

Moreover, a variety of intelligent solutions of ML and DL algorithms, IoT devices, and imaging techniques have been proposed in previous studies to deal with the problem of elderly people falling [15,16]. Intelligent fall detection methods can employ vision, sound, or a wearable device to detect falls. The accelerometer, gyroscopes, or a combination of sensing devices are widely used in wearable fall detection sensors. Furthermore, multiple wearable sensors have been formed in the medical IoT context to detect and avoid falls in elderly patients at home. These devices are mostly monitoring and alarm systems that are meant to prevent, detect, and warn caregivers in the event of a fall. People over the age of 65 accounts for more than 50 percent of all injury-related hospitalizations. As a result, countries are funding more in fall detection device development to reduce the amount of money spent on medical care and treatment for post-fall injuries.

An integrated Histogram of Oriented Gradients (HOG) with SVM have been used for fall detection using IoT scheme [17]. For digital images of older adults, a smart wearable sensor is used instead of an Red-Green-Blue (RGB) camera to provide privacy and strong light intensity modifications. The attributes of an individual are recovered using the HOG algorithm after the acquisition of denoised binary images. Then, they are classified to identify the fall circumstances using linear SVM.

The authors in [18] used an edge-computing architecture to identify daily patient activities including fall events based on a long short-term memory (LSTM) model in real-time. The edge computing platform can detect falls with a 95.8% accuracy using real-time flowing data processing. A centralized IoT-based fall detection was proposed for real-time surveillance of a large population [19]. Many types of customized embedded devices can be utilized to monitor a massive population, such as Raspberry Pi and Arduino boards. Finally, the proposed detection method achieved accuracy score of 99.7%. The authors of [20] presented two temporal inference models, classification model I and classification model II, to detect fall incidents by utilizing wearable IoT altimeter sensors. The results were promising based on such inference models for indoor fall monitoring of senior people, with the highest prediction accuracy score of 98% for the proposed classification model II.

The authors in [21] described an approach for preventing falls in the elderly by designing and deploying a fall monitoring system that uses machine learning to make decisions and the IoT to store data and broadcast alerts. Extreme Gradient Boosting (XGBoost), a machine learning technique noted for its accuracy and speed, was employed and it has a 96% accuracy rate. An IoT-enabled elderly fall detection algorithm is presented in [22] for smart homecare that uses an optimum deep convolutional neural network (IMEFD-ODCNN). The IMEFD-ODCNN model's purpose is to make it possible for smartphones and sophisticated deep learning (DL) algorithms to recognize falls in the smart home. The results showed that the IMEFD-ODCNN model outperformed other current techniques, with an accuracy of 99.76% maximum and 99.57% on the multiple cameras fall down and University of Rzeszow (UR) fall detection datasets, respectively. In [23], the authors presented a noise tolerant Fall Detection System (FDS) that performs well in the event of missing values in data. To construct FDS based on wearable sensors, the study used Recurrent Neural Networks (RNNs) with an underlying Bidirectional Long Short-Term Memory (BiLSTM) structure. The results indicate that BiLSTM's capacity to preserve long-term interconnections makes it a suitable model for dealing with missing values in wearable fall detection systems. An IoT-based solution for human falls in people's homes is proposed in [24]. Edge, fog, and cloud make up the solution's three-layered cognitive architecture. Human falls are classified using a mathematical model based on the Morlet wavelet and artificial neural networks (ANNs) model. The results showed that combining both models is efficient and practical to the developed system, with 92.5% accuracy. To identify fall/non-fall events, the research in [25] provided an Improving Archimedes Optimization Approach with Deep Learning Empowered Fall Detection (IAOA-DLFD) model. The IAOA leverages excellent Capsule Network (CapsNet) hyperparameter selection to dramatically improve overall fall detection performance. The test images' suitable class labels are also determined using a radial basis function (RBF) network. The improved IAOA-DLFD technique had a 99.7% accuracy rate, according to the obtained results. Tab. 1 shows a comparison of recent fall detection methods based on the utilized techniques.

Table 1: A comparison of different fall detection methods in the literature

Method	Types of sensors	Dataset	Advantages	Disadvantages
Boosted decision tree [11]	Accelerometer & gyroscope	SisFall	Size of stored and transmitted data IS small	Unnecessary emergency actions
Big data machine learning (BigML) [13]	Accelerometer	SisFall	High success rate in fall detection	Error rate is 33%
Histogram of oriented gradients with support vector machine [17]	Deep sensor	Mixed dataset	Accuracy improves to 98.1%	Detection errors still occur
Long short-term memory (LSTM) [18]	MetaMotionR wearable sensor	MobiAct	Real-time streaming with an accuracy of 95.8%.	Data reduction
Linear classifier [19]	Accelerometer	tFall	The response time is quickly	Requires consistent network connectivity
Temporal inference [20]	Wearable altimeter sensor	Synthetic YouTube videos	Early warning of a fall incident	Methodological limitation
Extreme gradient boosting (XGBoost) [21]	Accelerometer & gyroscope	SisFall	Increase the accuracy and reduce false alarms	Data acquired from falls including young individuals
Squeeze-and-excitation network based classifier [22]	Cameras	Fall dataset of multiple cameras	Maximum accuracy of 99.76%	Requires scalable and solid versions of the proposed design

(Continued)

Table 1 (continued)

Method	Types of sensors	Dataset	Advantages	Disadvantages
Recurrent neural network with bidirectional LSTM [23]	Accelerometer & gyroscope	SisFall & UP-Fall detection	Handle missing values in data	Decrease in correct predictions
Morlet wavelet & artificial intelligence [24]	Accelerometer doppler sensor	Raw data created	Accuracy of 92.5% without false negatives	Require higher computational effort
Capsule network (CapsNet) & radial basis function [25]	Accelerometer	Fall dataset of multiple cameras	Maximum accuracy of 99.7%	Unimodal fusion model

3 Dataset and Methods

3.1 Fall Dataset

A public fall dataset from Texas State University [26,27] has been used in this study. This dataset was acquired from 14 healthy individuals by using a wearable smartwatch. They are between 21 to 60 years old, and their heights and weights range from 1.52 to 1.98 m and 45 to 104 Kg, respectively. The volunteered individuals simulated fall situations to give two dataset classes of fall and non-fall cases only. The original dataset has measurements of the embedded accelerometer in 3 directions of x, y, z. The outcomes are represented in a binary form, such that ones and zeros present fall and non-fall instances, respectively. Tab. 2 illustrates a total of 183806 samples of the whole fall dataset, which are divided into two main files for training and testing phases. Total training samples are 92781 (8175 samples for fall cases and 84606 samples for non-fall cases), while total testing samples are 91025, including 5025 fall instances and 86000 non-fall instances.

Table 2: Samples distribution of the fall dataset in this study

Data	Fall	Non-fall	Total
Training samples	8175	84606	92781
Testing samples	5025	86000	91025
Total	13200	170606	183806

3.2 Traditional Random Forest

Random forests are firstly introduced by Breiman in 2001 [28]. Basically, the random forest (RF) is a combination of multiple decision trees to form an ensemble learning algorithm, as depicted in Fig. 1. For one decision tree, a bootstrap technique is applied to extract randomized samples from original dataset. A subspace of random features is utilized to choose a sorting point at all nodes of the decision tree. Then, the final prediction result of this group of decision trees is obtained based on a majority voting. Therefore, the RF is considered as one of the most powerful ML algorithms, including the following main steps [29]:

- Using bagging or bootstrap aggregating method, a random sampling of the training dataset with replacement is done to form an ensemble model of the decision trees.

- For each new sub-dataset created by sampling, the RF decision tree is built. Tree nodes are divided by randomly picking m features from the basic D features, such that m < D. A reduction in the information gain ratio is the selection criteria. The preceding procedure is continued until all of the tree nodes have been built and the halting requirements have been satisfied. There are fewer than a number of samples that has been predetermined in the leaf nodes, or all of the data in the sub-datasets can be distinguished.
- Each classifier of decision trees is used to generate its own prediction in order to establish the classification of new data classes, and the outcomes of each tree are then used to output the final distinct findings by majority vote. Fig. 1 depicts the basic decision flowchart of a simple RF model.

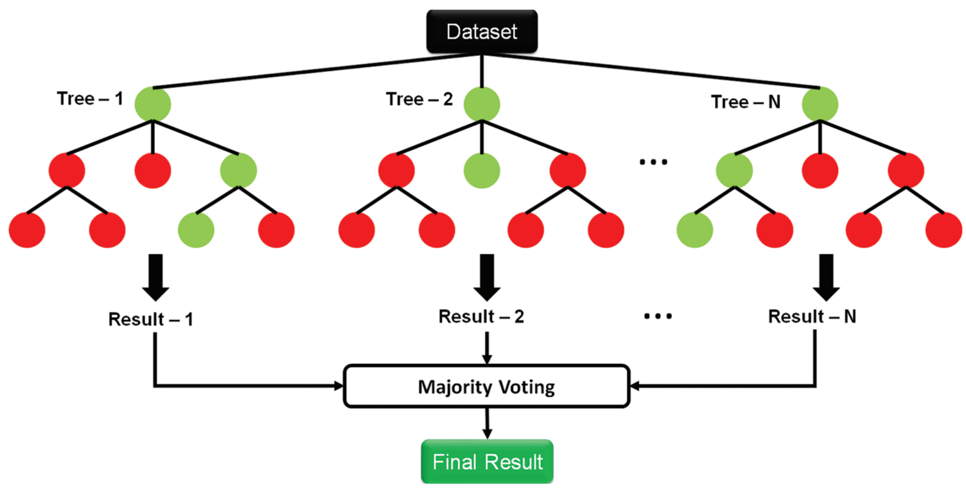

Figure 1: Schematic diagram of a simplified random decision forest

The classical RF algorithm as an ensemble method of decision trees showed good performance in real-world applications, e.g., efficient authentication for smartphone users [30] and detection of human voice pathology [31]. However, it is still depending only on the original feature representation during the training phase, without generating new features into the model itself [32]. That limited the traditional RF algorithm to achieve good generalization performance compared to recent deep neural networks.

3.3 Deep Forest Model

Deep forest is a decision-tree ensemble technique similar to deep networks but with smaller number of hyper-parameters [32]. It is developed in a cascade structure such that each level of the cascade receives feature information processed by the previous level, as depicted in Fig. 2. The deep forest levels are estimated based on the processed input data automatically. One level of the deep forest presents an ensemble of random forests. For instance, two random forests and two completely random forests [28] are included to enhance diversity, as shown in Fig. 2; where each completely random forest is composed of 500 complete decision trees [33], created by defining a random feature for dividing at every tree node. Then, a growing tree is stopped when each leaf node has only one class of all instances, i.e., Gini value (measures the impurity at the leaf node) = 0. In the same way, each random forest consists of 500 decision trees, where the number of trees is a hyperparameter for each forest. The number of input features d is randomly chosen as $\sqrt{d}$ feature candidates and selected the one which achieved the best Gini value for splitting [32].

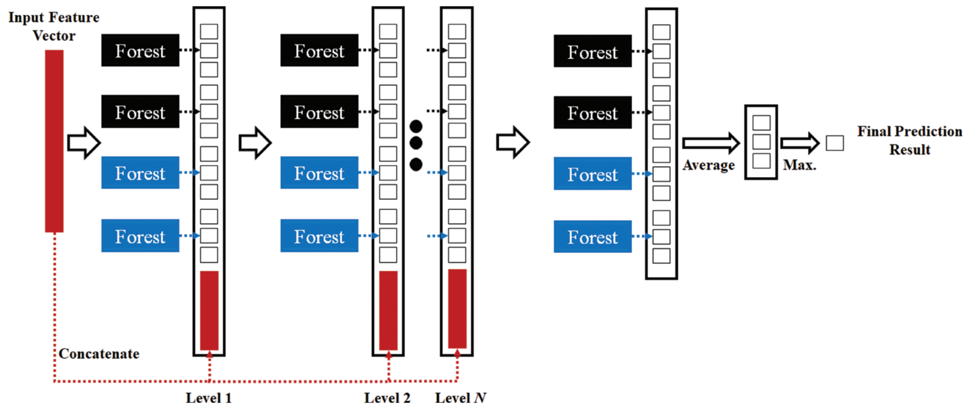

Figure 2: Main structure of deep forest [32]. Assuming three predicted classes and each level is composed of two random forests (*black*) and two completely random forests (*blue*). Each random forest generates a concatenated 3D class vector for feeding the input of the next cascade levels

Each RF can calculate class distribution for a given instance by estimating the percentage of different classes of training samples at the leaf node where the instance is located, and then averaging over all decision trees in the same RF. Each class vector is formed based on the estimated class distribution. Then, it is concatenated with the original feature vector to be the input for the next level of the deep forest. For instance, each one of four RFs will generate a 3D class vector for three classes. Hence, the input of the next level receives augmented features of 12 ($= 3 \times 4$), as presented in Fig. 2.

To avoid overfitting of overall deep forest model, k-fold cross validation [34,35] has been applied to generate the resulted class vector for each random forest. That means each instance will be utilized as $k - 1$ times of the training dataset, yielding $k - 1$ class vectors, which will be averaged to generate the final class vector as augmented features for the next stage of the cascade forest levels. The overall performance of the deep forest model may be assessed on the validation set after extending a new level, and the training operation will stop if no significant performance is detected. Therefore, the number of deep forest levels can be estimated automatically. In case of the computational cost of training phase and/or limited computing resources are available, the training error rather than cross-validation error can be employed to restrict the growth of forest levels. That allows adaptive model complexity of the deep forest to be constructed at different scales of training data [32].

3.4 Proposed Fall Detection System

Fig. 3 shows our proposed smart fall detection system for senior people with smartwatches based on a medical IoT framework. In this study, accelerometer sensing device is assumed to be one of the built-in sensors in smartwatches to give time-series readings of patient conditions, i.e., fall or non-fall, in 3-dimensional cartesian coordinates. The workflow of our smart IoT-based fall detection system can be described in three main steps as follows.

First, the old patient wears a smartwatch integrated with a built-in three-axis accelerometer to detect any unexpected change in the body motions along the day, as presented in previous fall detection systems [36]. These measurements of the user's movements are recorded and sent to medical cloud services for further analysis via a wireless home network, e.g., Wi-Fi or Bluetooth networks, connected to the Internet.

Second, medical cloud services facilitate recording and analyzing different types of patient data like signals and images, without *in-situ* high computing resources at medical centers and hospitals [37]. There, the acquired readings of the patient movements are classified using fine-tuned deep forest model, as introduced above in Fig. 2.

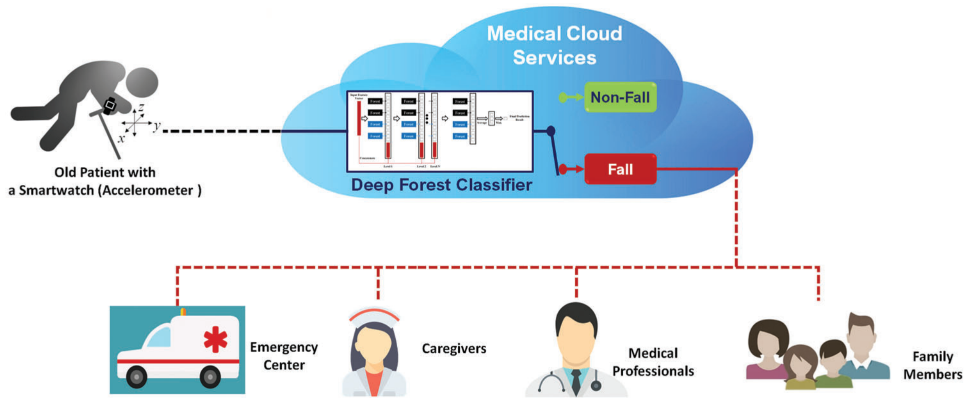

Figure 3: Proposed deep forest classifier for identifying fall accidents of old patients with a built-in three-axis accelerometer of smartwatches in a medical IoT framework

Third, two output modes of the deep forest classifier are defined as non-fall (safe mode) and fall (emergency mode). In non-fall mode, the physical activity of patients is normal, and no further actions are needed. But the emergency mode is immediately activated if a fall accident is detected. Then, emergency response agents, i.e., emergency center, caregivers, medical professionals and family members are notified with automatic phone calls or messages for helping patients and giving them urgent medical care, as shown in Fig. 3.

3.5 Fall Classification Evaluation Metrics

The following evaluation metrics for fall detection have been used to assess our proposed deep forest classifier. A confusion matrix of 2×2 was created using cross-validation estimation [38]. True Positive (TP), True Negative (TN), False Positive (FP), and False Negative (FN) are the four predicted outcomes of non-fall and fall classes. Additionaally, four well-known classification metrics, namely accuracy, precision, recall (sensitivity), and F1-measure have been calculated, as given in Eqs. (1)–(4).

$$accuracy = \frac{TP + TN}{TP + FP + FN + TN} 100\% \tag{1}$$

$$precision = \frac{TP}{TP + FP} \tag{2}$$

$$recall(sensitivity) = \frac{TP}{TP + FN} \tag{3}$$

$$F1 - measure = \frac{2(precision \times recall)}{precision + recall} \tag{4}$$

4 Experimental Results and Evaluation

The Tensorflow 2 and Keras packages were used to implement the suggested deep forest classifier and additional machine learning models [39]. The classification tests were run on a laptop with 16 GB of RAM and an Intel(R) Core (TM) i7-2.2 GHz processor. A graphical processing unit (GPU) NVIDIA of 4 GB is also exploited for executing all experiments. Two files of the public SmartFall dataset [26,27] including total samples of 92781 and 91025 are used for training and testing phases, respectively. For the testing file, the number of non-fall cases (86000 samples) is approximately 17 times higher than the fall cases

(5025 samples), as illustrated in Tab. 2. The original values of accelerometer coordinates (x, y, z) in the used dataset have been directly applied as inputs for all tested classifiers without any scaling or modification.

Besides our proposed deep forest classifier, three machine learning models, namely, K nearest neighbors (KNN), decision tree and traditional RF, and two deep learning models, namely, deep dense network and 1-D convolutional network (CNN) have been implemented and tested to identify fall cases. That allows to verify the advantageous performance of applied deep forest model. The number of trees in traditional RF and deep forest is carefully set to 10. To avoid the overfitting conditions, early stopping of the model training is activated based on automatic estimation of the validation accuracy for each generated layer, as illustrated in Tab. 3. The final optimal number of layers of our deep forest classifier is 3.

Table 3: Automatic layer estimation of our cascade forest model

Layer number	Validation accuracy	Early stopping (max = 2)
0	94.30	-
1	94.40	-
2	94.41	-
3	94.39	**Detected 1 of 2**
4	94.37	**Detected 2 of 2**

Results of performance evaluation for all tested machine and deep learning classifiers are illustrated in Tab. 4, based on four classification metrics, namely accuracy, precision, sensitivity or recall, and F1-score, as presented above in Eqs. (1)–(4). The best classification metrics has been achieved by our proposed deep forest model, resulting the highest accuracy score of 98.0%. 1D CNN model achieved the second-highest accuracy value of 95%. Traditional RF and KNN classifiers give the middle performance of accuracy values of 94.0% and 93.0%, respectively, while the decision tree algorithm failed to correctly identify fall cases with the lowest accuracy of 88.0%. The performance of dense neural network is lower than the 1D CNN, achieving 91.0% accuracy.

Table 4: Evaluated performance metrics of all tested fall detection models in this study

Fall detection model	Class	Precision	Recall (Sensitivity)	F1-score	Accuracy (%)
K-nearest neighbors	Non-fall	0.96	0.97	0.96	93.0
	Fall	0.29	0.23	0.26	
Decision tree	Non-fall	0.96	0.91	0.93	88.0
	Fall	0.16	0.30	0.21	
Random forest	Non-fall	0.95	0.98	0.97	94.0
	Fall	0.35	0.20	0.25	
Deep dense network	Non-fall	0.96	0.94	0.95	91.0
	Fall	0.23	0.29	0.25	
1D convolutional neural network	Non-fall	0.95	**1.00**	0.97	95.0
	Fall	0.81	0.12	0.22	
Proposed deep forest	Non-fall	**0.98***	0.99	**0.99**	**98.0**
	Fall	**83.0**	**0.72**	**0.78**	

Note: *Bold values present best classifier performance.

Furthermore, Tab. 5 illustrates a comparison between the performances of our proposed deep forest model and other machine learning and deep learning models in previous studies using the same public dataset. Obviously, the deep forest classifier showed a superior performance by achieving the highest accuracy score of 98.0%, while the traditional deep learning model using CNN architecture achieved the lowest accuracy value of 86.0%. Lightweight CNN presents the second-best accuracy value of approximately 97% for detecting fall events and can be useful in the application of mobile embedded systems with limited computing resources.

Table 5: Comparative performance of proposed deep forest with other fall detection models in the literature using the same dataset in this study

Fall detection model	Accuracy (%)
Deep learning model [26]	86.00
Long short-term memory (LSTM) [40]	93.46
Lightweight convolutional neural network [41]	96.79
Proposed deep forest	**98.00***

Note: *Bold values present best classifier performance.

5 Discussion

Compared to classical machine learning and/or deep learning classifiers, the results of this study demonstrated that our deep forest classifier is efficient and accurate to identify fall cases of individuals, as illustrated in Tabs. 4 and 5. The proposed classifier achieved the highest accuracy of 98.0% using the public smartwatch dataset. It is assumed to be the main module of our suggested medical IoT framework as depicted in Fig. 3. The main reasons of the successful performance of our cascade forest classifier can be presented as follows. First, the cascade structure of ensemble forest model allows to produce new features from the original input features as depicted in Fig. 2. Second, automatic number estimation of deep forest levels or layers (See Tab. 3) supports a good fitting to complex dataset such as fall classification data. Finally, number of proposed deep forest classifier parameters is minimal compared to the large number of parameters in deep neural networks like CNN. Therefore, the parameters of deep forest model can be easily tuned by the user.

However, the estimation of the optimal number of the deep forest classifier layers takes a long time even in the presence of early stopping criteria. But proposed medical IoT-based framework facilitates the computing tasks with multi-GPUs by utilizing cloud computing services as shown in Fig. 3. In addition, the hyperparameter tuning of the deep forest model can be automatically accomplished using bio-inspired optimization approaches, such as Particle Swarm Optimization (PSO) and Genetic Algorithms (GAs) [42]. Nevertheless, the optimization procedure is a time-consuming task to find the best group of the model hyperparameters.

Advanced technologies of security and privacy solutions for fall detection systems have been recently proposed at the medical IoT edge computing level [43], which are not considered in our study nor other classical fall detection devices. Additionally, secure IoT-based healthcare systems should include the encryption and decryption processes, which become essential module for processing real-time tasks. There are still open areas of research such as federated learning [44], explainable and/or generalizable deep learning [45] and edge IoT layer [46] for developing future versions of fall detection systems. However, our proposed medical IoT framework for accurate fall detection of old people is still accurate and valid for real-time application deployment.

6 Conclusion and Future Work

Fall detection technologies are important tools for the care of the elderly patients and are critical to their health and well-being. As a result, it's necessary to come up with innovative and effective ways for preventing and reducing the detrimental effects of human falls. This work proposes an intelligent fall detection system that is both simple and computationally efficient. The suggested medical IoT appliance makes use of low-cost sensors that may be worn and installed in residential houses and other facilities. The proposed system allows our deep forest algorithm and IoMT devices to detect the occurrence of elderly patients falls in smart homes with high accuracy score of 98.0%. In our future work, the performance of fall detection classifier can be improved by utilizing other advanced lightweight DL models for classification process with various structures and parameters. Also, more datasets can be used for the training and experimental improvements.

Acknowledgement: The authors extend their appreciation to the Deputyship for Research & Innovation, Ministry of Education in Saudi Arabia for funding this research work through the Project Number (IFP2021-043).

Funding Statement: This research received the support from the Deputyship for Research & Innovation, Ministry of Education in Saudi Arabia through the Project Number (IFP2021-043).

Conflicts of 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.

References

[1] Falls. "World health organization," 2021. Accessed on 25 April 2022. [Online]. Available: https://www.who.int/en/news-room/fact-sheets/detail/falls.

[2] N. Mozaffari, J. Rezazadeh, R. Farahbakhsh, S. Yazdani and K. Sandrasegaran, "Practical fall detection based on IoT technologies: A survey," *Internet of Things*, vol. 8, pp. 1–40, 2019.

[3] X. Wang, J. Ellul and G. Azzopardi, "Elderly fall detection systems: A literature survey," *Frontiers in Robotics and AI*, vol. 7, no. 71, pp. 1–23, 2020.

[4] A. K. Bourke and G. M. Lyons, "A Threshold-based fall-detection algorithm using a bi-axial gyroscope sensor," *Medical Engineering & Physics*, vol. 30, no. 1, pp. 84–90, 2008.

[5] Q. T. Huynh, U. D. Nguyen, L. B. Irazabal, N. Ghassemian and B. Q. Tran, "Optimization of an accelerometer and gyroscope-based fall detection algorithm," *Journal of Sensors*, vol. 2015, pp. 1–8, 2015.

[6] L. Tong, Q. Song, Y. Ge and M. Liu, "HMM-Based human fall detection and prediction method using tri-axial accelerometer," *IEEE Sensors Journal*, vol. 13, no. 5, pp. 1849–1856, 2013.

[7] P. Pierleoni, L. Pernini, A. Belli, L. Palma, S. Valenti *et al.,* "SVM-Based fall detection method for elderly people using android low-cost smartphones," in *2015 IEEE Sensors Applications Symp. (SAS)*, Zadar, Croatia, pp. 1–5, 2015.

[8] B. Aguiar, T. Rocha, J. Silva and I. Sousa, "Accelerometer-based fall detection for smartphones," in *2014 IEEE Int. Symp. on Medical Measurements and Applications (MeMeA)*, Lisboa, Portugal, pp. 1–6, 2014.

[9] A. Shojaei-Hashemi, P. Nasiopoulos, J. J. Little and M. T. Pourazad, "Video-based human fall detection in smart homes using deep learning," in *2018 IEEE Int. Symp. on Circuits and Systems (ISCAS)*, Florence, Italy, pp. 1–5, 2018.

[10] Q. Han, H. Zhao, W. Min, H. Cui, X. Zhou *et al.,* "A Two-stream approach to fall detection with MobileVGG," *IEEE Access*, vol. 8, pp. 17556–17566, 2020.

[11] D. Mrozek, A. Koczur and B. Małysiak-Mrozek, "Fall detection in older adults with mobile IoT devices and machine learning in the cloud and on the edge," *Information Sciences*, vol. 537, pp. 132–147, 2020.

[12] S. K. Bhoi, S. K. Panda, B. Patra, B. Pradhan, P. Priya *et al.,* "FallDS-IoT: A fall detection system for elderly healthcare based on IoT data analytics," in *2018 Int. Conf. on Information Technology (ICIT)*, Bhubaneswar, India, pp. 155–160, 2018.

[13] D. Yacchirema, J. Suárez de Puga, C. Palau and M. Esteve, "Fall detection system for elderly people using IoT and big data," *Procedia Computer Science*, vol. 130, pp. 603–610, 2018.

[14] A. Tahir, W. Taylor, A. Taha, M. Usman, S. A. Shah *et al.,* "IoT based fall detection system for elderly healthcare," in *Internet of Things for Human-Centered Design, Studies in Computational Intelligence*, vol. 1011, Singapore: Springer, pp. 209–232, 2022.

[15] M. Pech, H. Sauzeon, T. Yebda, J. Benois-Pineau and H. Amieva, "Falls detection and prevention systems in home care for older adults: Myth or reality?," *JMIR Aging*, vol. 4, no. 4, pp. 1–8, 2021.

[16] R. Tanwar, N. Nandal, M. Zamani and A. A. Manaf, "Pathway of trends and technologies in fall detection: A systematic review," *Healthcare*, vol. 10, no. 1, pp. 1–27, 2022.

[17] X. Kong, Z. Meng, N. Nojiri, Y. Iwahori, L. Meng *et al.,* "A HOG-SVM based fall detection IoT system for elderly persons using deep sensor," *Procedia Computer Science*, vol. 147, pp. 276–282, 2019.

[18] D. Ajerla, S. Mahfuz and F. Zulkernine, "A Real-time patient monitoring framework for fall detection," *Wireless Communications and Mobile Computing*, vol. 2019, pp. 1–13, 2019.

[19] S. Nooruddin, M. Islam and F. A. Sharna, "An IoT based device-type invariant fall detection system," *Internet of Things*, vol. 9, pp. 1–23, 2020.

[20] G. Kyriakopoulos, S. Ntanos, T. Anagnostopoulos, N. Tsotsolas, I. Salmon *et al.,* "Internet of things (IoT)-enabled elderly fall verification, exploiting temporal inference models in smart homes," *International Journal of Environmental Research and Public Health*, vol. 17, no. 2, pp. 1–14, 2020.

[21] D. K. Cahoolessur and B. Rajkumarsingh, "Fall detection system using XGBoost and IoT," *R&D Journal of the South African Institution of Mechanical Engineering*, vol. 36, pp. 8–18, 2020.

[22] T. Vaiyapuri, E. L. Lydia, M. Y. Sikkandar, V. G. Díaz, I. V. Pustokhina *et al.,* "Internet of things and deep learning enabled elderly fall detection model for smart homecare," *IEEE Access*, vol. 9, pp. 113879–113888, 2021.

[23] M. Waheed, H. Afzal and K. Mehmood, "NT-FDS—A noise tolerant fall detection system using deep learning on wearable devices," *Sensors*, vol. 21, no. 6, pp. 1–26, 2021.

[24] O. Ribeiro, L. Gomes and Z. Vale, "IoT-Based human fall detection system," *Electronics*, vol. 11, pp. 1–10, 2022.

[25] A. S. Alluhaidan, M. Alajmi, F. N. Al-Wesabi, A. M. Hilal, M. A. Hamza *et al.,* "Improved archimedes optimization algorithm with deep learning empowered fall detection system," *Computers, Materials & Continua*, vol. 72, no. 2, pp. 2713–2727, 2022.

[26] T. R. Mauldin, M. E. Canby, V. Metsis, A. H. H. Ngu and C. C. Rivera, "SmartFall: A smartwatch-based fall detection system using deep learning," *Sensors*, vol. 18, no. 10, pp. 1–19, 2018.

[27] T. R. Mauldin, A. H. H. Ngu, V. Metsis, M. E. Canby and J. Tesic, "Experimentation and analysis of ensemble deep learning in IoT applications," *Open Journal of Internet of Things*, vol. 5, no. 1, pp. 133–149, 2019.

[28] L. Breiman, "Random forests," *Machine Learning*, vol. 45, no. 1, pp. 5–32, 2001.

[29] L. Zhu, X. Zhou and C. Zhang, "Rapid identification of high-quality marine shale gas reservoirs based on the oversampling method and random forest algorithm," *Artificial Intelligence in Geosciences*, vol. 2, pp. 76–81, 2021.

[30] M. W. Abo El-Soud, T. Gaber, F. AlFayez and M. M. Eltoukhy, "Implicit authentication method for smartphone users based on rank aggregation and random forest," *Alexandria Engineering Journal*, vol. 60, no. 1, pp. 273–283, 2021.

[31] X. Zhang, C. Zhou, X. Zhu, Z. Tao and H. Zhao, "Class-imbalanced voice pathology classification: Combining hybrid sampling with optimal two-factor random forests," *Applied Acoustics*, vol. 190, pp. 108618, 2022.

[32] Z. -H. Zhou and J. Feng, "Deep forest," *National Science Review*, vol. 6, no. 1, pp. 74–86, 2019.

[33] F. T. Liu, K. M. Ting, Y. Yu and Z. -H. Zhou, "Spectrum of variable-random trees," *Journal of Artificial Intelligence Research*, vol. 32, no. 1, pp. 355–384, 2008.

[34] M. E. Karar, A. -H. Abdel-Aty, F. Algarni, M. Fadzil Hassan, M. A. Abdou *et al.,* "Smart IoT-based system for detecting RPW larvae in date palms using mixed depthwise convolutional networks," *Alexandria Engineering Journal*, vol. 61, no. 7, pp. 5309–5319, 2022.

[35] T. -T. Wong, "Performance evaluation of classification algorithms by k-fold and leave-one-out cross validation," *Pattern Recognition*, vol. 48, no. 9, pp. 2839–2846, 2015.

[36] M. E. Karar, H. I. Shehata and O. Reyad, "A survey of IoT-based fall detection for aiding elderly care: Sensors, methods, challenges and future trends," *Applied Sciences*, vol. 12, no. 7, pp. 3276, 2022.

[37] M. E. Karar, B. Alotaibi and M. Alotaibi, "Intelligent medical IoT-enabled automated microscopic image diagnosis of acute blood cancers," *Sensors*, vol. 22, no. 6, pp. 1–16, 2022.

[38] M. Sokolova and G. Lapalme, "A systematic analysis of performance measures for classification tasks," *Information Processing & Management*, vol. 45, no. 4, pp. 427–437, 2009.

[39] A. Gulli, A. Kapoor and S. Pal, *Deep Learning with TensorFlow 2 and Keras: Regression, ConvNets, GANs, RNNs, NLP, and More with TensorFlow 2 and the Keras API*, 2nd ed., Packt Publishing, Birmingham, United Kingdom, 2019.

[40] M. Farsi, "Application of ensemble RNN deep neural network to the fall detection through IoT environment," *Alexandria Engineering Journal*, vol. 60, no. 1, pp. 199–211, 2021.

[41] S. K. Tasoulis, G. I. Mallis, S. V. Georgakopoulos, A. G. Vrahatis, V. P. Plagianakos *et al.,* "Deep learning and change detection for fall recognition," in *Engineering Applications of Neural Networks*, Cham: Springer, pp. 262–273, 2019.

[42] C. B. Gonçalves, J. R. Souza and H. Fernandes, "CNN architecture optimization using bio-inspired algorithms for breast cancer detection in infrared images," *Computers in Biology and Medicine*, vol. 142, pp. 105205, 2022.

[43] A. Tahir, W. Taylor, A. Taha, M. Usman, S. A. Shah *et al.,* "IoT based fall detection system for elderly healthcare," in *Internet of Things for Human-Centered Design: Application to Elderly Healthcare*, Singapore: Springer, pp. 209–232, 2022.

[44] C. Zhang, Y. Xie, H. Bai, B. Yu, W. Li *et al.,* "A survey on federated learning," *Knowledge-Based Systems*, vol. 216, pp. 106775, 2021.

[45] X. Bai, X. Wang, X. Liu, Q. Liu, J. Song *et al.,* "Explainable deep learning for efficient and robust pattern recognition: A survey of recent developments," *Pattern Recognition*, vol. 120, pp. 108102, 2021.

[46] V. Divya and R. L. Sri, "Docker-based intelligent fall detection using edge-fog cloud infrastructure," *IEEE Internet of Things Journal*, vol. 8, no. 10, pp. 8133–8144, 2021.